# FreeBSD гарын авлага

Товч агуулга

FreeBSD-д тавтай морилно уу! Энэ гарын авлага нь *FreeBSD 11.2-RELEASE* болон *FreeBSD 12.0-RELEASE* хувилбаруудын суулгацаас эхлээд өдөр тутмын хэрэглээг тайлбарласан билээ. Энэ гарын авлага нь *төгс дуусаагүй* бөгөөд маш олон хүний хамтын бүтээл юм. Зарим хэсэг нь хуучирч шинэчлэгдэх шаардлагатай байж болзошгүй. Хэрэв та энэ төсөлд хувь нэмрээ оруулж, бидэнд туслахыг хүсвэл FreeBSD баримтжуулах төслийн захидлын жагсаалт хаяг руу захиа бичнэ үү. Энэ гарын авлагын хамгийн сүүлийн хувилбарыг FreeBSD вэб хуудаснаас татаж авч болно. (энэ гарын авлагын өмнөх хувилбаруудыг http://docs.FreeBSD.org/doc/-с авах боломжтой). Мөн янз бүрийн шахаж бэлтгэсэн хэлбэрээр FreeBSD FTP серверээс эсвэл толин тусгал хуудаснаас татаж авагдаж болно. Хэрэв та энэ гарын авлагын хэвлэсэн хуулбарыг авахыг хүсвэл FreeBSD Mall хуудсанд зорчино уу. Та мөн гарын авлага дотор хайлт хийх боломжтой.

# Гарчиг

Өмнөх үг . . . 11
Энэ номын зориулалт . . . 11
Гурав дахь хэвлэлээс хойшх өөрчлөлтүүд . . . 11
Хоёр дахь хэвлэлээс (2004) хойшх өөрчлөлтүүд . . . 11
Эхний хэвлэлээс (2001) хойшх өөрчлөлтүүд . . . 12
Энэ номын зохион байгуулалт . . . 13
Энэ номонд ашиглагдсан бичиглэлийн хэлбэрүүд . . . 17
Талархал . . . 18
I: Эхлэл . . . 19
1. Танилцуулга . . . 20
1.1. Ерөнхий агуулга . . . 20
1.2. FreeBSD-д тавтай морилно уу! . . . 20
1.3. FreeBSD Төслийн тухай . . . 23
2. FreeBSD 8.X болон түүнээс өмнөх хувилбар суулгах нь . . . 30
2.1. Ерөнхий агуулга . . . 30
2.2. Системд тавигдах төхөөрөмжийн шаардлага . . . 30
2.3. Суулгацын өмнө . . . 31
2.4. Суулгацыг эхлүүлэх . . . 39
2.5. Sysinstall-н танилцуулга . . . 46
2.6. Дискний зайг зохицуулах . . . 51
2.7. Юу суулгахаа сонгох . . . 67
2.8. Суулгацын төхөөрөмжөө сонгох . . . 69
2.9. Суулгацыг баталж гүйцээх . . . 71
2.10. Суулгацын дараах тохиргоо . . . 72
2.11. Хүндрэл тулгарвал . . . 106
2.12. Суулгацын нэмэлт гарын авлага . . . 110
2.13. Өөртөө зориулж тусгай суулгацын төхөөрөмж бэлдэх нь . . . 112
3. FreeBSD 9.X болон түүнээс хойшхи хувилбаруудыг суулгах нь . . . 120
3.1. Ерөнхий агуулга . . . 120
3.2. Тоног төхөөрөмжийн шаардлагууд . . . 120
3.3. Суулгахын өмнөх ажлууд . . . 121
3.4. Суулгалтыг эхлүүлэх нь . . . 126
3.5. bsdinstall-г танилцуулах нь . . . 133
3.6. Сүлжээнээс суулгах нь . . . 137
3.7. Дискний зай авах нь . . . 138
3.8. Суулгалтыг гүйцэтгэх нь . . . 147
3.9. Суулгасны дараах үйлдлүүд . . . 150
3.10. Асуудлыг олж засварлах . . . 179

3.11. Амьд CD ашиглах нь . . . 181
4. Юниксийн үндэс . . . 182
4.1. Ерөнхий агуулга . . . 182
4.2. Виртуал консол болон терминал . . . 182
4.3. Зөвшөөрлүүд . . . 186
4.4. Сангийн бүтэц . . . 191
4.5. Диск зохион байгуулалт . . . 194
4.6. Файл системийг холбох болон салгах . . . 201
4.7. Процесс буюу програмын явц . . . 204
4.8. Далд чөтгөр буюу дэмонууд, дохионууд, мөн процессуудыг зогсоох нь . . . 206
4.9. Shell буюу бүрхүүл . . . 208
4.10. Текст засварлагчид . . . 211
4.11. Төхөөрөмж ба төхөөрөмжийн цэгүүд . . . 212
4.12. Хоёртын хэлбэрүүд . . . 212
4.13. Нэмэлт мэдээлэл олж авах нь . . . 214
5. Програм суулгах: Багцууд болон портууд . . . 217
5.1. Ерөнхий агуулга . . . 217
5.2. Програм суулгацын ерөнхий ойлголт . . . 217
5.3. Програмаа олох нь . . . 219
5.4. Багцалсан системийг хэрэглэх нь . . . 221
5.5. pkgng-г бинар багцыг удирдахад ашиглах нь . . . 224
5.6. Портын цуглуулгыг хэрэглэх нь . . . 230
5.7. Порт суулгасны дараах үйлдлүүд . . . 242
5.8. Эвдрэлтэй портыг засах нь . . . 243
6. X Цонхот систем . . . 244
6.1. Ерөнхий агуулга . . . 244
6.2. X-г таньж мэдэх . . . 244
6.3. X11-г суулгах нь . . . 247
6.4. X11-н тохируулга . . . 248
6.5. X11 дээр үсгийн маяг хэрэглэх нь . . . 254
6.6. X Display Manager буюу харуулалт зохицуулагч . . . 259
6.7. Дэлгэцийн орчин . . . 261
II: Нийтлэг асуудлууд . . . 267
7. Ширээний програмууд . . . 268
7.1. Ерөнхий агуулга . . . 268
7.2. Хөтчүүд . . . 269
7.3. Бүтээмж . . . 274
7.4. Бичиг баримт үзүүлэгчид . . . 279
7.5. Санхүү . . . 281
7.6. Дүгнэлт . . . 283
8. Мультимедиа . . . 284

8.1. Ерөнхий агуулга . . . 284
8.2. Дууны картыг тохируулах . . . 285
8.3. MP3 аудио . . . 290
8.4. Видео тоглуулах . . . 293
8.5. TV картуудыг тохируулах . . . 302
8.6. MythTV . . . 304
8.7. Дүрс скан хийгчид . . . 305
9. FreeBSD цөмийг тохируулах нь . . . 311
9.1. Ерөнхий агуулга . . . 311
9.2. Яагаад өөрчлөн тохируулсан цөм бүтээх хэрэгтэй вэ? . . . 311
9.3. Системийн тоног төхөөрөмж хайж олох нь . . . 312
9.4. Цөмийн драйверууд, дэд системүүд, болон модулиуд . . . 313
9.5. Өөрчлөн тохируулсан цөмийг бүтээх ба суулгах нь . . . 314
9.6. Тохиргооны файл . . . 317
9.7. Хэрэв ямар нэг юм буруутвал . . . 334
10. Хэвлэлт . . . 336
10.1. Ерөнхий агуулга . . . 336
10.2. Танилцуулга . . . 336
10.3. Үндсэн тохируулга . . . 337
10.4. Хэвлэгчийн илүү нарийн тохиргоо . . . 353
10.5. Хэвлэгчдийг ашиглах нь . . . 385
10.6. Хэвлэгчийн стандарт түр хадгалагчидтай ижил хадгалагчид . . . 393
10.7. Алдааг олж засварлах . . . 394
11. Линуксийн Хоёртын Нийлэмж . . . 398
11.1. Ерөнхий агуулга . . . 398
11.2. Суулгалт . . . 398
11.3. Mathematica® суулгах нь . . . 402
11.4. Maple™ суулгах нь . . . 405
11.5. MATLAB® суулгах нь . . . 407
11.6. Oracle® суулгах нь . . . 410
11.7. Нэмэлт сэдвүүд . . . 414
III: Системийн Удирдлага . . . 417
12. Тохиргоо ба Тааруулалт . . . 418
12.1. Ерөнхий агуулга . . . 418
12.2. Эхний Тохиргоо . . . 418
12.3. Гол Тохиргоо . . . 420
12.4. Програмын Тохиргоо . . . 421
12.5. Үйлчилгээнүүдийг эхлүүлэх нь . . . 421
12.6. `cron` хэрэгслийг тохируулах нь . . . 423
12.7. FreeBSD дээр rc(8) ашиглах нь . . . 425
12.8. Сүлжээний интерфэйс картууд суулгах нь . . . 427

12.9. Виртуал Хостууд . . . 434
12.10. Системийн лог хийгч syslogd-г тохируулах нь . . . 435
12.11. Тохиргооны Файлууд . . . 438
12.12. sysctl(8) ашиглан тааруулах нь . . . 441
12.13. Дискнүүдийг тааруулах нь . . . 442
12.14. Цөмийн хязгаарууд тохируулах нь . . . 446
12.15. Swap зай нэмэх нь . . . 450
12.16. Тэжээл болон Эх үүсвэрийн Удирдлага . . . 452
12.17. FreeBSD-ийн ACPI-г ашиглах нь ба дибаг хийх нь . . . 454
13. FreeBSD-ийн Ачаалах процесс . . . 462
13.1. Ерөнхий агуулга . . . 462
13.2. Ачаалалтын асуудал . . . 462
13.3. Ачаалагч Менежер болон Ачаалалтын шатууд . . . 463
13.4. Ачаалах үе дэх цөмийн харилцан үйлдэл . . . 469
13.5. Төхөөрөмжийн Сануулгууд . . . 470
13.6. Init: Процесс хяналтын эхлүүлэлт . . . 471
13.7. Унтраах дараалал . . . 472
14. Хэрэглэгчид ба үндсэн бүртгэл зохицуулалт . . . 473
14.1. Ерөнхий агуулга . . . 473
14.2. Танилцуулга . . . 473
14.3. Супер хэрэглэгчийн бүртгэл . . . 475
14.4. Систем бүртгэл . . . 475
14.5. Энгийн хэрэглэгчийн бүртгэл . . . 475
14.6. Бүртгэлд өөрчлөлт хийх . . . 476
14.7. Хэрэглэгчдэд хязгаарлалт хийх . . . 480
14.8. Бүлэг . . . 483
15. Аюулгүй байдал . . . 485
15.1. Ерөнхий агуулга . . . 485
15.2. Танилцуулга . . . 485
15.3. FreeBSD-н аюулгүй байдлыг хангах нь . . . 487
15.4. DES, Blowfish, MD5, SHA256, SHA512 болон Crypt . . . 497
15.5. Нэг удаагийн нууц үгүүд . . . 497
15.6. TCP Гүйцэтгэлийг хялбаршуулагчид . . . 501
15.7. Kerberos5 . . . 505
15.8. OpenSSL . . . 514
15.9. IPsec дээгүүр VPN хийх . . . 517
15.10. OpenSSH . . . 524
15.11. Файлын системийн хандалт хянах жагсаалтууд(ACL-үүд) . . . 530
15.12. Гуравдагч талын аюулгүй байдлын асуудлуудыг монитор хийх нь . . . 532
15.13. FreeBSD-ийн аюулгүй байдлын зөвлөгөөнүүд . . . 534
15.14. Процессийн бүртгэл хөтлөх . . . 536

16. Jails буюу Шоронгууд . . . 538
16.1. Ерөнхий агуулга . . . 538
16.2. Шоронгуудтай холбоотой ойлголтууд . . . 538
16.3. Танилцуулга . . . 539
16.4. Шоронг үүсгэж хянах нь . . . 540
16.5. Нарийн тааруулалт болон удирдлага . . . 543
16.6. Шоронгийн хэрэглээ . . . 544
17. Mandatory Access Control буюу Албадмал Хандалтын хяналт . . . 552
17.1. Ерөнхий агуулга . . . 552
17.2. Энэ бүлэг дэх түлхүүр ухагдахуунууд . . . 553
17.3. MAC-ийн тайлбар . . . 555
17.4. MAC хаяг/шошгонуудыг ойлгох нь . . . 556
17.5. Аюулгүй байдлын тохиргоог төлөвлөх нь . . . 562
17.6. Модулийн тохиргоо . . . 563
17.7. MAC seeotheruids модуль . . . 563
17.8. MAC bsdextended модуль . . . 564
17.9. MAC ifoff модуль . . . 565
17.10. MAC portacl модуль . . . 565
17.11. MAC хуваалтын модуль . . . 567
17.12. MAC олон түвшинт аюулгүй байдлын модуль . . . 568
17.13. MAC Biba модуль . . . 570
17.14. MAC LOMAC модуль . . . 572
17.15. MAC Шорон дахь Nagios . . . 573
17.16. Хэрэглэгчийг түгжих . . . 577
17.17. MAC Тогтолцооны алдааг олж засварлах . . . 578
18. Аюулгүй байдлын үйл явцад аудит хийх нь . . . 580
18.1. Ерөнхий агуулга . . . 580
18.2. Энэ бүлгийн түлхүүр ухагдахуунууд . . . 581
18.3. Аудит хийх дэмжлэг суулгах нь . . . 582
18.4. Аудитийн тохиргоо . . . 582
18.5. Аудит дэд системийг удирдах нь . . . 585
19. Хадгалалт . . . 589
19.1. Ерөнхий агуулга . . . 589
19.2. Төхөөрөмжийн нэрс . . . 589
19.3. Диск нэмэх . . . 590
19.4. RAID . . . 593
19.5. USB хадгалалтын төхөөрөмжүүд . . . 598
19.6. Оптик зөөвөрлөгчийг (CD-үүд) үүсгэж ашиглах нь . . . 601
19.7. Оптик зөөвөрлөгчийг (DVD-үүд) үүсгэж ашиглах нь . . . 608
19.8. Уян дискнүүдийг үүсгэж ашиглах нь . . . 614
19.9. Өгөгдлийн соронзон хальснууд үүсгэж ашиглах нь . . . 616

19.10. Уян диск уруу нөөцлөх . . . 619
19.11. Нөөцлөх стратегууд . . . 620
19.12. Нөөцлөлтийн үндсүүд . . . 621
19.13. Сүлжээ, санах ой болон файл дээр тулгуурласан файлын системүүд . . . 626
19.14. Файлын системийн хормын хувилбарууд . . . 629
19.15. Файлын системийн ноогдлууд . . . 631
19.16. Дискний хуваалтуудыг шифрлэх нь . . . 635
19.17. Swap зайг шифрлэх . . . 643
19.18. Highly Available Storage (HAST) . . . 644
20. GEOM: Διαχείριση Συστοιχιών Δίσκων . . . 653
20.1. Ерөнхий агуулга . . . 653
20.2. GEOM-ийн Танилцуулга . . . 653
20.3. RAID0 - Судал үүсгэх . . . 653
20.4. RAID1 - Толин тусгал үүсгэх . . . 655
20.5. RAID3 - Parity бүхий байтын түвшний тусдаа судалжуулалт (Striping) . . . 665
20.6. GEOM Хаалга Сүлжээний Төхөөрөмжүүд . . . 667
20.7. Дискний төхөөрөмжүүдийг хаяглах нь . . . 668
20.8. GEOM-ийн тусламжтай UFS тэмдэглэл . . . 671
21. Файлын системийн дэмжлэг . . . 673
21.1. Ерөнхий агуулга . . . 673
21.2. Z Файлын Систем (ZFS) . . . 673
21.3. Linux® файлын системүүд . . . 684
22. Vinum Эзлэхүүн Менежер . . . 686
22.1. Ерөнхий агуулга . . . 686
22.2. Дискнүүд хэтэрхий жижиг . . . 686
22.3. Хандалтын тагларалтууд . . . 687
22.4. Өгөгдлийн бүрэн бүтэн байдал . . . 688
22.5. Vinum обьектууд . . . 689
22.6. Зарим жишээнүүд . . . 691
22.7. Обьектийн нэрлэлт . . . 698
22.8. Vinum тохируулах нь . . . 700
22.9. Root файлын системийн хувьд Vinum-ийг ашиглах нь . . . 701
23. Virtualisation . . . 707
23.1. Ерөнхий агуулга . . . 707
23.2. FreeBSD-г зочин OS маягаар ашиглах . . . 707
23.3. FreeBSD-г хост буюу эх маягаар ашиглах . . . 739
24. Локалчлал - I18N/L10N-ийн хэрэглээ болон тохируулга . . . 742
24.1. Ерөнхий агуулга . . . 742
24.2. Үндсүүд . . . 742
24.3. Локалчлалыг ашиглах нь . . . 743
24.4. I18N програмуудыг эмхэтгэх . . . 750

24.5. FreeBSD-г өөр хэлнүүд уруу локалчлах . . . 750
25. FreeBSD-г шинэчилж сайжруулах нь . . . 754
25.1. Ерөнхий агуулга . . . 754
25.2. FreeBSD-ийн шинэчлэлт . . . 754
25.3. Portsnap: Портын цуглуулгыг шинэчлэх хэрэгсэл . . . 762
25.4. Баримтын цуглуулгыг шинэчлэх нь . . . 763
25.5. Хөгжүүлэлтийн салбарыг дагах нь . . . 768
25.6. Өөрийн эхийг хамгийн сүүлийн хэлбэрт аваачих нь . . . 773
25.7. "Ертөнц"ийг дахин бүтээх нь . . . 774
25.8. Хуучин файлууд, хавтаснууд болон сангуудыг устгах . . . 792
25.9. Олон машины хувьд дагах нь . . . 793
26. DTrace . . . 796
26.1. Ерөнхий агуулга . . . 796
26.2. Шийдлийн ялгаанууд . . . 796
26.3. DTrace дэмжлэгийг идэвхжүүлэх . . . 797
26.4. DTrace-г ашиглах . . . 798
26.5. D хэл . . . 801
IV: Сүлжээний Холболт . . . 802
27. Цуваа холбоонууд . . . 803
27.1. Ерөнхий агуулга . . . 803
27.2. Танилцуулга . . . 803
27.3. Терминалууд . . . 808
27.4. Dial-in буюу гаднаас залгах үйлчилгээ . . . 813
27.5. Dial-out буюу гадагш залгах үйлчилгээ . . . 822
27.6. Цуваа консолыг тохируулах нь . . . 826
28. PPP болон SLIP . . . 835
28.1. Ерөнхий агуулга . . . 835
28.2. Хэрэглэгчийн PPP ашиглах . . . 835
28.3. Цөмийн PPP-г ашиглах . . . 849
28.4. PPP холболтуудын алдааг олж засварлах . . . 857
28.5. Ethernet дээгүүр PPP ашиглах нь (PPPoE) . . . 861
28.6. ATM дээгүүр PPP (PPPoA) ашиглах . . . 863
28.7. SLIP ашиглах . . . 867
29. Цахим Шуудан . . . 877
29.1. Ерөнхий Агуулга . . . 877
29.2. Цахим Захидлыг Хэрэглэх нь . . . 877
29.3. sendmail-г Тохируулах нь . . . 880
29.4. Өөрийн Захидал Дамжуулах Агентийг Солих нь . . . 883
29.5. Гэмтэл саатлыг арилгах нь . . . 885
29.6. Дээд түвшний сэдвүүд . . . 889
29.7. UUCP-тэй SMTP . . . 891

29.8. Зөвхөн илгээхээр тохируулах нь . . . 893
29.9. Гадагш залгах холболтоор захидлыг ашиглах нь . . . 894
29.10. SMTP нэвтрэлт танилт . . . 896
29.11. Хэрэглэгчийн захидал дамжуулагчид . . . 897
29.12. fetchmail-г ашиглах нь . . . 905
29.13. procmail-г ашиглах нь . . . 906
30. Сүлжээний орчны Серверүүд . . . 908
30.1. Ерөнхий агуулга . . . 908
30.2. inetd"Супер-Сервер" . . . 908
30.3. Сүлжээний Файлын Систем (NFS) . . . 913
30.4. Сүлжээний Мэдээллийн Систем (NIS/YP) . . . 920
30.5. Автомат Сүлжээний Тохиргоо (DHCP) . . . 939
30.6. Домэйн Нэрийн Систем (DNS) . . . 944
30.7. Apache HTTP Сервер . . . 963
30.8. Файл Дамжуулах Протокол (FTP) . . . 969
30.9. Microsoft® Windows® харилцагчдад зориулсан Файл болон Хэвлэх Үйлчилгээ (Samba) . . . 970
30.10. ntpd-р Цаг Тааруулах нь . . . 973
30.11. `syslogd` ашиглан алсын хост руу бүртгэх нь . . . 976
31. Галт хана . . . 982
31.1. Танилцуулга . . . 982
31.2. Галт ханын тухай ойлголтууд . . . 982
31.3. Галт ханын багцууд . . . 983
31.4. OpenBSD Пакет шүүгч (PF) ба ALTQ . . . 983
31.5. IPFILTER (IPF) Галт хана . . . 987
31.6. IPFW . . . 1009
32. Сүлжээний нэмэлт ойлголтууд . . . 1030
32.1. Ерөнхий агуулга . . . 1030
32.2. Гарцууд болон Чиглүүлэлтүүд . . . 1030
32.3. Утасгүй сүлжээ . . . 1038
32.4. Bluetooth . . . 1059
32.5. Гүүр . . . 1068
32.6. Холбоос нэгтгэлт ба ослыг тойрон гарах (Link Aggregation and Failover) . . . 1075
32.7. Дискгүй ажиллагаа . . . 1080
32.8. NFS root файлын систем ашиглан PXE ачаалалт хийх . . . 1088
32.9. ISDN . . . 1092
32.10. Network Address Translation буюу Сүлжээний Хаягийн Хөрвүүлэлт . . . 1097
32.11. IPv6 . . . 1101
32.12. Asynchronous Transfer Mode (ATM) буюу Асинхрон Дамжуулах Горим . . . 1106
32.13. Common Address Redundancy Protocol (CARP) . . . 1108
33. Хавсралтууд . . . 1112

Appendix A: Se procurer FreeBSD . . . 1113
A.1. CDROM болон DVD Нийтлэгчид . . . 1113
A.2. FTP сайтууд . . . 1114
A.3. Нэргүй CVS(хуучирсан) . . . 1120
A.4. CTM-г ашиглах нь . . . 1120
A.5. Subversion ашиглах нь . . . 1124
A.6. Subversion толин сайтууд . . . 1126
A.7. CVSup-г ашиглах нь (хуучирсан) . . . 1128
A.8. CVS Tags буюу хаягууд . . . 1136
A.9. rsync сайтууд . . . 1143
Appendix B: Ном зүй . . . 1145
B.1. FreeBSD-ийн талаар тусгайлан бичсэн ном & сэтгүүлүүд . . . 1145
B.2. Хэрэглэгчдийн гарын авлагууд . . . 1146
B.3. Администраторууд зориулсан зааврууд . . . 1146
B.4. Програм зохиогчдод зориулсан зааврууд . . . 1146
B.5. Үйлдлийн системийн дотоод бүрэлдэхүүнүүдийн талаар . . . 1147
B.6. Аюулгүй байдлын тухай номнууд . . . 1148
B.7. Тоног төхөөрөмжийн тухай номнууд . . . 1148
B.8. UNIX®-ийн түүх . . . 1148
B.9. Сонин, сэтгүүлүүд . . . 1149
Appendix C: Интернэт дэх эх үүсвэрүүд . . . 1150
C.1. Захидлын жагсаалтууд . . . 1150
C.2. Usenet Newsgroups буюу мэдээний бүлгүүд . . . 1170
C.3. Word Wide Web серверүүд . . . 1171
C.4. Цахим захидлын хаягууд . . . 1171
Appendix D: PGP түлхүүрүүд . . . 1173
D.1. Officers . . . 1173

# Өмнөх үг

## Энэ номын зориулалт

FreeBSD-ийн шинэ хэрэглэгч нь энэ номын эхний хэсэг хэрэглэгчид FreeBSD-ийн суулгалтын процесс дундуур хөтөлж UNIX®-ийн үгт байдаг ойлголтууд болон ёс заншлыг аажим танилцуулдаг болохыг олж мэдэх болно. Энэ хэсэг дээр ажиллах нь шинээр олж мэдэх хүсэл эрмэлзлээс арай илүү хүсэл болон танилцуулсан шинэ ойлголтуудыг хүлээн авах чадварыг шаарддаг.

Гарын авлагын хоёр дахь хамаагүй том хэсэг хүрэх үед FreeBSD системийн администраторуудын сонирхлыг татах бүх сэдвүүдийг хамарсан нэвтэрхий лавлахыг та олох болно. Эдгээр бүлгүүдийн зарим нь таныг урьдчилан уншихыг зөвлөж болох бөгөөд энэ нь бүлэг бүрийн эхэнд ерөнхий агуулгад тэмдэглэгдсэн байдаг.

Мэдээллийн нэмэлт эхүүдийн жагсаалтыг Ном зүй хэсгээс үзнэ үү.

# Гурав дахь хэвлэлээс хойшх өөрчлөлтүүд

Гарын авлагын одоогийн хувилбар нь сүүлийн 10 жилийн хугацааны туршид олон зуун хувь нэмэр болгон оруулагчийн оруулсан хүчин чармайлтын үр дүн юм. Дараах нь 2004 онд хэвлэгдсэн гурав дахь хэвлэлийн хоёр ботиос хойш орсон мэдэгдэхүйц өөрчлөлтүүд юм:

- DTrace буюу DTrace нь хүчирхэг DTrace үйл ажиллагааны шинжилгээ хийх хэрэгслийн талаарх мэдээллийн хамтаар нэмэгдсэн.
- Файлын системийн дэмжлэг буюу Файлын системийн дэмжлэг нь Sun™-ий ZFS зэрэг FreeBSD дээрх эх биш файлын системүүдийн талаарх мэдээллийн хамтаар нэмэгдсэн.
- Аюулгүй байдлын үйл явцад аудит хийх нь Аюулгүй байдлын үйл явцад аудит хийх бүлэг FreeBSD дэх аудит хийх шинэ боломж болон түүний хэрэглээг тайлбарсан мэдээллийн хамтаар нэмэгдсэн.
- Виртуалчлал буюу Виртуалчлал нь виртуалчлалын програм хангамж дээр FreeBSD-ийг суулгах мэдээллийн хамтаар нэмэгдсэн.
- FreeBSD 9.X болон түүнээс хойшхи хувилбаруудыг суулгах нь буюу bsdinstall гэсэн шинэ суулгалтын хэрэгсэл ашиглан FreeBSD-г суулгах талаар FreeBSD 9.*x* болон түүнээс хойшхи хувилбарыг суулгах нь бүлэгт бичигдэн нэмэгдсэн.

# Хоёр дахь хэвлэлээс (2004) хойшх өөрчлөлтүүд

Энэ гурав дахь хэвлэлт нь FreeBSD баримтжуулах төслийн гишүүдийн хоёр жил илүүтэй хугацаанд хийж гүйцэтгэсэн ажлын оргил юм. Хэвлэсэн хувилбарын хэмжээ ихэссэн учир хоёр тусдаа боть болгох шаардлагатай болсон. Энэ шинэ хэвлэлтэд орсон гол өөрчлөлтүүдийг доор дурдав:

- Тохиргоо ба Тааруулалт Тохиргоо ба тааруулалт нь ACPI тэжээл болон эх үүсвэрийн удирдлага, `cron` системийн хэрэгсэл болон цөмийн илүү олон тааруулах тохируулгуудын

тухай шинэ мэдээллээр өргөжсөн.

- Аюулгүй байдал Аюулгүй байдал нь виртуал хувийн сүлжээнүүд (VPN-үүд), файлын системийн хандалт хяналтын жагсаалт (ACL-үүд) болон аюулгүй байдлын зөвлөгөөнүүдийн тухай шинэ мэдээллээр өргөжсөн.
- Mandatory Access Control буюу Албадмал Хандалтын хяналт Mandatory Access Control буюу Албадмал Хандалтын хяналт нь энэ хэвлэл дэх шинэ бүлэг юм. Энэ нь MAC гэж юу болох, FreeBSD системийг аюулгүй болгоход энэ аргыг хэрхэн ашиглах талаар тайлбарладаг.
- Хадгалалт Хадгалалт нь USB хадгалалтын төхөөрөмжүүд, файлын системийн хормын хувилбарууд, файлын системийн хувь/ноогдлууд, файл болон сүлжээн дээр тулгуурласан файлын системүүд болон шифрлэгдсэн дискний хуваалтуудын тухай шинэ мэдээллээр өргөжсөн.
- Vinum Эзлэхүүн Менежер Vinum нь энэ хэвлэл дэх шинэ бүлэг юм. Энэ нь Vinum-ийг хэрхэн ашиглах, төхөөрөмжийн хувьд хамааралгүй логик дискнүүдийн боломжийг хангадаг логик эзлэхүүн менежер болон програм хангамжийн RAID-0, RAID-1 болон RAID-5-г тайлбарладаг.
- Алдааг олж засварлах хэсэг PPP болон SLIP PPP ба SLIP-д нэмэгдсэн.
- Цахим Шуудан Цахим шуудан нь өөр тээвэрлэх агентуудыг ашиглах, SMTP нэвтрэлт танилт, UUCP, fetchmail, procmail, болон бусад дэвшилтэт сэдвүүдийн тухай шинэ мэдээллээр өргөжсөн.
- Сүлжээний орчны Серверүүд Сүлжээний серверүүд нь энэ хэвлэлд бүгд шинээр орсон. Энэ бүлэг нь Apache HTTP Server, ftpd-г тохируулах болон Microsoft® Windows® клиентүүдийн хувьд Samba серверийг тохируулах тухай мэдээллийг агуулдаг. Сүлжээний нэмэлт ойлголтууд Сүлжээний нэмэлт ойлголтууд дахь зарим хэсгүүд нь танилцуулгыг сайжруулахын тулд ийшээгээ зөөгдсөн.
- Сүлжээний нэмэлт ойлголтууд Сүлжээний нэмэлт ойлголтууд нь FreeBSD дээр Bluetooth® төхөөрөмжүүдийг ашиглах, утасгүй сүлжээнүүд болон Асинхрон Дамжуулах Горим (ATM) сүлжээг тохируулах тухай шинэ мэдээллээр өргөжсөн.
- Номонд ашиглагдсан техникийн ухагдахуунуудын тодорхойлолтуудад зориулагдсан төв байршлыг бий болгон нэр томъёо хэсэг нэмэгдсэн.
- Номонд байгаа хүснэгтүүд болон зургуудад хэд хэдэн гоо зүйн сайжруулалтууд хийгдсэн.

# Эхний хэвлэлээс (2001) хойшх өөрчлөлтүүд

Хоёр дахь хэвлэлт нь FreeBSD баримтжуулах төслийн гишүүдийн хоёр жил илүүтэй хугацаанд хийж гүйцэтгэсэн ажлын оргил юм. Энэ хэвлэлтэд орсон гол өөрчлөлтүүдийг доор дурдав:

- Бүрэн хэмжээний үгсийн жагсаалт нэмэгдсэн.
- Бүх ASCII зурагнууд график диаграмуудаар солигдсон.
- Бүлэг ямар мэдээлэл агуулсан болох, уншигч юу мэдэхийг харуулсан товч мэдээллийг өгөхийн тулд бүлэг болгонд стандарт ерөнхий агуулга нэмэгдсэн.
- Агуулга логикийн хувьд гурван хэсэгт дахин зохион байгуулагдсан: "Эхлэл", "Системийн

удирдлага", болон "Хавсралтууд".

- FreeBSD 8.X болон түүнээс өмнөх хувилбар суулгах нь ("FreeBSD-г суулгах нь") шинэ хэрэглэгчдэд текстийг хялбар авах боломжийг бүрдүүлэх олон дэлгэцийн агшнуудтайгаар бүрэн хэмжээгээр дахин бичигдсэн.
- Юниксийн үндэс ("UNIX®-ийн үндэс") процессууд, дэмонууд болон дохионуудын тухай нэмэлт мэдээллийг агуулж өргөжсөн.
- Програм суулгах. Багцууд болон портууд ("Програм суулгах") хоёртын багцын удирдлагын тухай нэмэлт мэдээллийг агуулж өргөжсөн.
- X Цонхот систем ("X Цонхот систем") нь орчин үеийн ширээний технологиуд болох KDE болон GNOME-г XFree86™ 4.X дээр ашиглах тал дээр онцлон анхаарч бүрэн хэмжээгээр дахин бичигдсэн.
- FreeBSD-ийн Ачаалах процесс ("FreeBSD-ийн ачаалах процесс") нь өргөжсөн.
- Хадгалалт ("Хадгалалт") өмнө нь тусдаа байсан "Дискнүүд" болон "Нөөцлөлтүүд" гэсэн хоёр бүлгээс бичигдсэн. Сэдвүүдийг нэг бүлэг болгон танилцуулах нь хялбар болохыг бид мэдэрсэн. RAID-ийн тухай хэсэг (тоног төхөөрөмж болон програм хангамж) бас нэмэгдсэн.
- Цуваа холбоонууд ("Цуваа холболтууд") FreeBSD 4.X/5.X-ийн хувьд бүрэн хэмжээгээр дахин зохион байгуулагдаж шинэчлэгдсэн.
- PPP болон SLIP ("PPP ба SLIP") нэлээн хэмжээгээр шинэчлэгдсэн.
- Олон хэсэг Сүлжээний нэмэлт ойлголтууд-д ("Сүлжээний тухай нэмэлт ойлголтууд") нэмэгдсэн.
- Цахим Шуудан ("Цахим шуудан") sendmail-г тохируулах тухай илүү мэдээллийг агуулж өргөжсөн.
- Линуксийн Хоёртын Нийлэмж ("Linux®-ийн хоёртын нийлэмж") Oracle® болон SAP® R/3®-г суулгах тухай мэдээллийг агуулж өргөжсөн.
- Дараах шинэ сэдвүүдийг энэ хоёр дахь хэвлэлд хэлэлцсэн:
  - Тохиргоо ба тааруулалт (Тохиргоо ба Тааруулалт).
  - Мультимедиа (Мультимедиа)

# Энэ номын зохион байгуулалт

Энэ ном нь таван логикийн хувьд тусдаа хэсгүүдэд хуваагдана. Эхний хэсэг *Эхлэл* нь FreeBSD-ийн суулгалт болон үндсэн хэрэглээг тайлбарлах болно. Эдгээр бүлгүүдийг дарааллаар нь магадгүй мэддэг сэдвүүдийг тайлбарлах бүлгүүдийг орхин уншина гэдэгт уншигч авхай нарт итгэдэг. Хоёр дахь хэсэг *Нийтлэг асуудлууд* нь FreeBSD-ийн зарим нэг байнга ашиглагддаг боломжуудыг тайлбарладаг. Энэ хэсэг болон дараачийн хэсгүүдийг дараалал хамаарахгүйгээр уншиж болно. Бүлэг бүр нь бүлэг юу тайлбарладаг болон уншигч юуг аль хэдийн мэдсэн байх ёстойг тайлбарласан хураангуй агуулгаас эхэлдэг. Энэ нь уншигчид өөрийн сонирхсон бүлгээ олоход туслах зорилготой юм. Гурав дахь хэсэг *Системийн удирдлага* нь удирдлагын сэдвүүдийг хөнддөг. Дөрөв дэх хэсэг *Сүлжээний холболт* нь сүлжээ болон серверийн сэдвүүдийг хамардаг. Тав дахь хэсэг нь лавлагаа мэдээллийн тухай хавсралтуудыг агуулдаг.

*Танилцуулга* ***Танилцуулга***

FreeBSD-г шинэ хэрэглэгчид танилцуулна. Энэ нь FreeBSD төслийн түүх, түүний зорилгууд болон хөгжүүлэлтийн загварыг тайлбарладаг.

*FreeBSD 8.X болон түүнээс өмнөх хувилбар суулгах нь* ***буюу FreeBSD 8._x болон түүнээс өмнөх хувилбарыг суулгах нь_***

Хэрэглэгчийг FreeBSD 8.*x* болон түүнээс өмнөх хувилбарын суулгалтын процесс дундуур хөтөлнө. Цуваа консолоор суулгах зэрэг илүү нарийн суулгалтын сэдвүүдийг бас тайлбарлана.

*FreeBSD 9.X болон түүнээс хойшхи хувилбаруудыг суулгах нь* ***буюу FreeBSD 9._x болон түүнээс хойшхи хувилбарыг суулгах нь_***

Хэрэглэгчийг bsdinstall ашиглан FreeBSD 9.*x* болон түүнээс хойшхи хувилбарын суулгалтын процесс дундуур хөтөлнө.

*Юниксийн үндэс*, ***UNIX®-ийн үндэс***

FreeBSD үйлдлийн системийн үндсэн тушаалууд болон ажиллагааг тайлбарладаг. Хэрэв та Linux® эсвэл UNIX®-ийн өөр төрлийг мэддэг бол та энэ бүлгийг алгасаж болох юм.

*Програм суулгах. Багцууд болон портууд* ***Програм суулгах нь***

FreeBSD-ийн шинэ санаачлага "портын цуглуулга" болон стандарт хоёртын багцуудын тусламжтай гуравдагч програм хангамжуудын суулгалтыг тайлбарладаг.

*X Цонхот систем* ***X Цонхот систем***

X Цонхот системийг ерөнхийд нь болон FreeBSD-г ялангуяа X11 дээр ашиглах талаар тайлбарлах болно. KDE болон GNOME зэрэг ширээний түгээмэл орчнуудыг бас тайлбарладаг.

*Ширээний програмууд* ***Ширээний програмууд***

Вэб хөтчүүд болон бүтээмжийн цуглуулгууд зэрэг зарим нэгэн ширээний програмуудыг дурдах бөгөөд тэдгээрийг FreeBSD дээр хэрхэн суулгах талаар тайлбарлах болно.

*Мультимедиа* ***Мультимедиа***

Дуу болон дүрс тоглуулалтын дэмжлэгийг өөрийн систем дээр хэрхэн тохируулахыг үзүүлнэ. Бас зарим жишээ аудио болон видео програмуудыг тайлбарладаг.

*FreeBSD цөмийг тохируулах нь* ***FreeBSD цөмийг тохируулах нь***

Танд яагаад шинэ цөм хэрэгтэй болж болохыг тайлбарлаж цөмийг тохируулах, бүтээх болон суулгах тухай дэлгэрэнгүй зааврыг өгнө.

*Хэвлэлт* ***Хэвлэлт***

Сурталчилгаа хуудаснууд, хэвлэгчийн бүртгэл болон эхний тохиргоо зэрэг мэдээллийг оруулаад хэвлэгчдийг FreeBSD дээр удирдах талаар тайлбарладаг.

*Линуксийн Хоёртын Нийлэмж*, ***Linux®-ийн хоёртын нийлэмж***

FreeBSD-ийн Linux®-тэй нийцтэй боломжуудыг тайлбарладаг. Бас Oracle®, болон Mathematica® зэрэг Linux®-ийн олон түгээмэл програмуудыг суулгах дэлгэрэнгүй зааврыг өгдөг.

*Тохиргоо ба Тааруулалт* ***Тохиргоо ба тааруулалт***

FreeBSD-г хамгийн оновчтой ажиллагаанд зориулж тааруулахад туслах системийн администраторуудад зориулсан параметрүүдийг тайлбарладаг. Бас FreeBSD-д ашиглагдах төрөл бүрийн тохиргооны файлуудыг тайлбарлаж тэдгээрийг хаанаас олохыг хэлж өгдөг.

*FreeBSD-ийн Ачаалах процесс* ***Эхлүүлэх процесс***

FreeBSD-ийн эхлүүлэх процессийг тайлбарлаж тохиргооны сонголтуудын тусламжтай энэ процессийг хэрхэн хянах талаар дурддаг.

*Хэрэглэгчид ба үндсэн бүртгэл зохицуулалт* ***Хэрэглэгчид ба үндсэн бүртгэл зохицуулалт***

Хэрэглэгчийн бүртгэлүүдийг үүсгэх болон удирдахыг тайлбарладаг. Бас хэрэглэгчид дээр эх үүсвэрийн хязгаарлалтуудыг тохируулах болон бусад бүртгэл удирдах ажлуудын талаар тайлбарладаг.

*Аюулгүй байдал* ***Аюулгүй байдал***

Kerberos, IPsec, болон OpenSSH зэргийг оролцуулаад таны FreeBSD системийг аюулгүй болгоход туслах олон хэрэгслүүдийг тайлбарладаг.

*Jails буюу Шоронгууд* ***Jails буюу Шоронгууд***

Шоронгийн тогтолцоо болон FreeBSD-ийн уламжлалт chroot дэмжлэг дээр нэмэгдсэн шоронгийн сайжруулалтуудыг тайлбарладаг.

*Mandatory Access Control буюу Албадмал Хандалтын хяналт* ***Mandatory Access Control буюу Албадмал Хандалтын хяналт***

Mandatory Access Control (MAC) буюу Албадмал Хандалтын хяналт гэж юу болохыг тайлбарладаг бөгөөд FreeBSD системийг аюулгүй болгоход энэ арга замыг хэрхэн ашиглах талаар хэлэлцдэг.

*Аюулгүй байдлын үйл явцад аудит хийх нь* ***Аюулгүй байдлын үйл явцад аудит хийх нь***

FreeBSD-ийн үйл явцын аудит гэж юу болох, түүнийг хэрхэн суулгах, тохируулах болон аудит мөрнүүд хэрхэн шалгагдаж эсвэл монитор хийгддэг талаар тайлбарладаг.

*Хадгалалт* ***Хадгалалт***

Хадгалалтын зөөвөрлөгч болон файлын системүүдийг FreeBSD-ээр хэрхэн удирдахыг тайлбарладаг. Үүнд физик дискнүүд, RAID массивууд, оптик болон соронзон хальсан зөөвөрлөгч, санах ой дээр тулгуурласан дискнүүд болон сүлжээний файлын системүүд ордог.

*GEOM. Модульчлагдсан Диск Хувиргах Тогтолцоо* ***GEOM***

FreeBSD дээрх GEOM тогтолцоо гэж юу болох, төрөл бүрийн дэмжигдсэн RAID түвшнүүдийг хэрхэн тохируулах талаар тайлбарладаг.

*Файлын системийн дэмлэг* ***Файлын системийн дэмлэг***

Sun™-ий Z файлын систем зэрэг FreeBSD дээрх төрөлхийн биш файлын системүүдийн дэмжлэгийн талаар тайлбарладаг.

*Vinum Эзлэхүүн Менежер* ***Vinum***

Төхөөрөмжөөс хамааралгүй логик дискнүүд болон програм хангамжийн RAID-0, RAID-1 болон RAID-5 зэргийн боломжийг олгодог логик эзлэхүүн менежер Vinum-ийг хэрхэн ашиглах талаар тайлбарладаг.

*Виртуалчлал* ***Виртуалчлал***

Виртуалчлалын системүүд юу санал болгодог болон тэдгээрийг FreeBSD-тэй хэрхэн ашиглаж болохыг тайлбарладаг.

*Локалчлал - I18N/L10N-ийн хэрэглээ болон тохируулга* ***Локалчлал***

Англи хэлнээс бусад хэлнүүд дээр FreeBSD-г хэрхэн ашиглах талаар тайлбарладаг. Систем болон програмын түвшингийн локалчлалыг дурддаг.

*FreeBSD-г шинэчилж сайжруулах нь* ***FreeBSD-г шинэчилж сайжруулах нь***

FreeBSD-STABLE, FreeBSD-CURRENT болон FreeBSD-ийн хувилбаруудын ялгаануудыг тайлбарладаг. Хөгжүүлэлтийн системийг дагахад ямар хэрэглэгчдэд ашигтайг тайлбарлаж тэр процессийг дурддаг. Хэрэглэгчид өөрсдийн системийг аюулгүй байдлын хамгийн сүүлийн үеийн хувилбар руу шинэчлэх аргуудыг тайлбарладаг.

*DTrace* ***DTrace***

FreeBSD дээр Sun™-ий DTrace хэрэгслийг хэрхэн тохируулж хэрэглэх талаар тайлбарладаг. Динамикаар дагаж мөрдөх нь ажиллаж байх явцад системийн шинжилгээ хийж гүйцэтгэн үйл ажиллагааны асуудлуудыг олоход тусалж чадна.

*Цуваа холбоонууд* ***Цуваа холбоонууд***

Терминалууд болон модемуудыг өөрийн FreeBSD системд гадагш залгах болон дуудлага хүлээн авах холболтуудад зориулж хэрхэн холбох талаар тайлбарладаг.

*PPP болон SLIP* ***PPP ба SLIP***

FreeBSD дээр PPP, SLIP, эсвэл Ethernet дээгүүрх PPP-г хэрхэн ашиглаж алсын системд холбогдохыг тайлбарладаг.

*Цахим шуудан* ***Цахим шуудан***

Цахим захидлын серверийн бүрэлдэхүүн хэсгүүдийг тайлбарлаж хамгийн түгээмэл захидлын програм хангамж болох sendmail-д зориулсан хялбар тохиргооны сэдвүүдийн талаар өгүүлдэг.

*Сүлжээний орчны Серверүүд* ***Сүлжээний серверүүд***

Таны FreeBSD машиныг сүлжээний файлын систем сервер, домэйн нэрний сервер, сүлжээний мэдээллийн систем сервер эсвэл цаг тааруулах сервер болгон хэрхэн тохируулах тухай дэлгэрэнгүй зааврууд болон жишээ тохиргооны файлуудыг дурддаг.

*Галт хана* ***Галт хананууд***

Програм хангамж дээр суурилсан галт ханануудын цаадах философийг тайлбарлаж FreeBSD-д зориулсан төрөл бүрийн галт ханануудын тохиргооны тухай дэлгэрэнгүй мэдээллийг өгүүлдэг.

*Сүлжээний нэмэлт ойлголтууд* ***Сүлжээний нэмэлт ойлголтууд***

Өөрийн LAN доторх бусад компьютеруудтай Интернэтийн холболтоо хуваалцах, чиглүүлэлтийн дэвшилтэт сэдвүүд, утасгүй сүлжээ, Bluetooth®, ATM, IPv6 зэрэг зүйлсийг оролцуулаад сүлжээний олон сэдвүүдийг тайлбарладаг.

*FreeBSD-г олж авах нь* ***FreeBSD-г олж авах нь***

FreeBSD зөөвөрлөгчийг CDROM эсвэл DVD дээр олж авахад зориулагдсан төрөл бүрийн эхүүд болон FreeBSD-г татан авч суулгах боломжийг танд олгох Интернэт дэх олон сайтуудыг дурддаг.

*Ном зүй* ***Ном зүй***

Энэ ном нь маш олон төрөл бүрийн сэдвүүдийг хөнддөг бөгөөд таныг магадгүй илүү дэлгэрэнгүй мэдээлэл авах хүсэлд хөтлөж болох юм. Ном зүй нь энэ номонд ашигласан олон гайхалтай номнуудыг дурдсан байдаг.

*Интернэт дэх эх үүсвэрүүд* ***Интернэт дэх эх үүсвэрүүд***

FreeBSD-ийн хэрэглэгчдэд зориулсан, асуултууд илгээж FreeBSD-ийн талаар техникийн хэлэлцүүлгүүд өрнүүлж болох олон хэлэлцүүлгүүд байдаг талаар тайлбарладаг.

*PGP түлхүүрүүд* ***PGP түлхүүрүүд***

FreeBSD-ийн хэд хэдэн хөгжүүлэгчдийн PGP түлхүүрүүдийг жагсаадаг.

# Энэ номонд ашиглагдсан бичиглэлийн хэлбэрүүд

Текстийг уншихад хялбар, дэс дараалалтай болгохын тулд энэ номонд хэд хэдэн бичиглэлийн хэлбэрүүдийг ашигласан.

## Хэвлэлийн хэлбэрүүд

***Italic***

*italic* фонт нь файлын нэрс, URL-ууд, онцолсон текст болон техникийн ухагдахуунуудын эхний хэрэглээнд *italic* үсгийн маяг ашиглагдсан.

`Monospace`

`monospace` хийгдсэн үсгийн маяг нь алдааны мэдэгдлүүд, тушаалууд, орчны хувьсагчууд, портуудын нэрс, хостын нэрс, хэрэглэгчдийн нэрс, бүлгийн нэрс, төхөөрөмжийн нэрс, хувьсагчууд болон кодын хэсгүүдэд ашиглагдсан.

**Bold**

bold үсгийн маяг нь програмууд, тушаалууд болон товчлуурууд ашиглагдсан.

## Хэрэглэгчийн Ороолт

Товчлуурууд нь бусад текстээс ялгарахын тулд **bold** хэлбэрээр харуулагдсан. Товчлуурууд хослолууд нь зэрэг бичигдэх ёстой бөгөөд товчлуурууд хооронд + тавигдан үзүүлэгдсэн, жишээ нь:

`Ctrl` + `Alt` + `Del`

Энэ нь хэрэглэгч `Ctrl`, `Alt`, болон `Del` товчлууруудыг зэрэг дарах ёстой гэсэн үг юм.

Дараалан бичих ёстой товчлууруудыг таслалаар тусгаарладаг, жишээ нь:

`Ctrl` + `X`, `Ctrl` + `S`

Энэ нь хэрэглэгч `Ctrl` болон `X` товчлууруудыг зэрэг дараад дараа нь `Ctrl` болон `S` товчлууруудыг зэрэг дарна гэсэн үг юм.

### Жишээнүүд

E:\> гэж эхэлсэн жишээнүүд нь MS-DOS® тушаалыг илэрхийлнэ. Тэмдэглэж хэлээгүй л бол эдгээр тушаалуудыг орчин үеийн Microsoft® Windows® орчны "Тушаал хүлээх мөр" цонхноос ажиллуулж болох юм.

```
E:\> tools\fdimage floppies\kern.flp A:
```

# гэж эхэлсэн жишээнүүд нь FreeBSD дээр супер хэрэглэгчээр ажиллуулах ёстой тушаалыг илэрхийлнэ. Тушаал бичихийн тулд та `root` эрхээр нэвтрэх юм уу эсвэл өөрийн ердийн эрхээр нэвтрээд супер хэрэглэгчийн зөвшөөрлүүдийг авахын тулд su(1)-г ашиглаж болох юм.

```
# dd if=kern.flp of=/dev/fd0
```

% гэж эхэлсэн жишээнүүд нь ердийн хэрэглэгчийн бүртгэлээс ажиллуулах ёстой тушаалыг илэрхийлнэ. Тэмдэглэж хэлээгүй л бол орчны хувьсагчууд болон бусад бүрхүүлийн тушаалуудыг тохируулахад C бүрхүүлийн синтакс ашиглагддаг.

```
% top
```

# Талархал

Таны барьж байгаа ном нь дэлхий даяарх олон зуун хүмүүсийн хүч хөдөлмөр юм. Бичгийн алдаануудын засварууд илгээсэн ч бай эсвэл бүхэл бүтэн бүлгүүдийг ирүүлсэн ч бай гэсэн эдгээр бүх оруулсан хувь нэмрүүд нь их ач холбогдолтой байсан юм.

Зохиогчдыг бүтэн ажлын өдрийн турш ажиллахыг дэмжин цалинжуулж, хэвлэлтийн төлбөрийг төлөх гэх зэргээр хэд хэдэн компаниуд энэ баримтын хөгжүүлэлтийг дэмжсэн билээ. Ялангуяа BSDi-ийн (дараахан нь Wind River Systems худалдан авсан) цалинжуулсан FreeBSD баримтжуулах төслийн гишүүд энэ номыг сайжруулж бүтэн ажлын өдрийн турш ажилласан нь 2000 оны 3 сард эхний хэвлэгдсэн хувилбар гаргахад хүргэсэн юм (ISBN 1-57176-241-8). Wind River Systems дараа нь хэвлэгдсэн гаралтын дэд бүтцэд хэд хэдэн сайжруулалтуудыг хийлгэж текстэд нэмэлт бүлгүүдийг нэмүүлэхийн тулд хэд хэдэн зохиогчийг цалинжуулсан билээ. 2001 оны 11 сард хэвлэгдсэн хоёр дахь хэвлэлд энэ ажил оржээ (ISBN 1-57176-303-1). 2003-2004 онуудад FreeBSD Mall, Inc гурав дахь удаагийн хэвлэлийг бэлтгэхэд зориулан Гарын авлагыг сайжруулахын тулд хэд хэдэн хувь нэмэр оруулагчдыг цалинжуулсан юм.

# Part I: Эхлэл

FreeBSD гарын авлагын энэ хэсэг нь FreeBSD-тэй шинээр танилцаж байгаа администратор болон хэрэглэгчдэд зориулагдсан. Энд гарах бүлгүүд нь:

- FreeBSD-н танилцуулга.
- FreeBSD-г суулгах үйл явц.
- UNIX®-ийн үндсийн ойлголт.
- FreeBSD-тэй хамт ирдэг гуравдагч-этгээдийн програмыг хэрхэн суулгах болон
- UNIX® цонхот систем X-ийг танд танилцуулж, илүү үр бүтээлтэй ажиллахын тулд ажлын ширээний орчноо хэрхэн тохируулах талаар тайлбарлана.

Бид Гарын авлагынхаа энэ хэсгийг аль болох цөөхөн хуудас дамжихаар бодолцсон бөгөөд ингэснээр танд энэ гарын авлагыг эхнээс нь дуустал хуудаснаас хуудас уруу үсрэлгүйгээр уншиж дуусгахад зориулсан билээ.

# Chapter 1. Танилцуулга

## 1.1. Ерөнхий агуулга

FreeBSD-г сонирхсон танд баярлалаа! Дараах бүлэг FreeBSD төслийн түүх, зорилго, хөгжүүлэх загвар зэрэг бусад төрөл бүрийн ойлголтуудыг хамарна.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD бусад компьютерийн үйлдлийн системүүдтэй ямар хамааралтай талаар.
- FreeBSD төслийн түүх.
- FreeBSD төслийн зорилгууд.
- FreeBSD-ийн нээлттэй-эх хөгжүүлэх загварын үндсүүд.
- Мөн мэдээж "FreeBSD" нэр хаанаас гарсан талаар.

## 1.2. FreeBSD-д тавтай морилно уу!

FreeBSD нь Интел (x86 болон Itanium®), AMD64, Sun UltraSPARC® компьютеруудад зориулагдсан 4.4BSD-Lite дээр үндэслэсэн үйлдлийн систем юм. Мөн өөр бусад архитектур уруу порт хийгдэж байгаа болно. Та мөн FreeBSD-ийн түүх эсвэл одоогийн хувилбар ыг унших боломжтой. Хэрэв та энэ төсөлд (код, тоног төхөөрөмж, сан) хувь нэмэр оруулахыг сонирхож байгаа бол FreeBSD-д хувь нэмэр оруулах нь нийтлэлийг уншина уу.

### 1.2.1. FreeBSD юу хийж чадах вэ?

FreeBSD олон тооны дурдахад бүрүүдахгүй боломжуудтай. Эдгээрийн зарим нь:

- Компьютерийг их ачаалалтай байсан ч програмууд болон хэрэглэгчдийн хоорондбартаагүй, ижил хуваалцах нөхцлийг бүрдүүлэх, динамик дараалал тохируулалттай, *илүү өндөр давуу эрх бүхий бодлогоор солигдох боломж бүхий олон бодлогочлол*.
- *Олон-хэрэглэгчийн боломжууд* нь олон хүмүүс FreeBSD системийг төрөл бүрийн зүйлд зориулан зэрэгцэн хэрэглэх боломжийг зөвшөөрдөг. Энэ нь, жишээ нь, хэвлэгчүүд болон соронзон хальсны хөтлөгчүүд зэрэг системийн захын төхөөрөмжүүд нь систем эсвэл сүлжээн дэх бүх хэрэглэгчдийн дунд зөвөөр хуваалцах ба эх үүсвэр бүр дэх хязгаарууд нь маш чухал системийн эх үүсвэрүүдийг илүү ашиглахаас хамгаалж хэрэглэгчдэд болон бүлэг хэрэглэгчдэд тавигдаж болно гэсэн үг юм.
- SCTP, DHCP, NFS, NIS, PPP, SLIP, IPsec, болон IPv6 зэрэг үйлдвэрлэлийн стандартуудын дэмжлэгтэй, хүчирхэг *TCP/IP сүлжээ*. Энэ нь таны FreeBSD машин бусад үйлдлийн системүүдтэй хялбар харилцан ажиллаж чадахаас гадна мөн NFS (файлд алсаас хандах) цахим захидлын үйлчилгээнүүд зэрэг чухал боломжуудыг хангадаг эсвэл Интернэтэд WWW, FTP, чиглүүлэх эсвэл галт ханын (аюулгүй байдал) үйлчилгээнүүдийн хамт та өөрийнхөө байгууллагыг байрлуулах Enterprise сервер шиг ажиллаж чадна гэсэн үг юм.
- *Санах ойн хамгаалалт* нь програмууд (эсвэл хэрэглэгчид) өөр хоорондоо нөлөөлөхгүй нөхцлийг хангадаг. Нэг програм сүйрэхэд бусдад аль ч замаар нөлөөлөхгүй.

- FreeBSD нь *32-бит* үйлдлийн систем (Itanium®, AMD64, болон UltraSPARC® дээр *64-бит*) бөгөөд бүр анхнаасаа эхлэн дизайн хийгдсэн юм.
- Үйлдвэрлэлийн стандарт болох *X Цонхны Систем* (X11R7) нь ердийн VGA карт ба дэлгэцийн үнэнд хэрэглэгчийн график интерфэйсийг (GUI) хангадаг бөгөөд бүрэн эхийн хамт ирдэг.
- Линукс, SCO, SVR4, BSDI ба NetBSD-д зориулан бүтээгдсэн олон программуудтай *хоёртын нийлэмж*.
- Олон мянган *ажиллахад бэлэн* программууд FreeBSD-ийн *портууд* болон *багцуудын* цуглуулганд байдаг. Эндээс бүгдийг олж болж байхад яагаад сүлжээнээс хайх хэрэгтэй гэж?
- Олон мянган нэмэлт болон *амархан порт хийх* боломжтой программууд Интернэтэд байдаг. FreeBSD нь эх кодын хувьд ихэнх олны мэддэг арилжааны UNIX® системүүдтэй нийцтэй бөгөөд ихэнх программуудыг хөрвүүлэхэд хэрэв байгаа бол цөөн өөрчлөлтүүдийг шаарддаг.
- Шаардлагаар хуудасладаг (demand paged) *виртуал санах ой* ба "нийлүүлсэн VM/түр хадгалагч" дизайн нь санах ойн хувьд их хэрэглэдэг програмуудын хүслийг үр ашигтайгаар хангадаг бөгөөд ингэхдээ бусад хэрэглэгчдэд харилцан хариу өгөх боломжийг олгосоор байдаг.
- Олон CPU-тай машинуудад зориулсан *SMP* дэмжлэг.
- *C* ба *C++* хөгжүүлэх багажуудын бүрэн бүрэлдэхүүн. Нарийн судалгаа болон хөгжүүлэлтэд зориулсан олон нэмэлт хэлнүүд бас портууд болон багцуудын цуглуулганд байдаг.
- Бүх системд зориулсан *эх код* гэдэг нь та орчноо хамгийн ихээр хянана гэсэн үг юм. Та жинхэнэ нээлттэй системтэй мөртлөө яагаад үйлдвэрлэгчийн буянд хаалттай шийдэлд цоожтой байх ёстой гэж?
- Дэлгэрэнгүй *онлайн бариmтжуулалт*.
- *гэх мэт өөр илүү олныг дурдаж болно!*

FreeBSD нь Беркли дэх Калифорнийн Их Сургуулийн Компьютерийн Системийн Судалгааны Бүлгээс (CSRG) гаргасан 4.4BSD-Lite хувилбар дээр үндэслэсэн бөгөөд BSD системийн хөгжүүлэлтийн бусдаас ялгаатай уламжлалыг үргэлжлүүлсээр байна. CSRG-ээс гаргасан сайхан ажлаас гадна FreeBSD Төсөл нь жинхэнэ амьдрал дээрх ачааллын үеийн найдвартай болон хамгийн сайн ажиллагааг хангах системийг нарийн тохируулахад олон мянган цагийг зориулжээ. Улам олон арилжааны аваргууд ийм боломжууд, ажиллагаа болон найдвартай байдал бүхий PC үйлдлийн системийг гаргах гэж тэмцэж байхад FreeBSD тэдгээрийг *одоо* санал болгож чадна!

FreeBSD-г ямар хэрэглээнд ашиглах нь зөвхөн таны төсөөллийн хүрээнд байна. Програм хангамжийн хөгжүүлэлтээс эхлээд үйлдвэрлэлийн автоматжуулалт, төлвийн хяналтаас авахуулаад алсын хиймэл дагуулын антеннуудын азимут засварлалт; хэрэв үүнийг арилжааны UNIX® бүтээгдэхүүнээр хийж чадах бол мэдээж та үүнийг FreeBSD-ээр ч бас хийж чадна! Мөн FreeBSD дэлхий даяар судалгааны төвүүд болон их сургуулиудад хөгжүүлсэн ихэнхдээ бага эсвэл үнэгүй байдаг олон мянган өндөр чанарын програмуудаас мэдэгдэхүйц үр ашгийг авдаг. Арилжааны программууд бас байдаг бөгөөд өдөр ирэх тусам их хэмжээгээр нэмэгдэж байна.

FreeBSD-ийн өөрийн эх код ерөнхийдөө байгаа болохоор тусгай програмууд эсвэл төслүүдэд зориулж системийг бараг сонсоогүй хэмжээнд өөрчлөх боломжтой бөгөөд ерөнхийдөө бусад ихэнх том том арилжааны үйлдвэрлэгчдийн үйлдлийн системүүдэд үүнийг хийх боломжгүй байдаг. Энд FreeBSD-г ашиглаж байгаа хүмүүсийн хэрэглээний зарим нэг байна:

- *Интернэтийн Үйлчилгээнүүд:* FreeBSD дэх хүчирхэг TCP/IP сүлжээ нь төрөл бүрийн Интернэтийн үйлчилгээнүүдэд хамгийн тохирсон тавцан болгодог:
  - FTP серверүүд
  - Дэлхий Даяарх Вэб серверүүд (стандарт эсвэл нууцлаг [SSL])
  - IPv4 болон IPv6 чиглүүлэлт
  - Галт хананууд болон NAT ("IP маскарад хийх") гарцууд
  - Цахим Захидлын серверүүд
  - USENET Мэдээнүүд эсвэл Зарлалын Самбарын Системүүд
  - гэх мэт өөр илүү олныг дурдаж болно...

    Та FreeBSD-тэй байхад хямд 386 ангиллын PC-тэй жижгээс хялбархнаар эхлээд таны Enterprise өсөхийн хэрээр RAID хадгалалт бүхий 4 процессортой Xeon хүртэл шинэчилж болно.

- *Боловсрол:* Та компьютерийн шинжлэх ухаан эсвэл түүнтэй холбоотой инженерчлэлийн салбарын оюунтан уу? Үйлдлийн систем, компьютерийн архитектур болон сүлжээний талаар сурахад FreeBSD-ийн далд хангаж өгдөг боломж шиг илүү сайн арга байхгүй. Чөлөөтэй байдаг хэд хэдэн CAD, математикийн болон графикийн дизайн хийх багцууд нь *өөр бусад* ажлуудыг хийхийн тулд компьютерийг ашигладаг хүмүүст FreeBSD-г илүү их ашигтай болгодог!
- *Судалгаа:* Бүхэл системийн хувьд эх кодтой FreeBSD нь үйлдлийн системүүд болон компьютерийн шижлэх ухааны бусад салбаруудын хувьд судалгааны маш сайн тавцан болдог. FreeBSD-ийн чөлөөтэй байдаг чанар нь алсад байгаа бүлгүүд тусгай лицензийн гэрээ эсвэл нээлттэй форум дээр хэлэлцэж болох хязгаарууд санаа зоволгүйгээр шинэ санаанууд болон хуваалцсан хөгжүүлэлт дээр хамтран ажиллах боломжийг бүрдүүлдэг.
- *Сүлжээ:* Шинэ чиглүүлэгч? Нэрийн сервер (DNS)? Хүмүүсийг дотоод сүлжээнээс гадуур байлгах галт хана хэрэгтэй байна уу? FreeBSD нь ашиглаагүй, буланд хэвтэж байгаа 386 эсвэл 486 PC-г пакет шүүлт хийх сайжруулсан боломжуудтай өргөтгөсөн чиглүүлэгч болгон хялбархан хувиргаж чадна.
- *X Цонхны ажлын станц:* FreeBSD нь чөлөөтэй байдаг X11 серверийг ашиглан хямд X терминалын шийдэлд зориулсан сайн сонголт болдог. X терминалаас ялгаатай нь хэрэв хүсвэл олон програмуудыг FreeBSD ажиллуулахыг зөвшөөрдөг бөгөөд ингэснээр төв серверийн ачааллыг хөнгөвчилдөг. FreeBSD нь "дискгүй" ачаалж чадсанаар ажлын станцуудыг хямд, амархан удирдах боломжтой болгодог.
- *Програм Хангамжийн Хөгжүүлэлт:* Үндсэн FreeBSD систем нь нийтэд алдаршсан GNU C/C++ хөрвүүлэгч болон алдаа ологч зэрэг програмчлалын үндсэн хэрэгслүүдтэй ирдэг.

FreeBSD нь CD-ROM, DVD, болон нэргүй FTP дээр эх болон хоёртын хэлбэрээр байдаг. FreeBSD-г олж авах талаар дэлгэрэнгүй мэдээллийг FreeBSD-г олж авах нь хаягаас үзнэ үү.

### 1.2.2. FreeBSD-г хэн ашигладаг вэ?

FreeBSD нь дэлхийн IT-ийн ихэнх томоохон компаниудын гаргадаг төхөөрөмжүүд болон бүтээгдэхүүнүүдэд тавцан болон ашиглагддаг:

- Apple
- Cisco
- Juniper
- NetApp

FreeBSD нь бас интернэт дэх зарим нэг хамгийн томоохон вэб хуудсуудыг ажиллуулдаг:

- Yahoo!
- Yandex
- Apache
- Rambler
- Sina
- Pair Networks
- Sony Japan
- Netcraft
- NetEase
- Weathernews
- TELEHOUSE America
- Experts Exchange

гэх мэт өөр илүү олон вэб хуудсуудыг дурдаж болно.

## 1.3. FreeBSD Төслийн тухай

Дараах хэсэг төслийн товч түүх, төслийн зорилгууд,болон төслийн хөгжүүлэлтийн загвар зэрэг төслийн тухай зарим нэг мэдээллийг өгнө.

### 1.3.1. FreeBSD-ийн товч түүх

FreeBSD төслийн үүсэл 1993 оны эхэн хэсэгт үүдэлтэй бөгөөд залруулах багцын сүүлийн 3 зохицуулагчид болох Нэйт Вилльямс, Род Граймс болон миний биеийн хамтран гаргасан "Албан ёсны бус 386BSD Залруулах багц"аас хэсэгчилэн хөгжсөн юм.

Бидний анхны зорилго хэд хэдэн асуудлуудыг шийдэхийн тулд 386BSD-ийн дундын хормын хувилбарыг гаргаж авах явдал байсан бөгөөд залруулах багцын механизм үүнийг шийдэж чаддаггүй байлаа. Та нарын зарим нэг нь төслийн ажлын анхны нэр тэр явдалтай холбоотой "386BSD 0.5" эсвэл "386BSD Дундын" гэж байсныг санаж магадгүй юм.

386BSD нь бараг жил орчмын үл ойшоолтоос болж ихээхэн зовж байсан тэр үед хүрсэн Билл

Жолицийн үйлдлийн систем байсан юм. Залруулалтын багц нь өдөр өнгөрөх бүр улам эвгүйгээр томорч байсан бөгөөд бид ямар нэг юм хийх ёстойг 100 хувь зөвшөөрч Биллд туслахаар шийдэж энэхүү дундын "цэвэрлэсэн" хормын хувилбарыг гаргасан юм. Тэдгээр төлөвлөгөөнүүд нь Билл Жолицийг юу хийгдэж болохыг тодорхой харуулахын оронд ямар ч зүйлгүйгээр төслийн өөрийн санкцаас гэнэт татгалзах шийдвэрт хүргэж бүдүүлэг зогсоолтод хүргэсэн юм.

Биллийн дэмжлэггүй ч гэсэн зорилго маань харамсахааргүй хэвээрээ үлдэхийг бид нэг их удалгүй шийдсэн бөгөөд Дэйвид Грийнмэний бодож олсноор "FreeBSD" нэрийг авсан юм. Бидний эхний зорилгууд системийн одоогийн хэрэглэгчидтэй зөвлөлдсөний дараа тодорхойлогдсон бөгөөд төсөл эхэлсэн нь тодорхой болсны дараа бүр магадгүй амьдрал дээр биеллээ олохын тулд би Интернэт уруу хялбар хандах боломжгүй олон золгүйчүүдэд зориулж FreeBSD-ийн түгээлтийн сувгуудыг сайжруулах зорилготойгоор Волнат Крийк CDROM-той холбоо тогтоосон юм. Волнат Крийк CDROM нь FreeBSD-г CD дээр түгээх санааг дэмжээд зогсоогүй ажиллах машин бас хурдан Интернэтийн холболтыг төсөлд зориулан хангасан юм. Волнат Крийк CDROM-ийн тэр үед тэр чигээрээ үл мэдэгдэх төсөлд бараг л урьдчилан тооцох аргагүй өгсөн итгэлгүйгээр FreeBSD одоогийн хүрсэн шиг ийм хол түвшинд, ийм хурдан хүрэхгүй байсан биз ээ.

Анхны CD-ROM (ерөнхийдөө сүлжээнд өргөнөөр) түгээлт 1993 оны 12 сард гаргасан FreeBSD 1.0 байлаа. Энэ нь Берклигийн К.И.С-ийн 4.3BSD-Lite ("Net/2") соронзон хальс дээр тулгуурласан, бас 386BSD болон Чөлөөт Програм Хангамжийн Сангаас хангасан олон хэсгүүдээс тогтсон байсан билээ.Энэ бол эхний удаад нэлээн боломжийн амжилт байсан бөгөөд дараа нь бид маш өндөр амжилт олсон FreeBSD 1.1 хувилбарыг 1994 оны 5 сард гаргасан юм.

Энэ үед урт удаан үргэлжилсэн Берклигийн Net/2 соронкон хальны хууль эрх зүйн статустай холбоотой Новэлл ба Берклигийн К.И.С нарын хоорондох зарга шийдэгдэж гэнэтийн хар үүлс бий болсон юм. Тэр тохиролцооны нөхцөл нь "саад болсон" код ба Новэллийн урьд нь AT&T-ээс авсан өмч болох Net/2-ийн ихэнх хэсэг дээр хийсэн Берклигийн К.И.С-ийн буулт байлаа. Хариуд нь Новэллийн "адислал" 4.4BSD-Lite хувилбар байсан бөгөөд энэ нь гарсныхаа дараа саадгүйгээр тунхаглагдаж түүн уруу Net/2-ийн бүх хэрэглэгчид маш хүчтэйгээр шилжихийг дэмжих явдал байлаа. Үүнд FreeBSD бас орсон бөгөөд төсөлд 1994 оны 7 сараас хүртэл хугацаа өгч өөрийн Net/2 дээр тулгуурласан бүтээгдэхүүнээ гаргахыг хориглосон юм. Тэр гэрээний хүрээнд эцсийн хугацаанаас өмнө төсөлд сүүлийн нэг хувилбар гаргахыг зөвшөөрсөн бөгөөд тэр хувилбар нь FreeBSD 1.1.5.1 байлаа.

Дараа нь FreeBSD бараг тэр чигээрээ шинэ, бүрэн биш 4.4BSD-Lite-ийн хэсгээс өөрийгээ дахин бүтээх хүнд бэрх бодлогыг тогтоосон юм. "Lite" хувилбарууд нь зарим талаараа хөнгөн байсан, учир нь ачаалагдаж ажиллах системийг бүтээхэд шаардлагатай кодын ихээхэн хэсгийг Берклигийн CSRG арилгасан (төрөл бүрийн хууль эрх зүйн шаардлагаар) бөгөөд 4.4-ийн Интелийн хэсэг бас нэлээн бүрэн биш байлаа. Энэ шилжилтийг хийсээр төсөл 1994 оны 11 сар хүргэсэн бөгөөд тэр үед сүлжээ болон CD-ROM (12 сарын сүүлээр) дээр FreeBSD 2.0-ийг гаргасан байна. Хэдийгээр гадуураа бага зэрэг түүхий байсан боловч энэ хувилбар нь ихээхэн амжилттай болсон бөгөөд удалгүй илүү хүчирхэг, амархан суулгадаг FreeBSD 2.0.5 хувилбарыг 1995 оны 6 сард гаргасан билээ.

1996 оны 8 сард FreeBSD 2.1.5-ыг бид гаргасан бөгөөд энэ нь ISP болон арилжааны хүрээнийхэнд нэлээн алдаршсан ба 2.1-STABLE салбарын бас нэг өөр хувилбар сайшаагдсан

билээ. Энэ нь 1997 оны 2 сард гарсан FreeBSD 2.1.7.1 бөгөөд 2.1-STABLE -ийн зонхилох хөгжүүлэлтийн төгсгөл болсон юм. Одоо үйл ажиллагааг хангах горимд зөвхөн аюулгүй байдлын өргөжүүлэлт болон бусад ноцтой алдааны засварлалтууд энэ салбарын (RELENG_2_1_0) хувьд хийгдэх болно.

FreeBSD 2.2 нь хөгжүүлэлтийн гол салбараас ("-CURRENT") 1996 оны 11 сард RELENG_2_2 салбар болон салбарлан гарсан бөгөөд анхны бүрэн хувилбар (2.2.1) 1997 оны 4 сард гарсан юм. 2.2 салбарын дараагийн хувилбарууд 97 оны зун болон намар гарцгаасан бөгөөд тэдгээрийн сүүлийнх (2.2.8) 1998 оны 11 сард гарчээ. Анхны албан ёсны 3.0 хувилбар 1998 оны 10 сард гарсан бөгөөд 2.2 салбарын хувьд төгсгөл болох эхлэлийг тавьсан юм.

1999 оны 1 сарын 20-нд мод дахин салбарласан бөгөөд 4.0-CURRENT болон 3.X-STABLE салбарууд гарахад хүргэжээ. 3.X-STABLE-ээс 3.1 1999 оны 2 сарын 15-нд гарсан, 3.2 1999 оны 5 сарын 15-нд, 3.3 1999 оны 9 сарын 16-нд, 3.4 1999 оны 12 сарын 20-нд, тэгээд 3.5 2000 оны 6 сарын 24-нд гарсан бөгөөд хэдхэн хоногийн дараагаар Kerberos-т сүүлийн минутын аюулгүй байдлын засваруудыг оруулсан багахан хувилбар шинэчлэл 3.5.1-ийг гарахад хүргэсэн юм. Энэ нь 3.X салбар дахь сүүлийн хувилбар юм.

2000 оны 3 сарын 13-нд нэг салбар гарсан нь 4.X-STABLE салбар бий болсон явдал юм. Үүнээс хэд хэдэн хувилбарууд гарсан: 4.0-RELEASE 2000 оны 3 сард танилцуулагдсан бөгөөд сүүлийн 4.11-RELEASE 2005 оны 1 сард гарсан юм.

Удаан хүсэн хүлээсэн 5.0-RELEASE 2003 оны 1 сарын 19-нд зарлагдсан. Ойролцоогоор 3 жилийн ажлыг шингээсэн энэ хувилбар нь FreeBSD-г өргөжүүлсэн олон процессор, програмын урсгал дэмжлэгийн замд гаргаж UltraSPARC® ба `ia64` тавцангуудад зориулсан дэмжлэгийг танилцуулсан билээ. Энэ хувилбарын дараагаа 2003 оны 6 сард 5.1 хувилбар гарсан. -CURRENT салбарын сүүлийн 5.X хувилбар 2004 оны 2 сард танилцуулагдсан 5.2.1-RELEASE байв.

RELENG_5 салбар 2004 оны 8 сард үүссэн бөгөөд дараагаар нь 5-STABLE салбар хувилбаруулын эхлэлийг тэмдэглэсэн 5.3-RELEASE гарсан юм. Хамгийн сүүлийн 5.5-RELEASE 2006 оны 5 сард гарсан. RELENG_5 салбараас дахиж нэмэлт хувилбарууд гарахгүй.

2005 оны 7 сард энэ удаа RELENG_6-д зориулж мод дахин салбарлажээ. 6.X салбарын анхны хувилбар болох 6.0-RELEASE 2005 оны 11 сард гарчээ. Хамгийн сүүлийн 6.4-RELEASE 2008 оны 11 сард гарсан. Энэ салбар нь Alpha-г дэмжих хамгийн сүүлийн салбар юм.

RELENG_7 салбар 2007 оны 10 сард үүссэн. Энэ салбарын эхний хувилбар 7.0-RELEASE байсан бөгөөд 2008 оны 2 сард гарчээ. Хамгийн сүүлийн 7.4-RELEASE 2011 оны 2 сард гарсан. RELENG_7 салбараас дахиж нэмэлт хувилбарууд гарахгүй.

2009 оны 8 сард энэ удаа RELENG_8-д зориулж мод дахин салбарлажээ. 8.X салбарын анхны хувилбар 8.0-RELEASE 2009 оны 11 сард гарсан. Хамгийн сүүлийн 11.2-RELEASE June 28, 2018 гарсан. RELENG_8 салбарын нэмэлт хувилбарууд гарах болно.

2011 оны 9 сард RELENG_9 салбар үүссэн. Энэ салбарын эхний хувилбар 9.0-RELEASE байсан бөгөөд 2012 оны 1 сард гарсан. Хамгийн сүүлийн 12.0-RELEASE нь December 11, 2018 гарсан. RELENG_9 салбарын нэмэлт хувилбарууд гарах болно.

Одоогоор урт хугацааны хөгжүүлэлтийн төслүүд 10.X-CURRENT (транк) салбарт үргэлжлэх

бөгөөд ажил урагшлах бүр CD-ROM дээрх (мэдээж сүлжээнд бас) 10.X-ийн SNAPshot хувилбарууд Хормын агшны сервэрээс үргэлжлэн гарсаар байх болно.

### 1.3.2. FreeBSD Төслийн Зорилгууд

FreeBSD Төслийн зорилгууд нь ямар нэг хязгаарлалттүйгээр дурын зорилгоор ашиглаж болох програм хангамжийг хангах явдал юм. Бидний олонхи нь кодонд (болон төсөлд) чухал хөрөнгө оруулалт хийцгээсэн бөгөөд одоо болон ирээдүйд багахан санхүүгийн нөхөн олговроос мэдээж татгалзахгүй, гэхдээ бид мэдээж үүнийг шаардахаар бэлтгэгдээгүй юм. Бидний анхны бөгөөд нэн тэргүүний "даалгавар" бол ирсэн дурын болон бүгдэд аль ч зорилгоор ашиглагдаж болох кодоор хангах бөгөөд код нь аль болох өргөн хэрэглэгдэж, аль болох өргөн үр ашгийг өгөх явдал юм. Энэ нь Чөлөөт Програм Хангамжийн хамгийн үндсэн зорилгуудын нэг бөгөөд бид үүнийг санаачлагатайгаар дэмжих ёстой гэдэгт би итгэж байна.

Бидний эх модонд байгаа GNU General Public License (GPL) эсвэл Library General Public License (LGPL) хүрээнд байдаг код нь арай илүү хязгаарлалттай бөгөөд ядаж л эсрэгээрээ байх биш харин ч хүчилсэн хандлагын талд байдаг. GPL програм хангамжийг арилжааны зорилгоор ашиглахад гарах нэмэлт төвөгтэй асуудлуудаас болоод бид ингэж хийхэд боломжийн сонголт байгаа нөхцөлд арай зөөлөн BSD Зохиогчийн Эрхийн доор програм хангамжийг ирүүлэхийг илүүд үздэг.

### 1.3.3. FreeBSD Хөгжүүлэх Загвар

FreeBSD-ийн хөгжүүлэлт нь бидний хувь нэмэр оруулагчдын жагсаалтаас харахад дэлхий даяар хэдэн зуун хүмүүсийн оруулсан хувь нэмэр дээр бүтээгдсэн, их нээлттэй, уян хатан процесс юм. FreeBSD-ийн хөгжүүлэлтийн дэд бүтэц нь эдгээр хэдэн зуун хөгжүүлэгчдийг Интернэтээр хамтран ажиллах боломжийг нээж өгдөг. Бид шинэ хөгжүүлэгчид, болон санаануудыг тогтмол хайж байдаг бөгөөд төсөлтэй илүү ойртохыг сонирхсон хэн ч гэсэн FreeBSD техникийн хэлэлцүүлгүүдийн захидлын жагсаалт хаягаар бидэнд хандаарай. Мөн бусад FreeBSD хэрэглэгчдэд гол гол ажлуудын талаар мэдээлэх FreeBSD зарлалын захидлын жагсаалт бас байгаа болно.

Чөлөөтэй болон нягт хамтын ажиллагаан доор ажилладгаас үл хамааран FreeBSD төсөл болон түүний хөгжүүлэлтийн процессийн талаар ашигтай зүйлсийн талаар мэдэхийг хүсвэл:

**SVN репозитор**

FreeBSD-ийн гол эх мод нь FreeBSD-тэй цуг багцалсан чөлөөтэй байдаг эх кодыг хянах багаж болох CVS-ээр (Зэрэгцээ Хувилбаруудын Систем) хэдэн жилийн турш тэтгэгдэж байсан. 2008 оны 6 сард Төсөл SVN (Subversion) рүү шилжин хэрэглэхээр болсон. Эх модны хурдацтай өсөлт болон хадгалаастай байсан түүхийн хуудаснуудын хэмжээнээс болоод CVS-ийн техникийн хязгаарлалтууд илэрхий болж энэ шилжилтийг зайлшгүй хэрэгцээтэй гэж үзсэн юм. Баримтжуулалтын төсөл ба Портын цуглуулгын репозиториуд бас CVS-с SVN рүү 2012 оны 5 ба 7 саруудад тус тус шилжсэн. FreeBSD `src/` репозиторыг авах талаар Өөрийн эх модоо хамгийн сүүлийн түвшинд аваачих хэсэг, FreeBSD-н портын цуглуулгыг хэрхэн авах талаар Портын цуглуулгыг ашиглах хэсэг рүү хандаж дэлгэрэнгүй мэдээллийг үзээрэй.

### Итгэмжлэн оруулагчдын жагсаалт

*Итгэмжлэн оруулагчид* нь Subversion мод уруу *бичих* эрхтэй бөгөөд FreeBSD-ийн эхэд өөрчлөлтүүд хийх зөвшөөрөлтэй хүмүүс юм ("итгэмжлэн оруулагч" гэсэн ойлголт нь репозитор уруу шинэ өөрчлөлтүүдийг хийдэг эхийг хянах `commit` тушаалаас гаралтай). Итгэмжлэн оруулагчдад илгээсэн зүйлээ хянуулахаар өгөх хамгийн шилдэг арга нь send-pr(1) тушаал юм. Хэрэв ямар нэг зүйл систем дээр гацвал FreeBSD итгэмжлэн оруулагчдын захидлын жагсаалт уруу та цахим захидал бас илгээн холбогдож болно.

### FreeBSD-ийн гол баг

FreeBSD-г компани гэж бодох юм бол *FreeBSD-ийн гол баг* нь захирлуудын зөвлөлтэй ижил утгатай юм. Гол багийн үндсэн үүрэг нь төсөл бүхэлдээ сайн байж, түүнийг зөв чиглэлээр явуулахыг хянаж байх явдал юм. Өөрийгөө бүрэн дайчилсан, хариуцлагатай хөгжүүлэгчдийг итгэмжлэн оруулагчдын бүлэгт урих нь гол багийн үүргүүдийн нэг бөгөөд зарим гишүүд шилжихэд гол багийн шинэ гишүүдийг шинээр авах нь бас нэг үүрэг нь юм. Одоогийн гол баг нь нэр дэвшигч итгэмжлэн оруулагчдаас 2012 оны 7 сард сонгогдсон. Сонгууль 2 жилд нэг удаа явагддаг.

Гол багийн зарим гишүүд тусгай хариуцлагатай бөгөөд системийн ихээхэн хэсэг сурталчилсан хэмжээгээрээ ажиллаж байхыг хянаж бие сэтгэлээ зориулж байдаг. FreeBSD хөгжүүлэгчдийн жагсаалт болон тэдний аль хэсэгт хариуцлагатайг Хувь нэмэр оруулагчдын жагсаалтаас үзнэ үү.

Гол багийн ихэнх гишүүд FreeBSD-ийн хөгжүүлэлтэд сайн дураар оролцдог бөгөөд төслөөс санхүүгийн хувьд ашиг олдоггүй болохоор "commitment" гэдэг нь "баталгаатай дэмжлэг" гэсэн үг гэж ойлгож болохгүй юм. Дээрх "захирлуудын зөвлөл" гэсэн аналог нь тийм ч зөв биш бөгөөд эдгээр хүмүүс нь FreeBSD-ийн тусын тулд өөрсдийнхөө илүү шийдлийн эсрэг амьдралаа орхисон хүмүүс гэвэл магадгүй илүү тохирох биз ээ!

### Гаднын хувь нэмэр оруулагчид

Эцэст нь, гэхдээ мэдээж хамгийн сүүлийнх биш, хамгийн том бүлэг хөгжүүлэгчид нь санал сэтгэгдэл болон алдааны засваруудыг бидэнд бараг л тогтмол илгээдэг хэрэглэгчид юм. FreeBSD-ийн илүү төвлөрсөн бус хөгжүүлэлттэй холбоотой байх үндсэн арга нь тийм зүйлсийн талаар хэлэлцдэг FreeBSD техникийн хэлэлцүүлгүүдийн захидлын жагсаалт-д бүртгүүлэх явдал юм. FreeBSD-ийн төрөл бүрийн захидлын жагсаалтын талаар дэлгэрэнгүйг Интернэт дэх эх үүсвэрүүд-ээс үзнэ үү.

FreeBSD-ийн Хувь нэмэр оруулагчдын Жагсаалт нь урт бөгөөд өсөн нэмэгдэж байгаагийн нэг, тийм болохоор яагаад өнөөдөр FreeBSD-д хувь нэмэр оруулж энэ жагсаалтад нэгдэж болохгүй гэж?

Код ирүүлэх нь төсөлд хувь нэмэр оруулах ганц арга биш юм; Хийх шаардлагатай байгаа зүйлсүүдийн бүрэн жагсаалтын талаар FreeBSD Төслийн вэб хуудсын хаягт хандаж үзнэ үү.

Дүгнэж хэлэхэд бидний хөгжүүлэлтийн загвар нь нэг нь нөгөөдөө багтсан, чөлөөтэй тойргууд маягаар зохион байгуулагдсан загвар юм. Төвлөрсөн загвар нь ирээдүйтэй хувь

нэмэр оруулагчдыг цааш түлхэлгүйгээр нэг төвийн кодын суурийг хялбар хянах боломжийг олгож FreeBSD-ийн *хэрэглэгчдэд* эвтэй байхаар зориулагдан дизайн хийгдсэн юм. Бидний хүсэл бол хэрэглэгчид амархан суулгаж ашиглаж болдог ихээхэн хэмжээний уялдаа холбоотой хэрэглээний програмуудтай тогтвортой үйлдлийн системийг бий болгох явдал юм - энэнд хүрэхэд энэ загвар нь маш сайн тохирон ажиллаж байна.

Төслийн амжилттай байсаар байгаад хүргэж байгаа одоогийн хүмүүсийн адил бие сэтгэлийнхээ зарим ч гэсэн хэсгийг зориулахыг FreeBSD хөгжүүлэгч болон бидэнтэй нэгдэж байгаа хүмүүсээс хүсэж байгаа бидний цорын ганц хүсэлт юм!

### 1.3.4. Одоогийн FreeBSD хувилбарууд

FreeBSD нь чөлөөтэй байдаг, Интел i386™, i486™, Pentium®, Pentium® Pro, Celeron®, Pentium® II, Pentium® III, Pentium® 4 (эсвэл нийцтэй), Xeon™, болон Sun UltraSPARC®-д зориулагдсан бүрэн эх 4.4BSD-Lite дээр тулгуурласан хувилбар юм. Энэ нь үндсэндээ NetBSD, OpenBSD, 386BSD, болон Чөлөөт Програм Хангамжийн Сангийн зарим өргөжүүлэлт хийгдсэн Берклигийн К.И.С.-ийн CSRG бүлгээс гаргасан програм хангамжууд дээр үндэслэсэн.

1994 оны сүүлд гарсан FreeBSD 2.0 хувилбараас хойш FreeBSD-ийн хурдан ажиллагаа, боломжууд болон тогтвортой байдал мэдэгдэхүйц сайжирсан. Хамгийн том өөрчлөлт нь нийлсэн VM/файл буферийн кэш бүхий засварласан виртуал санах ойн систем бөгөөд энэ нь ажиллагааг хурдасгаад зогсохгүй FreeBSD-ийн санах ойн мөрийг багасгаж 5 MB тохиргоог илүү боломжийн хамгийн бага хэмжээнд хүргэсэн. Бусад өргөтгөлүүдийг дурдвал гүйцэд NIS клиент ба серверийн дэмжлэг, шилжилтийн TCP дэмжлэг, шаардлагаар залгах PPP, цогц DHCP дэмжлэг, сайжруулсан SCSI дэд систем, ISDN дэмжлэг, ATM-ийн дэмжлэг, FDDI, Хурдан болон Гигабит Ethernet (1000 Mbit) картууд, сүүлийн үеийн Адаптек хянагчууд болон олон мянган алдаануудын засварууд зэрэг юм.

Үндсэн түгээлтүүдээс гадна FreeBSD нь мянга мянган байнга хайгддаг програмуудтай хөрвүүлсэн програм хангамжийн цуглуулгатай байдаг. Энэ авлагыг хэвлэж байх үед 36000 гаруй портууд байсан! Портуудын жагсаалтад http (WWW) серверүүдээс тоглоомууд, хэлнүүд, засварлагчид, зэрэг бараг л бүх төрлийн програмууд байдаг. Портуудын Цуглуулга бүхэлдээ ойролцоогоор 3 GB хэмжээний хадгалалт шаарддаг бөгөөд бүх портууд өөрсдийн жинхэнэ эхийн хувьд "дельта" болж илэрхийлэгддэг. Энэ нь бидэнд портуудыг шинэчлэхэд хялбар болгож хуучин 1.0 Портуудын Цуглуулгын шаарддаг байсан дискний зайн шаардлагыг ихээхэн багасгаж өгдөг. Портыг хөрвүүлэхийн тулд та суулгахыг хүсэж байгаа програмын сан уруу шилжиж `make install` хэмээн бичихэд систем цааш үлдсэнийг хийх болно. Таны бүтээх порт болгоны жинхэнэ эх бүрэн түгээлт динамикаар CD-ROM эсвэл локал FTP хаягаас татагдах бөгөөд танд зөвхөн хүссэн портоо бүтээх хангалттай дискний зай л шаардлагатай. Порт бүр урьдчилан хөрвүүлсэн "багц" хэлбэрээр бас байх бөгөөд өөрийн портыг эхээс хөрвүүлэх хүсэлгүй хүмүүс хялбар тушаалаар (`pkg_add`) ийм портыг суулгаж болдог. Багцууд болон портуудын талаар дэлгэрэнгүй мэдээллийг Програм суулгах. Багцууд болон портууд хаягаар олж болно.

FreeBSD-н сүүлийн үеийн бүх хувилбарууд нь суулгагч (sysinstall(8) эсвэл bsdinstall(8)) ашиглан системийг эхэлж суулгахдаа нэмэлт баримтуудыг /usr/local/shared/doc/freebsd санд суулгах боломжоор хангагдсан байдаг. Баримтжуулалтыг бас дараа нь Баримтжуулалтын багцуудыг ашиглах нь-д тайлбарласан шигээр багц хэлбэрээр суулгаж болно. Локалаар суусан гарын авлагуудыг та HTML боломжтой хөтчүүдийг ашиглан дараах хаягаас үзэж

болно:

**FreeBSD Гарын авлага**

/usr/local/shared/doc/freebsd/handbook/index.html

**FreeBSD-ийн БХА (байнга хариулагддаг асуултууд FAQ)**

/usr/local/shared/doc/freebsd/faq/index.html

Мөн та мастер (болон бусдаас их шинэчлэгддэг) хуулбаруудыг http://www.FreeBSD.org/ хаягаас үзэж болно.

# Chapter 2. FreeBSD 8.X болон түүнээс өмнөх хувилбар суулгах нь

## 2.1. Ерөнхий агуулга

FreeBSD нь текст хэлбэртэй, хэрэглэхэд хялбар суулгалтын програмтай ирдэг. FreeBSD 9.0-RELEASE болон түүнээс хойшх хувилбарууд bsdinstall гэгддэг суулгалтын програм ашиглах бөгөөд 9.0-RELEASE-с өмнөх хувилбарууд sysinstall-г суулгалтад хэрэглэдэг. Энэ бүлэгт sysinstall ашиглан FreeBSD суулгах талаар тайлбарлах болно. bsdinstall-н хэрэглээ FreeBSD 9.X болон түүнээс хойшхи хувилбаруудыг суулгах нь хэсэгт гарах болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD суулгадаг дискнүүдийг хэрхэн үүсгэх.
- FreeBSD таны диск уруу хэрхэн ханддаг болон хуваадаг талаар.
- sysinstall-г хэрхэн эхлүүлэх.
- sysinstall-аас танд тавигдах асуултууд, тэд ямар учиртай болох мөн хэрхэн хариулах тухай.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- Суулгах гэж буй FreeBSD-нхээ хувилбар, мөн түүний дэмжиж чаддаг төхөөрөмжүүдийн жагсаалтыг хянаж тэр дунд таны төхөөрөмжүүд байгаа эсэхийг магадлах.

Ер нь, энэ суулгах заавар нь i386™ ("PC төрлийн") архитектурт зориулж бичсэн. Шаардлагатай тохиолдолд бусад тусгай тавцангуудын хувьд зориулагдсан зааврууд оруулах болно. Хэдийгээр энэ гарын авлагыг аль болох шинэчилж байгаа боловч та суулгагч болон энд дурдсан хоёрын хооронд бага зэргийн өөрчлөлтүүд байгааг анзаарч болох юм. Тийм болохоор энэ гарын авлагыг үгчилсэн заавар биш ерөнхий суулгах заавар болгож хэрэглэхийг танд зөвлөж байна.

## 2.2. Системд тавигдах төхөөрөмжийн шаардлага

### 2.2.1. Хамгийн бага тохиргоо

FreeBSD-г суулгахад шаардагдах хамгийн бага тохиргоо нь FreeBSD-н хувилбар болон төхөөрөмжийн төрлөөс их хамаарна.

Эдгээр мэдээллийн ерөнхий дүгнэсэн мэдээллийг энэ хэсэгт та унших болно. FreeBSD-г суулгах аргаасаа хамаараад танд уян диск юм уу эсвэл CDROM төхөөрөмж, зарим тохиолдолд сүлжээний адаптер хэрэг болох болно. Энэ тухай Эхлэн ачаалах төхөөрөмжийг бэлдэх хэсэгт дурьдсан буй.

#### 2.2.1.1. FreeBSD/i386 болон FreeBSD/pc98

FreeBSD/i386 болон FreeBSD/pc98 хоёр хоёулаа 486 юм уу эсвэл түүнээс дээш илүү төрлийн процессор шаарддаг бөгөөд хамгийн багадаа 24 MB RAM буюу шуурхай санах ой хэрэглэдэг. Танд хамгийн бага хэмжээгээр суулгахын тулд ядаж 150 MB дискний сул хэмжээ хэрэгтэй.

Хуучны тохиргоонд бол, ихэнх тохиолдолд, их хэмжээний санах ой ба дискний хэмжээ нь хурдан процессороос илүү хэрэгтэй байдаг.

#### 2.2.1.2. FreeBSD/amd64

FreeBSD/amd64-г ажиллуулах боломжтой хоёр ангиллын процессор байдаг. Эхнийх нь AMD Athlon™64, AMD Athlon™64-FX, AMD Opteron™ эсвэл түүнээс дээш төрлийн процессоруудыг агуулсан AMD64 процессорууд юм.

FreeBSD/amd64-ийг ашигладаг хоёр дахь ангиллын процессоруудад Intel® EM64T архитектурыг ашигладаг процессорууд ордог. Эдгээр процессоруудын жишээнд Intel® Core™ 2 Duo, Quad, Extreme процессорын гэр бүлийнхэн ба Intel® Xeon™ 3000, 5000, болон 7000 дарааллын процессоруудыг дурдаж болно.

Хэрэв таны машин nVidia nForce3 Pro-150 дээр үндэслэсэн бол та BIOS-н тохируулга дээр IO APIC конголтыг хорих *ёстой*. Хэрэв танд ингэх конголт байхгүй бол ACPI-г оронд нь хорих хэрэгтэй. Pro-150 төрлийн бичил схемд алдаа байдаг бөгөөд одоогоор тэр алдаа засагдаагүй байгаа билээ.

#### 2.2.1.3. FreeBSD/sparc64

FreeBSD/sparc64-г суулгахын тулд түүний дэмждэг төхөөрөмжүүдийг нь хангасан байх ёстой (Дэмждэг төхөөрөмжүүд хэсэгт харна уу).

Танд FreeBSD/sparc64-д зориулсан диск тусад нь байх хэрэгтэй. Одоогоор диск дээр өөр төрлийн системүүдийг давхар суулгах боломж үгүй.

### 2.2.2. Дэмждэг төхөөрөмжүүд

Дэмждэг төхөөрөмжүүдийг FreeBSD-н хувилбар бүрийн Hardware Notes буюу төхөөрөмжийн мэдээлэл хэсэгт жагсаасан байдаг. Энэ мэдээлэл нь голдуу HARDWARE.TXT нэрээр суулгацын CDROM юм уу FTP-н хамгийн дээд сан дотор эсвэл sysinstall програмын documentation буюу баримт цэсэнд байрласан байдаг билээ. Энэ нь тухайн төрлийн архитектур бүрт ямар ямар төхөөрөмжүүд танигдаж болохыг FreeBSD-н хувилбар бүрт зориулж жагсаасан байдаг. Төрөл бүрийн хувилбар болон архитектурт зориулсан жагсаалтын бас нэг хуулбарыг FreeBSD-н вэб хуудасны Release Information буюу хувилбарын мэдээлэл хуудаснаас олж болно.

## 2.3. Суулгацын өмнө

### 2.3.1. Өөрийнхөө компьютерийг судална

Та FreeBSD-г суулгахын өмнө өөрийнхөө компьютерийн бүрдэл хэсгүүдийг судлах хэрэгтэй. FreeBSD суулгах явцдаа таны компьютерт буй бүрдлүүдийг (хатуу диск, сүлжээний карт,

CDROM хөтлөгч гэх мэт) тэдгээрийн загвар болон үйлдвэрлэгчийнх нь дугаартай нь харуулдаг. Мөн FreeBSD нь тэдгээр төхөөрөмжүүдийг IRQ болон IO порт зэргүүдийг автоматаар зөв тохируулахыг оролддог. Компьютерийн бүрдлийг үйлдвэрлэдэгчдийн түмэн төрлөөс хамаарч зөв тохируулах энэ автомат үйлдэл нь заримдаа тийм амжилттай болж чаддаггүй тул магадгүй та FreeBSD-н тодорхойлсон тохируулгыг өөрчлөх хэрэг гарч болзошгүй.

Хэрэв та Windows® эсвэл Линукс үйлдлийн систем суулгасан байгаа бол тухайн төхөөрөмжүүд ямар тохируулгаар суугдсан байгааг харах нь зүйтэй. Хэрвээ өргөтгөл картын дугаар болон нэрийг зөв таньсан эсэхээ мэдэхгүй бол уг карт дээр буй үйлдвэрлэсэн бичгийг нь харах хэрэгтэй. Байнгын хэрэглэдэг IRQ дугаарууд нь 3, 5, мөн 7 бөгөөд ихэнх хэрэглэгддэг IO портын хаягууд нь голдуу 0x330 гэх мэт арван зургаат тооллын систем дээр бичсэн дугаарууд байдаг.

FreeBSD-г суулгахаасаа өмнө эдгээр дугаарыудыг тэмдэглэж авахыг зөвлөж байна. Та дараах маягийн хүснэгт хөтлөх хэрэгтэй:

*Хүснэгт 1. Төхөөрөмжийн бүртгэлийн жишээ*

| **Төхөөрөмжийн нэр** | **IRQ** | **IO порт(ууд)** | **Тэмдэглэгээ** |
|---|---|---|---|
| Эхний хатуу диск | мэдэхгүй | мэдэхгүй | 40 ГБ, Seagate үйлдвэрлэсэн, эхний мастер IDE |
| CDROM | мэдэхгүй | мэдэхгүй | Эхний хоёр дахь IDE |
| Хоёр дахь хатуу диск | мэдэхгүй | мэдэхгүй | 20 ГБ, IBM үйлдвэрлэсэн, хоёр дахь мастер IDE |
| Эхний IDE хянагч | 14 | 0x1f0 | |
| Сүлжээний карт | мэдэхгүй | мэдэхгүй | Intel® 10/100 |
| Модем | мэдэхгүй | мэдэхгүй | 3Com® 56K факс модем, COM1 дээр зоогдсон |
| … | … | … | … |

### 2.3.2. Өөрийнхөө өгөгдлийг нөөцөлж авах

Хэрэв таны FreeBSD суулгах гэж байгаа компьютерт чухал мэдээлэл байвал, уг мэдээллээ нөөцөлж хадгалж авах нь зүйтэй бөгөөд уг нөөцөлсөн мэдээллээ зөв хадгалснаа шалгах хэрэгтэй. FreeBSD-н суулгах явц нь диск уруу бичиж эхлэхээсээ өмнө танаас лавлаж асуудаг бөгөөд хэрэв нэг бичигдээд эхэлбэл буцаах арга байхгүй.

### 2.3.3. FreeBSD-г хаана суулгахаа шийдэх

Хэрэв та FreeBSD-д бүх дискээ хэрэглүүлнэ гэж бодож байвал энэ хэсэгт анхаарлаа хандуулалгүй цааш нь унших хэрэгтэй.

Харин, хэрэв та FreeBSD-г өөр үйлдлийн системтэй хамт хэрэглэнэ гэж бодож байвал диск

дээр өгөгдөл хэрхэн байрладаг мөн хэрхэн ажилладаг талаар үндсэн мэдлэгтэй байх хэрэгтэй.

#### 2.3.3.1. FreeBSD/i386 дээрх дискний өгөгдөл байрлуулалт

Компьютерийн диск нь хэд хэдэн хэсэгт хэсэгчилж болдог. Эдгээр хуваагдсан хэсгүүдийг нь *partitions буюу хуваалтууд* гэж нэрлэдэг. FreeBSD нь дотроо бас хуваалтуудтай болохоор энэ нэрлэлт нь ойлгомжгүй болж болох юм. Тийм болохоор эдгээр дискний хэсэгчлэлүүдийг дискний зүсмэлүүд буюу ердөө л зүсмэлүүд гэж FreeBSD үздэг. Жишээ нь PC-ийн дискний хуваалтууд дээр ажилладаг FreeBSD-ийн `fdisk` хэрэгсэл хуваалтуудын оронд зүсмэлүүдийг хэрэглэдэг. Анхнаасаа компьютерийн нэг дискийг зөвхөн дөрөв хуваж болохоор зохиосон байна. Эдгээр хуваалтуудыг *primary partitions буюу анхдагч хуваалтууд* гэж нэрлэдэг. Энэ хязгаарлалтыг тойрон гарч дөрвөн хуваалтаас илүүг зөвшөөрөхийн тулд шинэ хуваалтын арга үүссэн бөгөөд түүнийг *extended partition буюу өргөтгөсөн хуваалт* гэж нэрлэжээ. Диск зөвхөн ганцхан өргөтгөсөн хуваалт агуулж болно. Өргөтгөсөн хуваалт дотор хичнээн бол хичнээн *logical partitions буюу логик хуваалтууд* агуулж болдог байна.

Хуваалт болгон *partition ID буюу хуваалтын ID дугаар* агуулж байдаг бөгөөд энэ дугаар нь тухайн хуваалтын төрлийг илэрхийлж байдаг. FreeBSD төрлийн хуваалтууд нь `165` гэсэн ID дугаартай байдаг.

Үйлдлийн систем болгон дискний хуваалтуудыг таних өөр өөрийн арга хэрэглэдэг. Жишээлбэл MS-DOS® буюу түүнтэй ижил төрлийн Windows® системүүд нь дискний хуваалтыг танихын тулд *үсэгчлэн дугаарласан диск* аргыг хэрэглэдэг бөгөөд C: үсгээр эхлэж тэмдэглэдэг.

FreeBSD нь primary partition буюу дискний анхдагч хуваалт дээр суугдах ёстой. FreeBSD таны үүсгэсэн файлуудыг, бас өөрийнхөө файлуудыг энэ хуваалт дээр хадгалдаг. Хэрэв танд олон диск байвал мөн та тэдгээр дээр эсвэл тэдний зарим дээр FreeBSD төрлийн хуваалт үүсгэж болно. FreeBSD суулгах үедээ дискний нэг хуваалтыг бэлэн байлгах хэрэгтэй. Энэ хуваалт нь таны урьдчилан бэлдсэн хоосон хуваалт юм уу эсвэл онц шаардлагагүй өгөгдөл хадгалсан хуваалт байсан ч болно.

Хэрэв та өөрийн бүх диск дэх бүх хуваалтыг хэрэглэж байсан бол тэдгээрийн нэгийг нь FreeBSD-д зориулж ямар нэгэн үйлдлийн системд байдаг хэрэгслийг ашиглан хоосон болгох хэрэгтэй (жишээ нь, MS-DOS® or Windows® дээр байдаг `fdisk` програм).

Хэрэв танд илүүчилж болохоор хуваалт байвал тэр хэсгийг бас хэрэглэж болно. Гэхдээ та өмнө нь байж байсан хуваалтын хэмжээг ихэсгэж юм уу багасгаж хэрэглэж хэрэгтэй болно.

FreeBSD суугдаж чадах хамгийн бага хэмжээ бол 100 MB билээ. Гэхдээ энэ хэмжээ бол өөрийнхөө файлуудыг хадгалахад бараг хүрэлцэхээргүй *хамгийн* бага хэмжээ юм. Арай боломжийн бага хэмжээ бол график орчныг оруулалгүйгээр 250 MB хэмжээ юм. Хэрэв график орчинг оруулбал 350 MB болно. Хэрэв та гуравдагч програм зохиогчдын програмыг суулгаж хэрэглэнэ гэж бодож байвал мэдээж түүнээс илүү хэмжээ хэрэгтэй.

Та PartitionMagic® гэдэг үнэтэй програмыг, эсвэл GParted зэрэг үнэгүй програмыг FreeBSD-д зориулж дискэндээ зай гаргахад хэрэглэж болох юм. PartitionMagic® ба GParted нар NTFS төрлийн хуваалт дээр ажиллаж чаддаг. GParted нь SystemRescueCD зэрэг хэд хэдэн Live CD тархацуудад байдаг.

Microsoft® Vista хуваалтуудын хэмжээг дахин өөрчлөхөд асуудалтай байгаа талаар мэдээлэл бий. Ийм үйлдэл хийхийг оролдох үедээ Vista-ийн суулгалтын CDROM-той байхыг зөвлөж байна. Мөн иймэрхүү дисктэй ажиллах ажлуудыг хийж гүйцэтгэхээсээ өмнө нөөцлөлтийг хийх нь маш чухал юм.

Эдгээр хэрэгслийг буруу ашиглавал дискэн дээр байгаа мэдээлэл тань устах болно. Тэдгээрийг хэрэглэхээсээ өмнө өөрийнхөө өгөгдлийг нөөцөлж хадгалж авах хэрэгтэй.

*Жишээ 1. Байгаа дискний хуваалтыг өөрчлөлгүйгээр хэрэглэх нь*

Жишээлбэл, Windows® систем суулгасан 4 ГБ диск танд байгаа гэж бодъё. Мөн та тэр дискийг 2 ГБ хэмжээгээр C: ба D: гэж хоёр хуваасан байгаа. Танд C: дээр 1 ГБ, D: дээр 0.5 ГБ өгөгдөл хадгалсан байгаа гэж үзье.

Энэ бол үсэгчлэн тэмдэглэсэн хоёр хуваалт танд байна гэсэн үг. Та D: дээр байгаа бүх өгөгдлийг C: руу хуулж чөлөөлөөд түүн дээр FreeBSD суулгаж болно.

*Жишээ 2. Байгаа хуваалтын хэмжээг сунгаж өөрчлөх*

Танд Windows® суулгасан 4 ГБ диск байна гэж үзье. Windows® суулгах үедээ та зөвхөн C: гэж нэрлэсэн 4 ГБ хэмжээтэй ганцхан бүхэл хуваалт үүсгэжээ. Та одоогоор уг хэмжээний 1.5 ГБ-ийг ашигласан гэж бодоцгооё. Тэгээд та уг дискний 2 ГБ хэмжээ дээр FreeBSD суулгахыг хүссэн гэж авч үзье.

FreeBSD-г суулгахын тулд доор дурдсанаас аль нэгийг хийх болно:

1. Windows® дээр байсан өгөгдлөө нөөцөлж хадгалж аваад Windows®-г дахин суулгаж гэхдээ түүндээ 2 ГБ хэмжээ үүсгэж хэрэглэх.
2. PartitionMagic® мэтийн програм ашиглаж Windows®-н дискний хэмжээг дээрх хэмжээнд тохируулж өөрчлөх.

## 2.3.4. Сүлжээний нарийвчилсан тохируулгаа мэдэх

Хэрэв та FreeBSD-н суулгацыг интернэт холболттой хийнэ гэж бодож байвал (жишээ нь, суулгацыг FTP эсвэл NFS серверээс татан суулгаж байвал), та сүлжээнийхээ тохируулгыг мэдэж байх хэрэгтэй. Ийм мэдээллийг суулгах үед танаас лавлаж асуугаад, тохируулж интернэтэд холбогддог.

### 2.3.4.1. Дотоод сүлжээ эсвэл Кабель/DSL Модемоор холбогдох

Хэрэв та дотоод сүлжээнд холбогдсон байгаа юм уу эсвэл кабел, DSL-ээр холбогдохоор бол дараах тохируулах мэдээллийг мэдсэн байх хэрэгтэй:

1. IP хаяг
2. Анхны gateway буюу гарцын IP хаяг
3. Hostname буюу серверийн нэр

4. DNS сервер IP хаяг
5. Subnet Mask буюу дэд сүлжээний баг (тусгаарлан ангилах дугаар)

Хэрэв та эдгээр мэдээллийг мэдэхгүй байгаа бол уг сүлжээ хариуцагч юм уу интернэт уруу холбогдох байгууллага уруу хандах хэрэгтэй. Гэтэл тэд танд, эдгээр мэдээлэл нь *DHCP* ашиглаж автоматаар тохируулагдана гэж хэлж магадгүй. Хэрэв тийм бол та эдгээр мэдээллийг заавал мэдсэн байх албагүй бөгөөд энэ хэсгийг зүгээр санаад авахад илүүдэхгүй.

#### 2.3.4.2. Модем хэрэглэж холбогдох

Хэрэв та интернэтээр хангагч байгууллага уруу ердийн модем ашиглан утсаар холбогддог бол, та мөн FreeBSD-г интернэтээр суулгаж болох бөгөөд жаахан удах л байх даа.

Дараах зүйлсийг мэдэж байх шаардлагатай:

1. ISP буюу интернэтийн үйлчилгээ үзүүлэгчийн холбогдох утасны дугаар
2. Модемийн хэрэглэх COM: портын дугаар
3. Интернэтийн үйлчилгээ үзүүлэгчид бүртгүүлсэн хэрэглэгчийн нэр болон нууц үг

### 2.3.5. FreeBSD-н алдааны бүртгэл

FreeBSD төсөл нь гаргаж буй хувилбар болгоноо алдаагүй баттай байлгахыг чармайж байдаг боловч зарим тохиолдолд жижиг алдаанууд гарах тохиолдол үүсдэг. Маш ховор үед ийм алдаа суулгах үед гардаг. Эдгээр алдааг илрүүлж засаад, энэ тухайгаа FreeBSD алдааны бүртгэл хуудсан дээр тэмдэглэж бичдэг. Та суулгаж байх үед ийм хүндрэлтэй тулгарахгүйн тулд энэ хуудас уруу орж шалгах хэрэгтэй.

Хувилбар бүрд гарсан алдаануудыг бүх хувилбартай нь жагсаасан бүртгэлийг FreeBSD-н вэб хуудасны хувилбарын мэдээлэл хэсэгт харж болно.

### 2.3.6. FreeBSD-н суулгац файлуудыг бэлдэх

FreeBSD-г суулгах явц нь дараах байршилд буй файлаас гүйцэтгэгдэж болно:

*Дотоод төхөөрөмжөөс*

- CDROM эсвэл DVD
- USB санах ойн зөөгч
- Уг компьютерт буй MS-DOS® хэсгээс
- SCSI эсвэл QIC бичлэгээс
- Уян дискнээс

*Сүлжээ*

- FTP хаягнаас. Хэрэв шаардлагатай бол галт хана эсвэл HTTP проксигоор дамжина
- NFS сервер
- Зориулалтын параллел юм уу цуваа холболт

Хэрэв та FreeBSD -н суулгацыг CD эсвэл DVD хэлбэрээр авсан бол танд хэрэгтэй бүх зүйл бэлэн болох бөгөөд энэ хэсгийг алгасаад дараагийн хэсэг уруу шилжиж болно. (Эхлэн ачаалах төхөөрөмжийг бэлдэх).

Хэрэв та FreeBSD-н суулгац файлуудыг бэлдэж аваагүй бол Өөртөө зориулж тусгай суулгацын төхөөрөмж бэлдэх нь хэсэг уруу очиж дээрх байршлаас хэрхэн бэлдэж авах талаар тайлбарласныг уншина уу. Тэр хэсгийг уншиж дуусаад буцаж эндээс Эхлэн ачаалах төхөөрөмжийг бэлдэх хэсэг уруу орох хэрэгтэй.

### 2.3.7. Эхлэн ачаалах төхөөрөмжийг бэлдэх

FreeBSD суулгац нь таны компьютер ачаалах үед эхэлдэг- энэ нь өөр үйлдлийн системээс эхлүүлдэг програм биш. Таны компьютер ердийн үед хатуу дискэн дээр суугдсан үйлдлийн системээр эхлэж ачаалагддаг. Гэхдээ CDROM диск эсвэл USB дискнээс эхлэн ачаалагдагдахаар тохируулж бас болдог.

Хэрэв та FreeBSD-г CDROM эсвэл DVD дээр (худалдаж авсан юм уу эсвэл өөрөө бэлдэж авсан бол) бэлдэж авсан бөгөөд таны компьютер CDROM эсвэл DVD-ээс эхлэн ачаалагдаж болдог (ихэнх BIOS дээр "Boot Order буюу ачаалах дараалал" гэсэн эсвэл үүнтэй төсөөтэй сонголтоор тохируулагддаг) бол энэ хэсгийг уншилгүй алгасаж болно. FreeBSD-н CDROM болон DVD дээр байгаа файлууд нь нэмэлт зүйлс шаардалгүй шууд суугдах боломжтой.

Ачаалагдаж болдог санах ойн зөөгч үүсгэхийн тулд дараах алхмуудыг хийнэ:

1. Санах ойн зөөгчид зориулсан дүрс файлыг олж авах нь

   FreeBSD 8.*X* болон түүнээс өмнөх хувилбарын хувьд санах ойн зөөгчид зориулсан дүрс файлыг ISO-IMAGES/ сан дахь `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/arch/ISO-IMAGES/version/FreeBSD-version-RELEASE-arch-memstick.img` хаягаас татан авч болно. *arch* болон *version* үгсийг та өөрийн суулгахыг хүссэн архитектур болон хувилбарын дугаараар солиорой. Жишээ нь FreeBSD/i386 11.2-RELEASE хувилбарт зориулсан санах ойн зөөгчид зориулсан дүрс файлыг ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/ISO-IMAGES/11.2/FreeBSD-11.2-RELEASE-i386-memstick.img хаягаас авч болно.

   FreeBSD 9.0-RELEASE болон түүнээс хойшхи хувилбаруудын хувьд өөр сан ашиглагддаг. FreeBSD 9.0-RELEASE болон түүнээс хойшхи хувилбаруудын хувьд татаж авч суулгах талаар дэлгэрэнгүйг FreeBSD 9.X болон түүнээс хойшхи хувилбаруудыг суулгах нь хэсгээс үзнэ үү.

   Санах ойн зөөгчид зориулсан дүрс нь .img өргөтгөлтэй байна. ISO-IMAGES/ сан нь төрөл бүрийн дүрсээс тогтох бөгөөд таны хэрэглэх шаардлагатай нь таны суулгаж байгаа FreeBSD-ийн хувилбараас хамаарах бөгөөд зарим тохиолдолд тоног төхөөрөмжөөс бас хамаарна.

   Эхлүүлэхээсээ өмнө өөрийн USB зөөгч дээр байгаа өгөгдлийг

*нөөцлөөрэй*, учир нь энэ алхам өгөгдлийг *устгах* болно.

2. Дүрс файлыг санах ойн зөөгч рүү бичих нь

**Procedure: FreeBSD ашиглан дүрсийг бичих**

Доорх жишээ дүрсийг хуулах төхөөрөмжийг /dev/da0 гэж үзэх болно. Зөв төхөөрөмж эсэхийг сайн нягтлаарай, эс тэгвээс өөрийн өгөгдлийг устгаж мэднэ шүү.

a. Дүрсийг dd(1) ашиглан бичих нь

.img нь санах ойн зөөгчид хуулдаг ердийн файл шиг *биш* юм. Энэ нь дискний бүрэн агуулга бүхий дүрс юм. Энэ нь та нэг дискнээс нөгөө диск рүү зүгээр л файлууд хуулдаг шиг хуулж *болохгүй* гэсэн үг юм. Түүний оронд та дүрсийг диск рүү шууд бичихийн тулд dd(1) тушаалыг ашиглах ёстой юм:

```
# dd if=FreeBSD-11.2-RELEASE-i386-memstick.img of=/dev/da0 bs=64k
```

Хэрэв `Operation not permitted` гэсэн алдаа гарах юм бол төхөөрөмж ашиглагдаагүй, холбогдоогүй эсвэл зарим нэг зориулалтын програмын тусламжтай автоматаар холбогдоогүй эсэхийг шалгаарай. Дараа нь дахин оролдоорой.

*Procedure: Windows® ашиглан дүрсийг бичих*

Гаралтын төхөөрөмж зөв эсэхийг нягтлаарай, эс тэгвээс та өөрийн өгөгдлийг дарж бичин устгаж мэднэ шүү.

a. Image Writer for Windows програмыг олж авах нь

Image Writer for Windows нь санах ойн зөөгч рүү дүрс файлыг зөв бичиж чаддаг чөлөөтэй түгээгддэг програм юм. https://launchpad.net/win32-image-writer/ хаягаас татаж аваад хавтас руу задлаарай.

b. Image Writer ашиглан дүрсийг бичих нь

Програмыг эхлүүлэхийн тулд Win32DiskImager дүрсэн дээр хулганаа хоёр товшоорой. `Device` дотор харагдаж байгаа хөтөч санах ойн зөөгч байгаа хөтөч эсэхийг шалгаарай. Хавтсан дээр дарж санах ойн зөөгч рүү хуулах дүрсийг сонгоно. **[ Save ]** дарж дүрс файлын нэрийг зөвшөөрнө. Бүгд зөв болохыг шалгаад санах ойн зөөгч дээрх аливаа хавтас өөр цонхон дээр нээгдээгүй эсэхийг шалгаарай. Төгсгөлд нь дүрс файлыг зөөгч рүү бичихийн тулд **[ Write ]** товчийг дараарай.

FreeBSD/pc98-д зориулж ачаалагдаж болдог уян диск бэлдэхийн тулд дараах алхмыг гүйцэтгэнэ:

1. Ачаалагддаг уян дискний Image буюу дүрс файлыг бэлдэх

   FreeBSD/pc98-д зориулсан ачаалагдах дискийг `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/pc98/version-RELEASE/floppies/` сангаас татаж авч болно. *version*-г суулгах хувилбараараа солиорой.

   Уян дискний images буюу дүрс файл нь .flp гэсэн өргөтгөлтэй байдаг. floppies/ сан нь янз бүрийн дүрс файл агуулж байдаг. boot.flp болон суулгалтын төрлөөс хамааран `kern.small*` эсвэл `kern*` зэрэг файлуудыг татаж аваарай.

   > Эдгээр дүрс файлыг татаж авах FTP програм нь *binary mode* буюу хоёртын файлын хэлбэр горимоор татаж авах ёстой. Зарим вэб хөтөч програмууд нь *текст* (эсвэл *ASCII*) горим хэрэглэдэг бөгөөд ийм үед таны уян диск анхлан ачаалагдаж чадахгүй.

2. Уян диск бэлдэх

   Татаж авсан дүрс файл болгонд нэг уян диск бэлдэх ёстой. Уг дискнүүд нь ямар нэгэн алдаагүй байх шаардлагатай. Шалгах хамгийн амар арга бол шууд форматлах буюу цэвэрлэх хэрэгтэй. Урьдчилан цэвэрлэсэн дискэнд итгэх хэрэггүй. Windows® -н цэвэрлэдэг хэрэгсэл нь дискэн дээр байгаа эвдэрсэн хэсгийг мэдээлдэггүй бөгөөд тэдгээрийг зүгээр "bad буюу муу" гэж тэмдэглээд өнгөрдөг. Шинэ диск хэрэглэн суулгах үйлдэл хийхийг танд зөвлөж байна.

   > Хэрэв таны FreeBSD-г суулгах явц гацах, эвдрэх, ямар нэг гаж нөлөө үзүүлбэл та хамгийн түрүүнд уян дискээ хардах хэрэгтэй. Шинэ дискэнд дүрс файлаа бичээд дахин оролдоорой.

3. Дүрс файлыг уян диск уруу бичих

   .flp файлууд нь диск уруу зүгээр хуулдаг *энгийн* файл биш юм. Тэд бол дискний бүхэл бүтцийг агуулсан дүрс файл. Тийм болохоор ийм файлыг диск уруу шууд хуулж *болохгүй*. Харин, дүрс файлыг диск уруу буулгах тусгай хэрэгсэл ашигладаг.

   Хэрэв та MS-DOS® / Windows® үйлдлийн систем дээр ажиллаж байгаа бол `fdimage` хэрэгсэл хэрэглэх хэрэгтэй.

   Хэрэв уян дискнүүд CDROM дээр байгаа бөгөөд таны CDROM E: гэж танигдсан бол та дараах тушаалыг өгөх хэрэгтэй:

   ```
   E:\> tools\fdimage floppies\boot.flp A:
   ```

   Энэ тушаалыг уян дискээ сэлгэж .flp файл болгонд гүйцэтгээд дараа нь дэс дараалан тэмдэглэх хэрэгтэй. .flp файлын байрлалаас хамааран тушаалаа тохируулж өгөх хэрэгтэй. Хэрэв танд CDROM байхгүй бол `fdimage` нь FreeBSD-н FTP багажнууд сангаас

хуулагдах боломжтой.

Хэрэв та уян дискийг UNIX® системээс бэлдэж байгаа бол( өөр FreeBSD системээс) та dd(1) тушаалыг ашиглан дүрс файлыг уян диск дээр буулгаж болно. FreeBSD дээр:

```
# dd if=boot.flp of=/dev/fd0
```

FreeBSD дээр /dev/fd0 гэхээр эхний уян дискний хөтлөгч уруу ханддаг ( A: төхөөрөмж). /dev/fd1 гэвэл B: төхөөрөмж гэх мэтчилэн үргэлжилдэг. Бусад UNIX® төрлийн систем дээр уян дискний төхөөрөмж нь өөр өөр нэртэй байж болох бөгөөд шаардлагатай бол тухайн системийн бичиг баримтаас лавлах хэрэгтэй.

Та одоо FreeBSD-н суулгацыг эхлүүлэхэд бэлэн боллоо.

## 2.4. Суулгацыг эхлүүлэх

Дараах мэдэгдлийг харах хүртэл суулгац програм нь таны диск(нүүд)эд ямар нэгэн өөрчлөлт хийдэггүй:

```
Last Chance: Are you SURE you want continue the installation?

If you're running this on a disk with data you wish to save then WE
STRONGLY ENCOURAGE YOU TO MAKE PROPER BACKUPS before proceeding!

We can take no responsibility for lost disk contents!
```

Үүнийг орчуулбал:

```
      Сүүлийн боломж: Та суулгацыг үргэлжлүүлэхдээ ИТГЭЛТЭЙ байна уу?

      Хэрэв та хэрэгтэй мэдээллээ хадгалсан дискнээс уг суулгацыг
      эхлүүлж байгаа бол, эхлүүлэхээсээ өмнө ӨӨРИЙНХӨӨ ӨГӨГДЛИЙГ
НАЙДВАРТАЙ ХАДГАЛЖ
      АВАХЫГ БИД ЗӨВЛӨЖ БАЙНА!

      Дискэнд буй өгөгдөлд учирсан эвдрэлд бид хариуцлага хүлээхгүй!
```

Суулгац програм нь энэ анхааруулга хүртэл дискэнд өөрчлөлт хийлгүйгээр цуцалж гарах боломжтой. Хэрэв та ямар нэгэн зүйл буруу тохируулсан юм уу өөрчлөх ёстой гэж бодож байвал, энэ мөчид та компьютераа унтраахад ямар ч эвдрэл үүсэхгүй.

### 2.4.1. Эхлэн ачаалалт

#### 2.4.1.1. i386™ системд эхлэн ачаалах

1. Компьютер унтарсан үеэс эхлэх хэрэгтэй.
2. Компьютерийг асаана. Эхлэх үед дэлгэц дээр системийн BIOS-н үндсэн тохируулга уруу ордог гарын товчлолыг харуулдаг. Энэ нь голдуу `F2`, `F10`, `Del`, эсвэл `Alt` + `S` гэсэн товчлолуудын нэг нь байдаг. Дэлгэцэнд юу гэж заасан байна, уг товлолыг дарж BIOS-н тохируулга уруу орох хэрэгтэй. Зарим тохиолдолд, эхлэх үед график зураг харуулдаг бөгөөд голдуу `Esc` товч дарснаар уг зургийг болиулж хүссэн текстээ дэлгэцэнд харах боломжтой.
3. Систем аль төхөөрөмжөөс эхлэж ачаалах вэ гэсэн тохируулгыг олох хэрэгтэй. Энэ нь голдуу "Boot Order буюу эхлэх дараалал" гэсэн хэсэгт байдаг бөгөөд ачаалж болох `Floppy`, `CDROM`, `First Hard Disk` гэсэн ачаалж болох төхөөрөмжийн жагсаалтыг агуулсан байдаг.

   Хэрэв та CDROM-с ачаална гэж бодсон бол уг жагсаалтнаас CDROM сонгох хэрэгтэй. Хэрэв та USB диск эсвэл уян дискнээс эхэлж ачаална гэж бодож байвал түүнийг сонгох хэрэгтэй. Аль нь зөв эсэхээ мэдэхгүй эргэлзэж байгаа бол уг компьютертай цуг ирдэг гарын авлагаас хараарай.

   Тохирсон өөрчлөлтөө хийж хадгалаад гарна. Компьютер ингэсний дараа шинээр дахин ачаалагдаж эхэлнэ.
4. Хэрэв та Эхлэн ачаалах төхөөрөмжийг бэлдэх дээр заасан шиг "ачаалагдах" USB зөөгч бэлдсэн бол, USB зөөгчөө компьютераа асаахаасаа өмнө залгаарай.

   Хэрэв та CDROM -с эхэлж ачаалж байгаа бол компьютераа асаангуутаа CDROM уруу дискээ хийж эхлүүлэх хэрэгтэй.

   FreeBSD/pc98 хувилбаруудад суулгалтын ачаалагдах боломжтой уян дискнүүд байдаг бөгөөд Эхлэн ачаалах төхөөрөмжийг бэлдэх хэсэгт тайлбарласан шиг бэлдэж болно. Эхний диск нь boot.flp байна. Энэ дискийг өөрийн уян дискний хөтчид хийгээд компьютераа ачаална.

   Хэрэв таны компьютер асаад, өмнө суусан байсан үйлдлийн системээс ердийнхөөрөө эхлэж байвал шалтгаан нь доор дурдсанаас аль нэг нь байж болно: .. Ачаалах явцаас өмнө нь амжиж дискээ оруулаагүй байх. Дискээ оруулаад компьютераа дахин шинээр ачаал. .. Өмнө хийсэн BIOS -ийн өөрчлөлт зөв хийгдээгүй байх. Уг өөрчлөх үйлдлийг дахин хийж зөв тохируулгыг хийх. .. Магадгүй таны бэлдсэн төхөөрөмжөөс эхлэж ачаалах үйлдлийг таны BIOS дэмжээгүй байж болно.
5. FreeBSD ачаалагдаж эхлэнэ. Хэрэв та CDROM -ноос эхлүүлсэн бол дараах зүйлтэй адилхан мэдэгдэл харах болно (хувилбарын хэсгийг оруулаагүй болно):

   ```
   Booting from CD-Rom...
   645MB medium detected
   CD Loader 1.2
   ```

```
Building the boot loader arguments
Looking up /BOOT/LOADER... Found
Relocating the loader and the BTX
Starting the BTX loader

BTX loader 1.00 BTX version is 1.02
Consoles: internal video/keyboard
BIOS CD is cd0
BIOS drive C: is disk0
BIOS drive D: is disk1
BIOS 636kB/261056kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1

Loading /boot/defaults/loader.conf
/boot/kernel/kernel text=0x64daa0 data=0xa4e80+0xa9e40 syms
=[0x4+0x6cac0+0x4+0x88e9d]
\
```

Хэрэв та уян дискнээс эхлүүлж байгаа бол дараах мэдэгдэлтэй ижил бичиглэлийг харна (хувилбарын хэсгийг оруулаагүй болно):

```
Booting from Floppy...
Uncompressing ... done

BTX loader 1.00  BTX version is 1.01
Console: internal video/keyboard
BIOS drive A: is disk0
BIOS drive C: is disk1
BIOS 639kB/261120kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1

Loading /boot/defaults/loader.conf
/kernel text=0x277391 data=0x3268c+0x332a8 |

Insert disk labelled "Kernel floppy 1" and press any key...
```

Уг зааврын дагуу boot.flp дискийг гаргаад kern1.flp дискийг оруулаад `Enter` товчийг дараарай. Эхний дискнээс эхэлж ачаалаад дараа нь шаардсан дискүүдийг нь дараалан оруулах хэрэгтэй.

6. CDROM, USB зөөгч эсвэл уян диск алинаас нь ч эхлүүлсэн бай гэсэн FreeBSD-ийн ачаалагч эхлүүлэгч цэсэнд тулж ирдэг:

*Зураг 1. FreeBSD-ийн ачаалагч эхлүүлэгч цэс*

Арван секунд хүлээх, эсвэл шууд `Enter` товч дарж болно.

#### 2.4.1.2. sparc64 систем дээр эхлүүлэх

Ихэнх sparc64 системүүд нь дискнээс автоматаар эхлэхээр тохируулагдсан байдаг. FreeBSD суулгахын тулд та сүлжээгээр юм уу эсвэл CDROM-оос эхлүүлэх шаардлагатай. Энэ нь PROM (OpenFirmware) руу орохыг танаас шаарддаг.

Ингэхийн тулд системийг дахин ачаалж эхлүүлэх үеийн мэдэгдэл гарч ирэхийг хүлээх хэрэгтэй. Энэ нь тухайн загвараас шалтгаалах боловч доор дурдсантай төстэй харагдах ёстой:

```
Sun Blade 100 (UltraSPARC-IIe), Keyboard Present
Copyright 1998-2001 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.2, 128 MB memory installed, Serial #51090132.
Ethernet address 0:3:ba:b:92:d4, Host ID: 830b92d4.
```

Хэрэв таны систем дискнээс энэ үед ачаалж эхэлбэл та PROM хүлээх мөрөнд орохын тулд `L1` + `A` эсвэл `Stop` + `A` товчлуурын хослолыг дарах юм уу эсвэл цуваа консолоос (жишээ нь tip(1) эсвэл cu(1) дээр `~#`-г ашиглан) `BREAK`-г илгээх хэрэгтэй. Энэ нь иймэрхүү харагдах болно:

```
ok     ①
ok {0} ②
```

① Энэ нь зөвхөн нэг CPU-тай системүүд дээр хэрэглэгддэг хүлээх мөр юм.

② Энэ нь SMP системүүд дээр хэрэглэгддэг хүлээх мөр юм. тоо нь идэвхтэй байгаа CPU-ийн

тоог илэрхийлдэг.

Энэ хүрэхэд өөрийн хөтөчдөө CDROM-оо хийгээд PROM хүлээх мөрөөс `boot cdrom` тушаалыг бичих хэрэгтэй.

### 2.4.2. Төхөөрөмжийн шалгаж бүртгэсэн хэсгийг харах

Сүүлд нь урсаж өнгөрдөг хэдэн зуун мөр текст нь дэлгэцийн түр хадгалагч уруу хадгалагддаг бөгөөд сүүлд дахин харж болдог.

Түр хадгалагдсан мөрийг дахин харахын тулд `Scroll Lock` товч дараарай. Энэ нь дэлгэцэнд урсаж өнгөрсөн текстүүдийг дээш нь эргүүлж харах боломж өгдөг. Та дээш заасан сум товчоор юм уу эсвэл `PageUp` мөн `PageDown` товчнуудаар дээш доош гүйлгэн харж болно. `Scroll Lock` товчийг дахин дарж гулгуулах үйлдлээ зогсоодог.

Та уг товчийг дарж дээш гулгуулан харах хэрэгтэй. Энэ нь цөм хэрхэн төхөөрөмжүүдийг таньсан тухай харуулдаг. Та Төхөөрөмж таньсан бүртгэл бичлэг дээр харуулсантай ижил бичиглэл харах бөгөөд харин төхөөрөмжүүдийн нэрс нь таны компьютерийнхаас өөр байж магадгүй.

*Төхөөрөмж таньсан бүртгэл бичлэг*

```
avail memory = 253050880 (247120K bytes)
Preloaded elf kernel "kernel" at 0xc0817000.
Preloaded mfs_root "/mfsroot" at 0xc0817084.
md0: Preloaded image </mfsroot> 4423680 bytes at 0xc03ddcd4

md1: Malloc disk
Using $PIR table, 4 entries at 0xc00fde60
npx0: <math processor> on motherboard
npx0: INT 16 interface
pcib0: <Host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcib1:<VIA 82C598MVP (Apollo MVP3) PCI-PCI (AGP) bridge> at device 1.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <Matrox MGA G200 AGP graphics accelerator> at 0.0 irq 11
isab0: <VIA 82C586 PCI-ISA bridge> at device 7.0 on pci0
isa0: <iSA bus> on isab0
atapci0: <VIA 82C586 ATA33 controller> port 0xe000-0xe00f at device 7.1 on pci0
ata0: at 0x1f0 irq 14 on atapci0
ata1: at 0x170 irq 15 on atapci0
uhci0 <VIA 83C572 USB controller> port 0xe400-0xe41f irq 10 at device 7.2 on pci
0
usb0: <VIA 83572 USB controller> on uhci0
usb0: USB revision 1.0
uhub0: VIA UHCI root hub, class 9/0, rev 1.00/1.00, addr1
uhub0: 2 ports with 2 removable, self powered
pci0: <unknown card> (vendor=0x1106, dev=0x3040) at 7.3
dc0: <ADMtek AN985 10/100BaseTX> port 0xe800-0xe8ff mem 0xdb000000-0xeb0003ff ir
q 11 at device 8.0 on pci0
dc0: Ethernet address: 00:04:5a:74:6b:b5
```

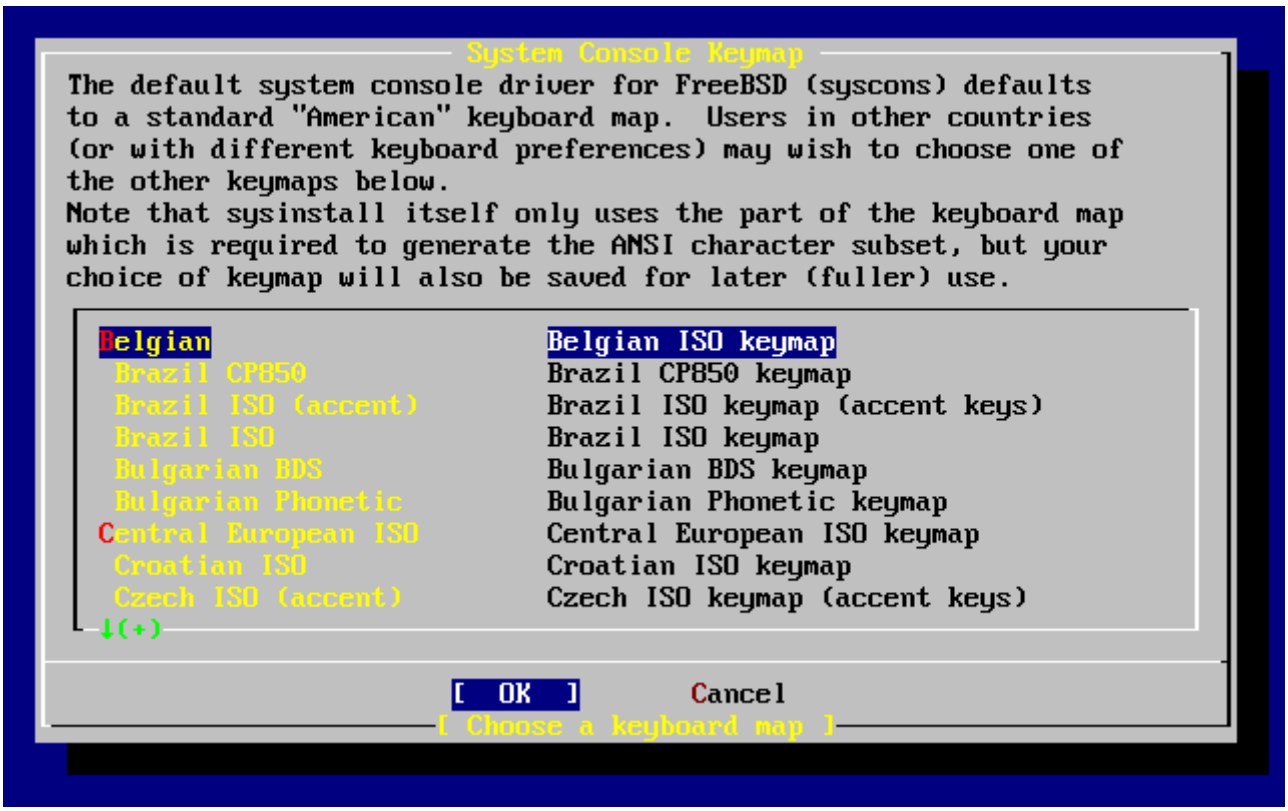
```
miibus0: <MII bus> on dc0
ukphy0: <Generic IEEE 802.3u media interface> on miibus0
ukphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
ed0: <NE2000 PCI Ethernet (RealTek 8029)> port 0xec00-0xec1f irq 9 at device 10.
0 on pci0
ed0 address 52:54:05:de:73:1b, type NE2000 (16 bit)
isa0: too many dependant configs (8)
isa0: unexpected small tag 14
orm0: <Option ROM> at iomem 0xc0000-0xc7fff on isa0
fdc0: <NEC 72065B or clone> at port 0x3f0-0x3f5,0x3f7 irq 6 drq2 on isa0
fdc0: FIFO enabled, 8 bytes threshold
fd0: <1440-KB 3.5" drive> on fdc0 drive 0
atkbdc0: <Keyboard controller (i8042)> at port 0x60,0x64 on isa0
atkbd0: <AT Keyboard> flags 0x1 irq1 on atkbdc0
kbd0 at atkbd0
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: model Generic PS/@ mouse, device ID 0
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
sio0 at port 0x3f8-0x3ff irq 4 flags 0x10 on isa0
sio0: type 16550A
sio1 at port 0x2f8-0x2ff irq 3 on isa0
sio1: type 16550A
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
pppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/15 bytes threshold
plip0: <PLIP network interface> on ppbus0
ad0: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata0-master UDMA33
acd0: CD-RW <LITE-ON LTR-1210B> at ata1-slave PIO4
Mounting root from ufs:/dev/md0c
/stand/sysinstall running as init on vty0
```

FreeBSD-дээр таны хүссэн төхөөрөмжийг зөв таньсан эсэхийг уг бичлэг дээр тулгаж харах хэрэгтэй. Хэрэв төхөөрөмж олдоогүй бол уг бичлэгт харуулагдахгүй. Өөрчлөн тохируулсан цөм нь дууны карт зэрэг GENERIC цөмд байхгүй төхөөрөмжүүдийн дэмжлэгийг нэмэх боломжийг танд олгодог.

Төхөөрөмжийг таних үйлдлийн дараагаар та Улс сонгох цэс-г харах болно. Сумтай товчлуур ашиглан улс, бүс, эсвэл бүлэг сонгох хэрэгтэй. Дараа нь `Enter` дарахад энэ нь таны улсыг хялбараар тохируулах болно.

*Зураг 2. Улс сонгох цэс*

Хэрэв та улсаараа United States-г сонгосон бол стандарт Америк гарын байрлал ашиглагдана, хэрэв өөр улс сонгосон бол дараах цэс гарч ирнэ. Сумнуудыг ашиглан зөв гарын байрлалыг сонгож `Enter`-г дарна.




*Зураг 3. Гарын цэсийг сонгох*

Улсыг сонгосны дараа sysinstall-н үндсэн цэс дэлгэц дээр харагдах болно.

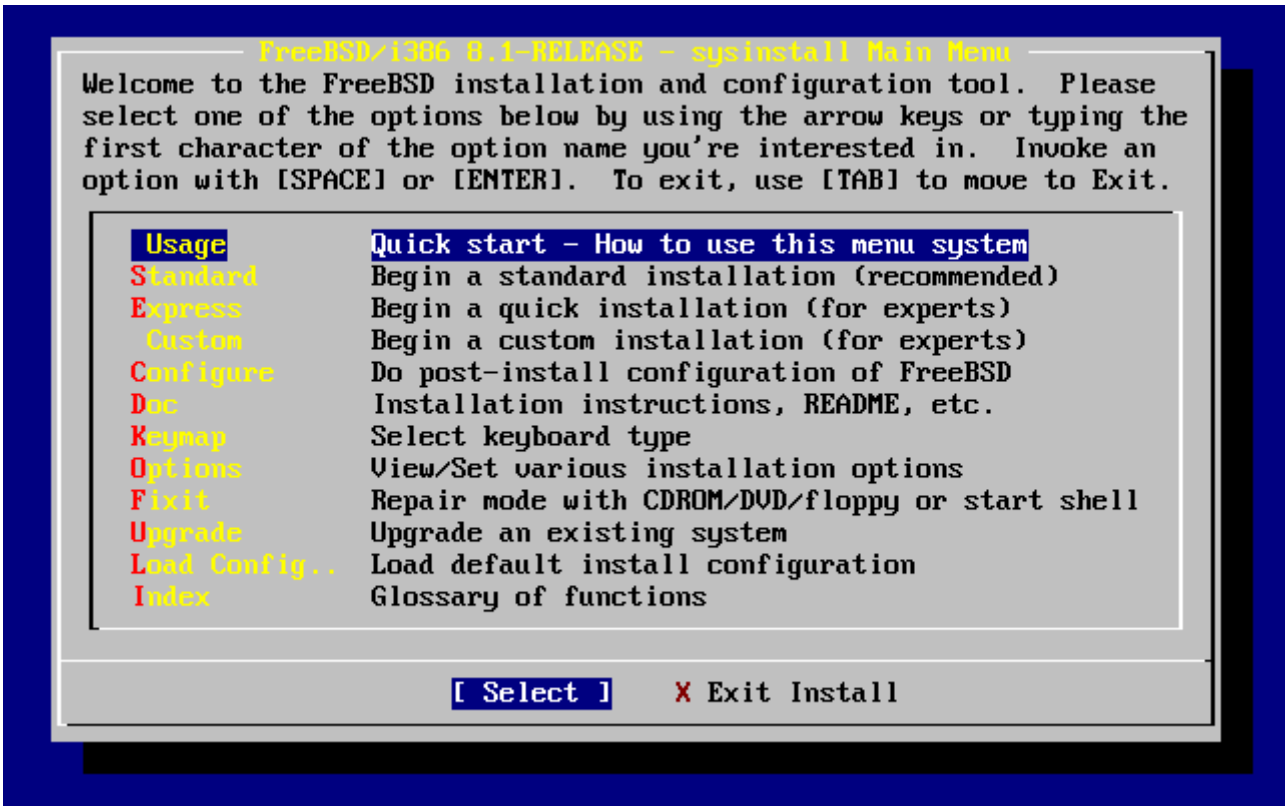
# 2.5. Sysinstall-н танилцуулга

sysinstall бол FreeBSD төслөөс гаргасан суулгац програм юм. Энэ нь консол дээр тулгуурлаж ажилладаг бөгөөд зохих цэсийг нь ашиглаж суулгах явцыг хянаж тохируулдаг.

sysinstall-н цэс нь сумтай товч, `Enter` буюу мөр нугалагч, `Tab`, `Space` буюу зай авагч болон бусад товчнуудаар залагддаг. Эдгээр товчны тухай дэлгэрэнгүй мэдээлэл sysinstall-н хэрэглэх зааварт бичээстэй буй.

Эдгээр мэдээллийг харахын тулд Usage буюу хэрэглээ хэсэгт очоод **[ Select ]** хэсгийг сонгож Sysinstall-н үндсэн цэсэнд Usage буюу хэрэглээг сонгох нь-д харуулсан шиг байдалд болгоод `Enter` товч дарах хэрэгтэй.

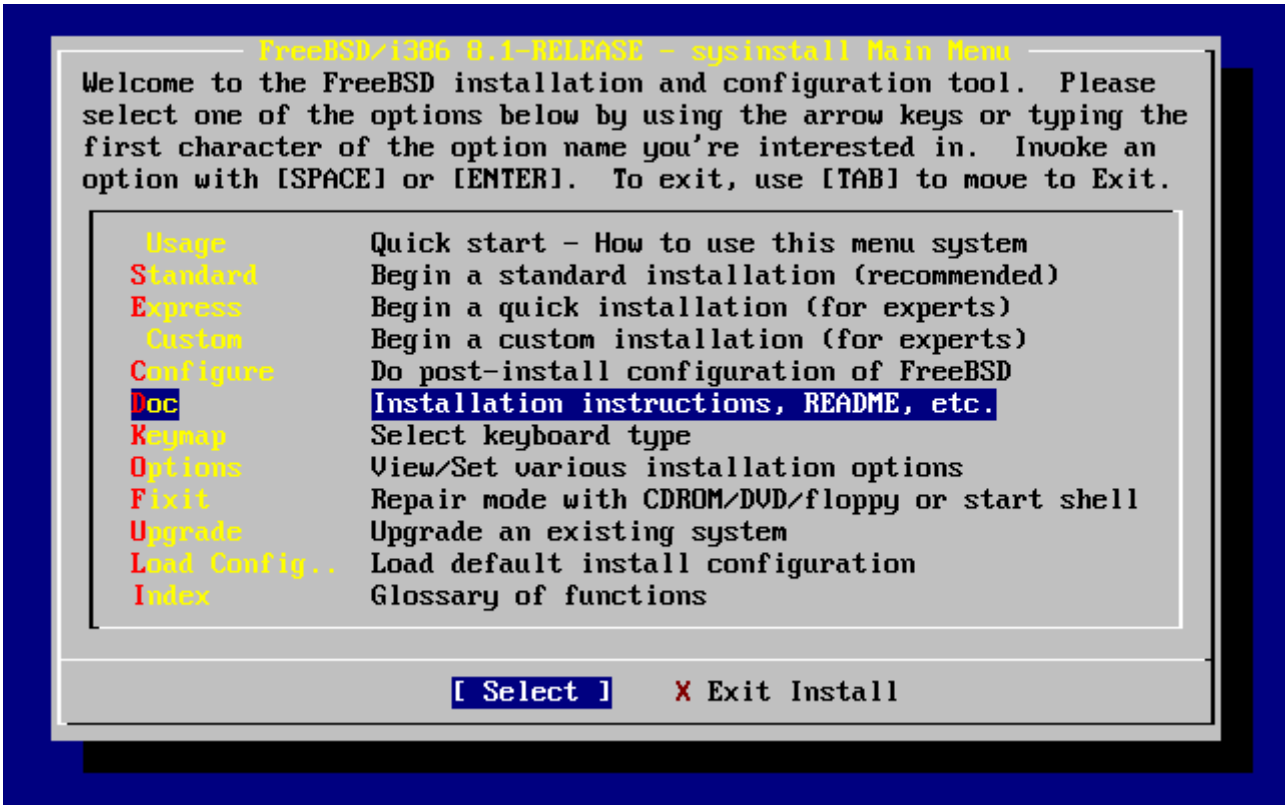
Цэстэй хэрхэн ажиллах тухай заавар харуулагдана. Хэрэглэх зааврыг уншиж дуусаад `Enter` товч дарж буцаад үндсэн цэсэнд очно.




*Зураг 4. Sysinstall-н үндсэн цэсэнд Usage буюу хэрэглээг сонгох нь*

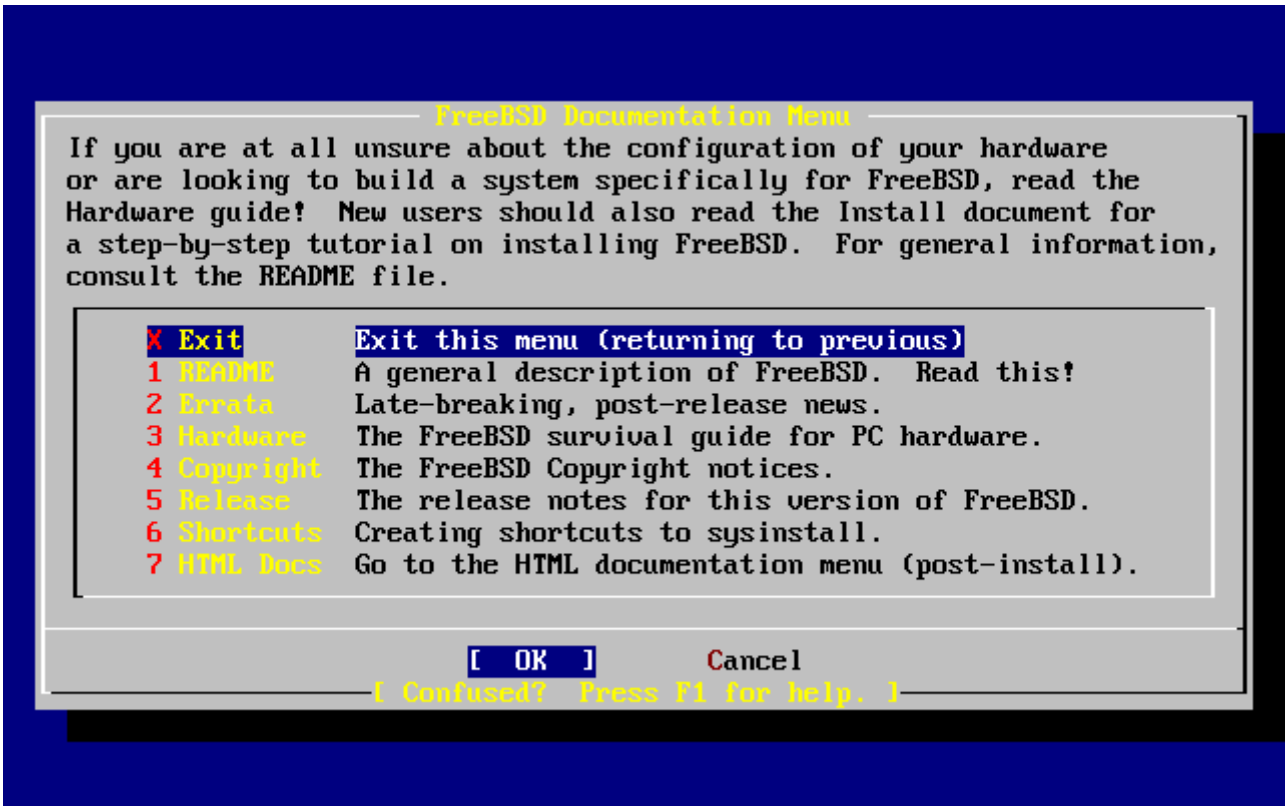
## 2.5.1. Documentation буюу бичиг баримт цэсийг сонгох

Үндсэн цэснээс сумтай товч ашиглан Doc цэсийг сонгоод `Enter` дарна.




*Зураг 5. Documentation буюу бичиг баримтын цэс сонгох*

Энэ нь бичиг баримтыг харуулах болно.




*Зураг 6. Sysinstall-н Documentation буюу бичиг баримтын цэс*

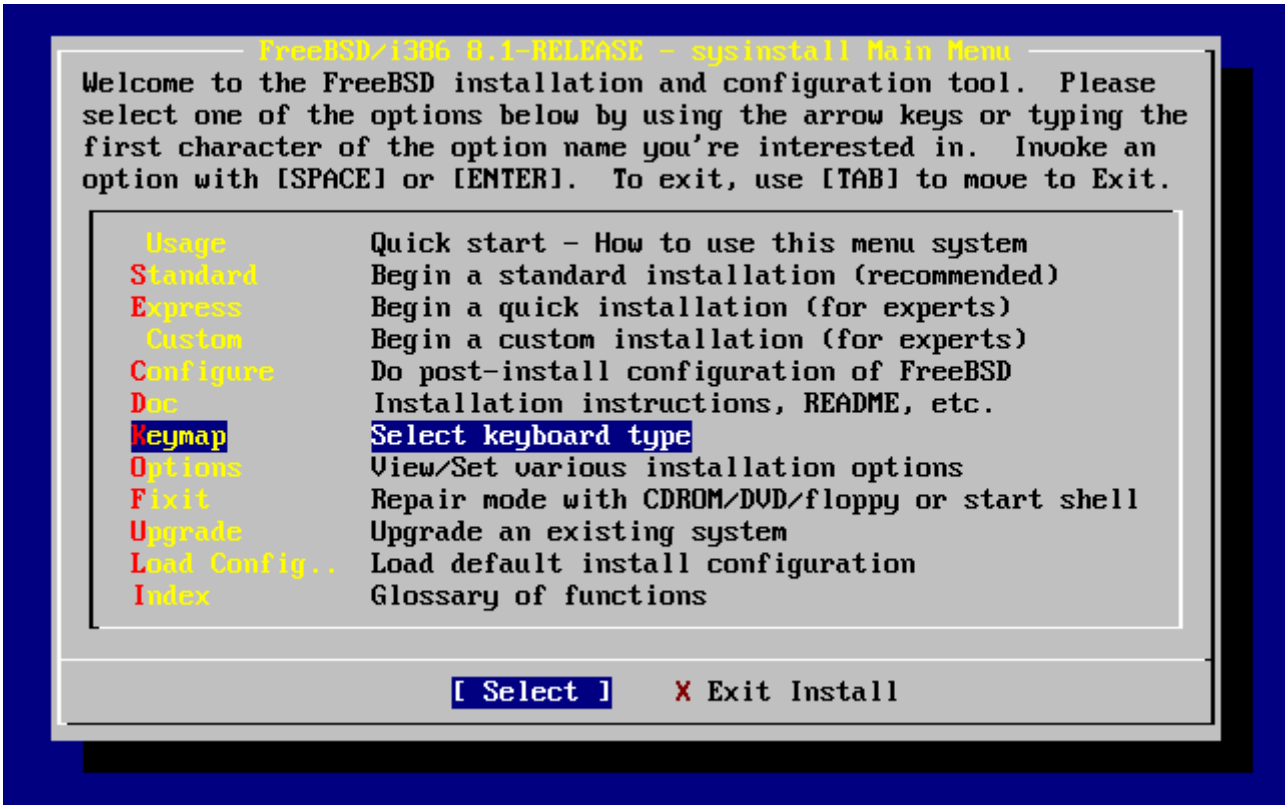
Хамт ирсэн бичиг баримтыг нь унших хэрэгтэй.

Бичиг баримтыг үзэхийн тулд сумтай товч хэрэглэж сонгоод `Enter` товч дарна. Уншиж дуусаад `Enter` товч дарвал Documentation буюу бичиг баримтын цэс уруу буцна.

Суулгацын үндсэн цэсэнд буцаж очихын тулд Exit буюу гарах гэсэн цэсийг сонгоод `Enter` товч дараарай.

### 2.5.2. Keymap буюу гарын товчлуур хуваарилалт цэсийг сонгох

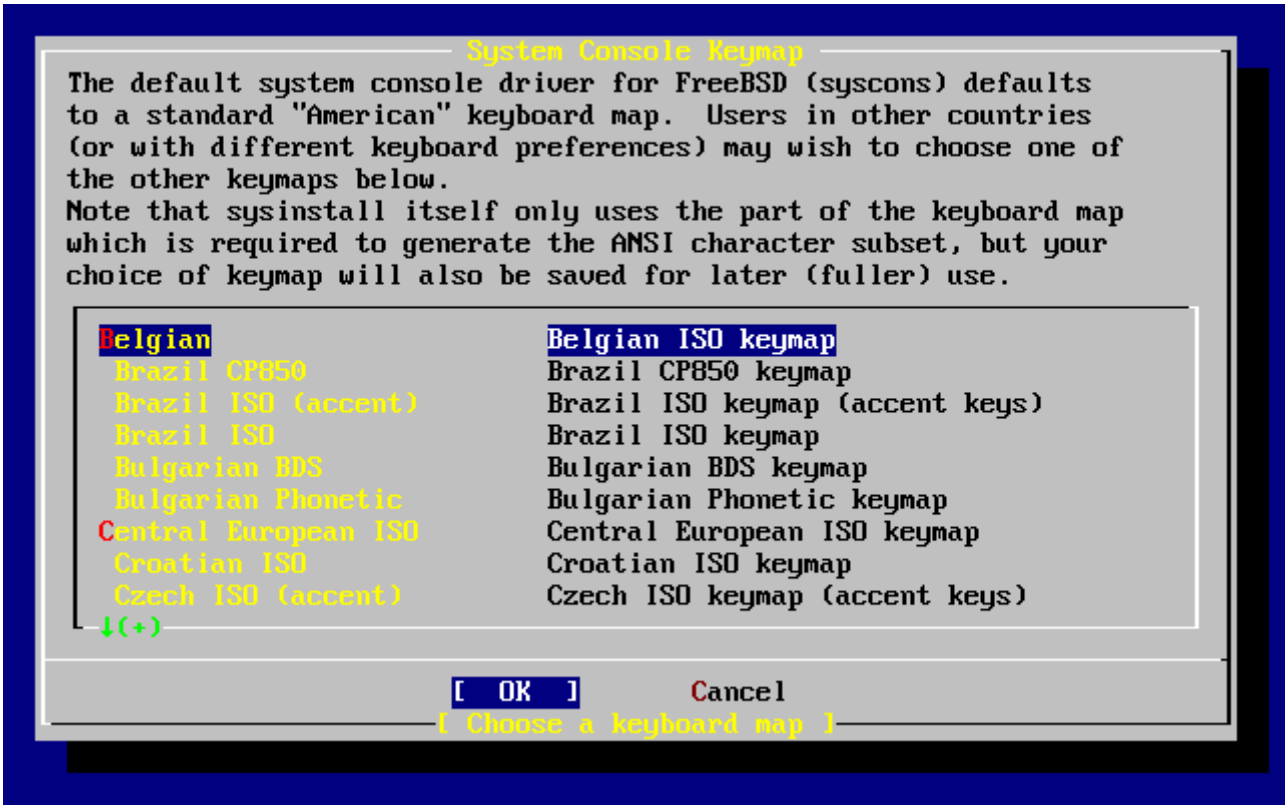
Гарын товчлуурын хуваарилалтыг өөрчлөхийг хүсвэл сумтай товчоор Keymap цэсийг сонгоод `Enter` товч дарна. Энэ нь зөвхөн стандарт буюу US америк гарын хуваарилалт хэрэглэдэггүй тохиолдолд л танд хэрэгтэй.




*Зураг 7. Sysinstall-н үндсэн цэс*

Өөр өөр гарын хуваарилалтыг та сумтай товчоор сонгоод `Space` товч дарж сонгох бөгөөд дахин `Space` дарж сонголтоо цуцлана. Сонгож дуусаад **[ OK ]** цэсийг сумтай товч ашиглан сонгоод `Enter` товч дараарай.

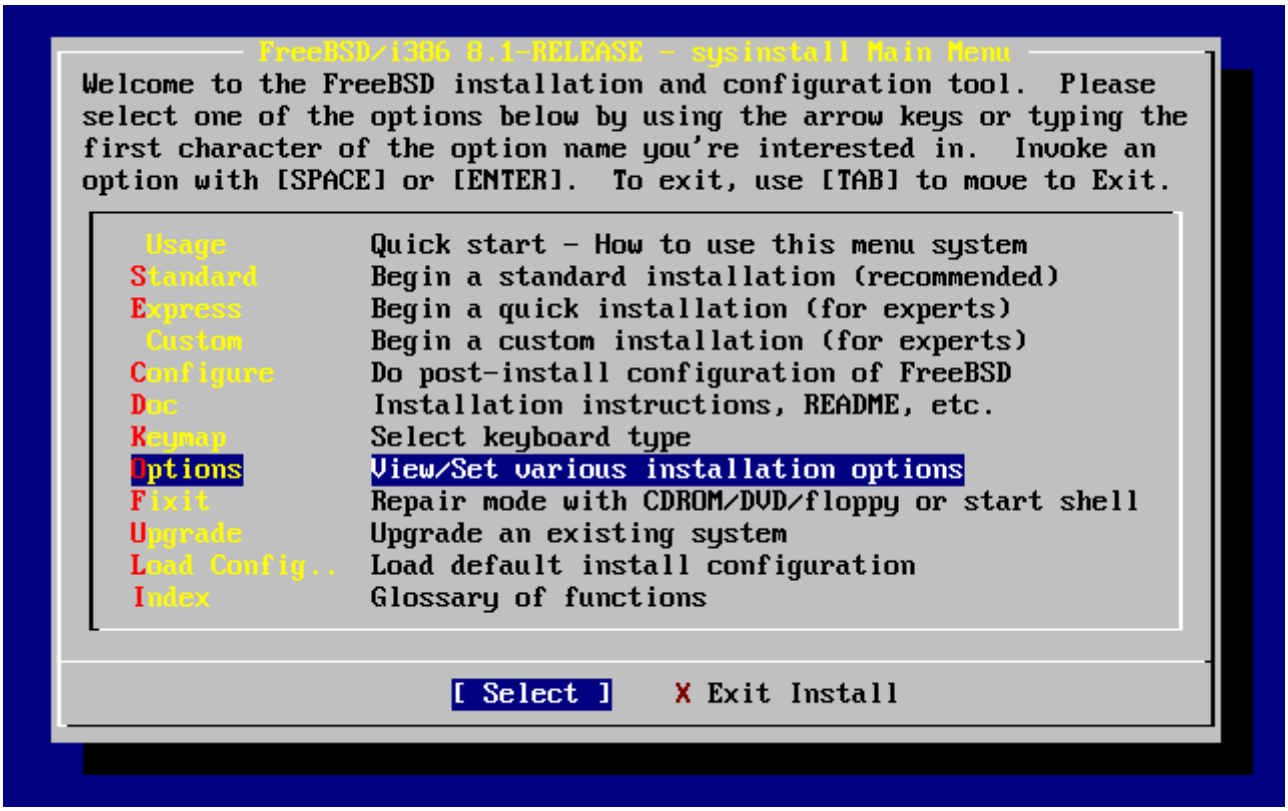
Энэ харуулсан дэлгэцэнд зөвхөн зарим хэсгийг нь харуулсан болно. `Tab` товч хэрэглэж **[ Cancel ]** цэсийг сонговол анхдагч гарын хуваарилалтыг сонгоод үндсэн цэс уруу буцдаг.



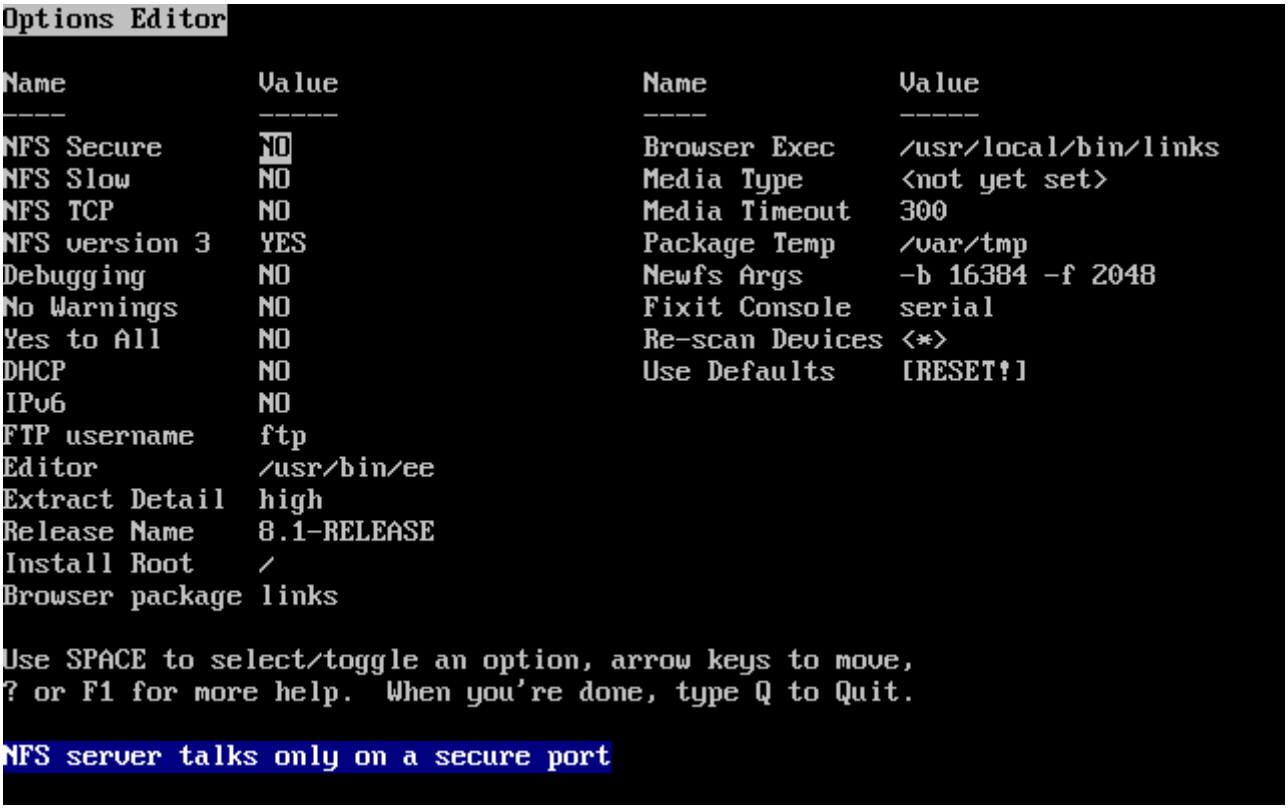

*Зураг 8. Sysinstall-н Кеутар буюу гарын төвчлуур хуваарилалт цэс*

### 2.5.3. Суулгацын Options буюу тохируулгууд нүүр

Options цэсийг сонгоод `Enter` дарна.




*Зураг 9. Sysinstall-н үндсэн цэс*




*Зураг 10. Sysinstall Options хэсэг*

Анхдагч утга нь ихэнх хэрэглэгчдэд өөрчлөлтгүйгээр хэрэглэгдэхэд хангалттай. Хувилбарын нэр нь суулгаж байгаа төрлөөсөө хамаарч өөр өөр байна.

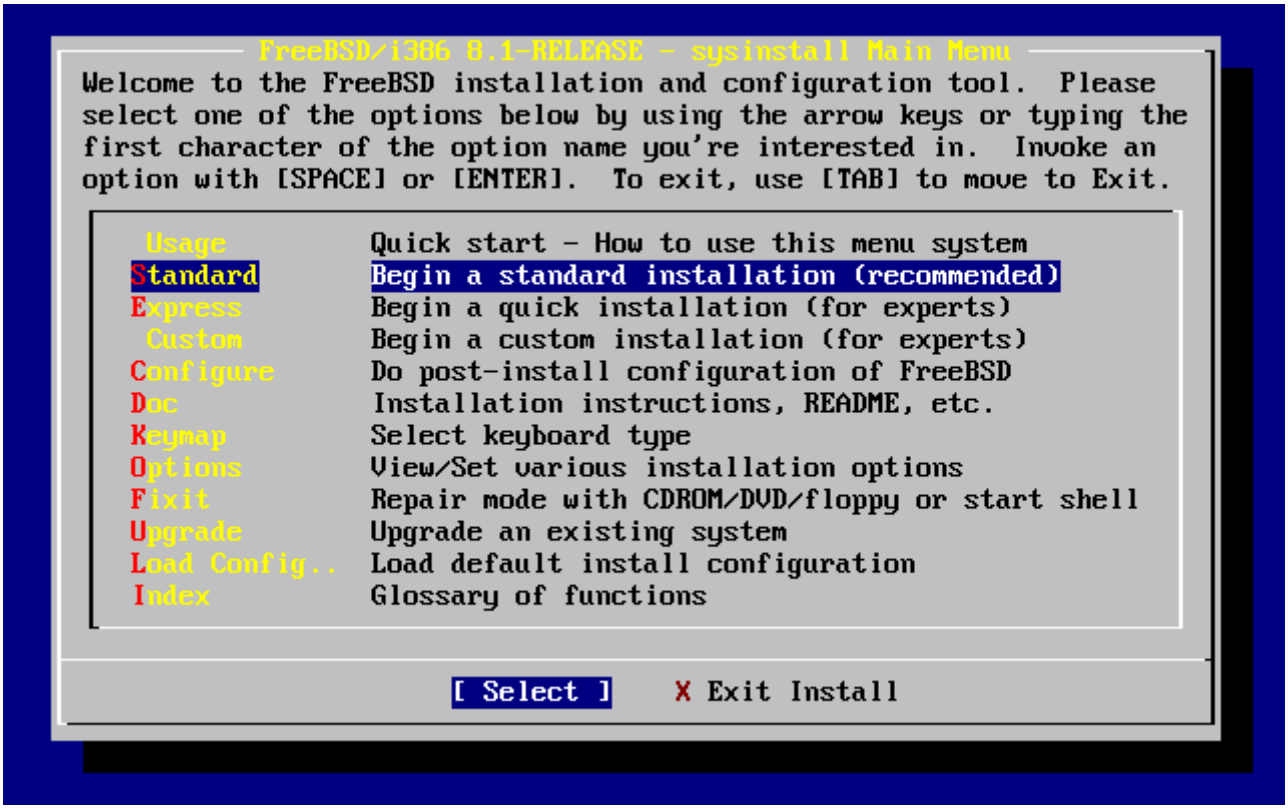
Сонгогдсон цэсийн тайлбар нь дэлгэцийн доод хэсэгт цэнхэр дэвсгэртэй бичигддэг. Тэмдэглэж хэлэхэд, Use Defaults цэсийг сонговол бүх утгыг анхдагч утгад нь тохируулдаг.

`F1` товч дарж сонголтын төрөл бүрийн мэдээлэл агуулсан туслах мэдээллийг харж болно.

`Q` товч дарвал үндсэн цэсэнд буцаж очно.

### 2.5.4. Үндсэн суулгацыг эхлүүлэх

UNIX® эсвэл FreeBSD үйлдлийн системийг сурч байгаа хүнд бол Standard цэсийг сонгож үндсэн суулгацыг эхлүүлэх хэрэгтэй. Сумтай товч хэрэглэн Standard цэсийг сонгоод `Enter` товч дарвал үндсэн суулгац эхэлнэ.




*Зураг 11. Үндсэн суулгацыг эхлүүлэх нь*

# 2.6. Дискний зайг зохицуулах

Таны эхний үйлдэл бол FreeBSD-д зориулж дискний зай бэлдэж түүндээ нэр өгнө. Ингэснээр sysinstall уг дискийг таньж бэлддэг. Үүнийг хийхийн тулд FreeBSD диск дээр байгаа мэдээллийг хэрхэн уншиж хэрэглэдэг талаар мэдэх хэрэгтэй.

## 2.6.1. BIOS-н диск дугаарлалт

Өөрийнхөө систем дээр FreeBSD -г суулгаж тохируулахын тулд зарим зүйлсийг анхаарч мэдэх хэрэгтэй. Ялангуяа та олон диск хэрэглэдэг бол энэ нь тун чухал.

Компьютер дээр BIOS-оос хамаарч ажилладаг MS-DOS® эсвэл Microsoft® Windows® зэрэг үйлдлийн системүүдэд, BIOS дискнүүдийг дугаарладаг бөгөөд уг үйлдлийн системүүд нь уг дугаарласан өөрчлөлтийг нь дагаж ажилладаг. Энэ нь "primary master буюу анхны дискнээс" өөр дискнээс үйлдлийн систем эхэлж ажиллах боломж өгдөг. Энэ арга нь Ghost эсвэл XCOPY зэрэг програм ашиглан нэг дискний ерөнхий зургийг хуулж түүнтэй адилхан хоёр дахь диск уруу хуулан системдээ найдвартай хадгалалт хийдэг хүмүүст тун хэрэгтэй байдаг. Тэгээд, хэрэв эхний диск эвдрэх, эсвэл вирустаж гэмтэх зэрэг хүндрэл гарвал, BIOS дээр дискнүүдийн дарааллыг өөрчилж хуулбарласан өгөгдөлтэй дискнээс үйлдлийн системээ эхлүүлэн ажиллаж болдог. Энэ нь бараг дискнүүдийн кабелийг хайрцгийг нь нээлгүйгээр сольж байгаатай ижил юм.

SCSI диск хянагчтай системүүд нь голдуу BIOS өргөтгөлтэй байдаг бөгөөд найм хүртэлх SCSI дискнүүдийг иймэрхүү зарчмаар дараалуулж чаддаг.

Иймэрхүү арганд дассан хүнд FreeBSD арай өөрөөр үйлчилдэг бөгөөд энэ нь их цочирдуулдаг. FreeBSD BIOS-г ашигладаггүй бөгөөд "BIOS-н логик дискний дугаарлалтыг

мэддэггүй". Энэ нь ялангуяа яг адилхан дискэн дээр дискний зургийг хуулбарласан үед бодсоноос өөр үйлчлэл үзүүлэхэд хүргэдэг.

FreeBSD-г хэрэглэх үедээ BIOS-г үргэлж өөрөөр нь диск дугаарлалт хийлгэж, тэр чигээр нь үлдээх хэрэгтэй. Хэрэв та дискний дугаарлалтыг өөрчлөхөөр бол, компьютерийнхаа хайрцгийг онгойлгож дискний сэлгүүр болон залгууруудыг тохируулж залгах хэрэгтэй.

Билл өөрийнхөө хуучин Wintel компьютераа янзалж FreeBSD суулгаад Фрэдэд өгөхөөр болжээ. Билл нэг SCSI хянагчтай бөгөөд түүний эхний SCSI диск дээр FreeBSD-г суулгажээ.

Фрэд суулгасан системийг хэрэглэж эхлэв. Гэвч хэсэг өдрийн дараа хуучин SCSI дискэн дээр зарим алдаанууд гараад байна гэж Биллд хэлжээ.

Хэдэн өдрийн дараа уг хүндрэлийг Билл засахаар шийджээ. Тэгээд арынхаа өрөөнөөс уг дисктэй ижил, "нөөц хадгалалт" хийсэн дискээ авчирчээ. Авчирсан дискэн дээрээ гадаргууны шалгалт хийхэд ямар ч алдаагүй гэж гарч ирэв. Тэгэхээр нь нөхөр Билл тэр дискийг SCSI хяналтын дөрөв дэх залгуур дээр залгаад эхний дискнээс дөрөв дэх диск уруу нөөц image буюу хуулбар дүрс хийж авч гэнэ. Шинэ залгасан диск сайхан ажиллаж байсан тул баярласан Билл уг дискийг цаашид хэрэглэхээр шийдээд SCSI BIOS дээр үйлдлийн системийг ачаалах дискний дугаарыг дөрөв болгож өөрчилжээ. FreeBSD-н эхлэх ачаалалт зүгээр байсан бөгөөд маш сайхан ажиллаж эхлэв.

Фрэд цааш нь хэдэн хоног ажилласны дараа төдий удалгүй Билл болон Фрэд хоёр FreeBSD-г шинэчлэх шинэ адал явдал хөөцөлдөх хүсэл төржээ. Билл SCSI хянагчийн эхний дискийг аваад оронд нь үүнтэй ижилхэн өөр "нөөцөлсөн" диск авчирж залгав. Билл FreeBSD -н шинэ хувилбарыг эхний SCSI диск дээр Фрэдийн шидэт интернэтээс FTP-ээс татаж авсан уян дискнээс эхлүүлж суулгав. Суулгац маш амжилттай болжээ.

Фрэд гуай FreeBSD -н шинэ хувилбарыг хэдэн хоног туршиж үзээд инженерийн салбарт хэрэглэхэд тун тохиромжтой юм байна гэсэн дүгнэлт өгч гэнэ. Ингээд хуучин хувилбар дээр хийж байсан ажлуудаа хэрэглэх хэрэг болж гэнэ. Тэгээд Фрэд гуай дөрөвдүгээр SCSI дискээ mount буюу холболт хийж (өмнө суулгасан FreeBSD-н хуучин хувилбар) гэнэ. Фрэд гуайн нүдэнд дөрөв дэх SCSI дискэн дээр өөрийнх нь хийж байсан ажлууд байхгүй байлаа.

Тэр өгөгдлүүд хаачсан бэ?

Эхний дискнээс дөрөв дэх диск уруу Билл гуай дүрс хуулалт хийснээр дөрөв дэх диск нь "хуулбар" болсон билээ. Билл гуайн SCSI BIOS дээр дөрөв дэх дискнээс ачаалалт эхлүүлнэ гэсэн тохиргоо бол тэнэглэл байжээ. FreeBSD нь SCSI BIOS тохируулгыг үл харгалзан эхний SCSI дискнээс эхлүүлсээр байсан байна. BIOS дээр иймэрхүү өөрчлөлт хийснээр зарим ачаалах үйлдэл болон ажиллуулагч тохируулгыг өөрчилдөг боловч, FreeBSD эхлэх үедээ энэ тохируулгыг харгалзаж үздэггүй бөгөөд өөрийнхөөрөө диск дугаарлалт хийж эхэлдэг. Энд үзүүлснээр, систем нь эхний SCSI дискнээс эхэлсээр байсан бөгөөд Фрэдийн бүх өгөгдөл дөрөв дээр биш эхний диск дээр байсан байна. Хүмүүст бол дөрөв дэх SCSI дискнээс эхэлж байгаа мэт харагдсан байна.

Ийм үйлдэл болсны дараа ямар ч өгөгдөл устаж алга болоогүй болохыг бид танд

мэдэгдэж байгаадаа баяртай байна. Учир нь, хуучин эхний SCSI дискийг буцааж залгаад Фрэдийн бүх өгөгдлийг буцааж авч чаджээ. (Билл эхний дискнээс эхлүүлсэн байна).

Хэдийгээр бид нар SCSI дискэн дээр жишээ татсан боловч иймэрхүү үйлдэл IDE диск дээр бас тохиолдож болно.

### 2.6.2. FDisk ашиглан дискний зүсмэл үүсгэх

Энэ хэсэгт хийсэн өөрчлөлт тань диск уруу бичигдэхгүй. Хэрэв та ямар нэгэн алдаа хийж гэж бодоод дахин шинээр эхлэхийг хүсвэл sysinstall-н гарах цэсийг ашиглах, эсвэл `U` товч дарж Undo буюу буцаж үйлдэж болно. Хэрэв та бүр эргэлзэж юу хийхээ мэдэхгүй болоод ирвэл компьютераа шууд унтрааж болно.

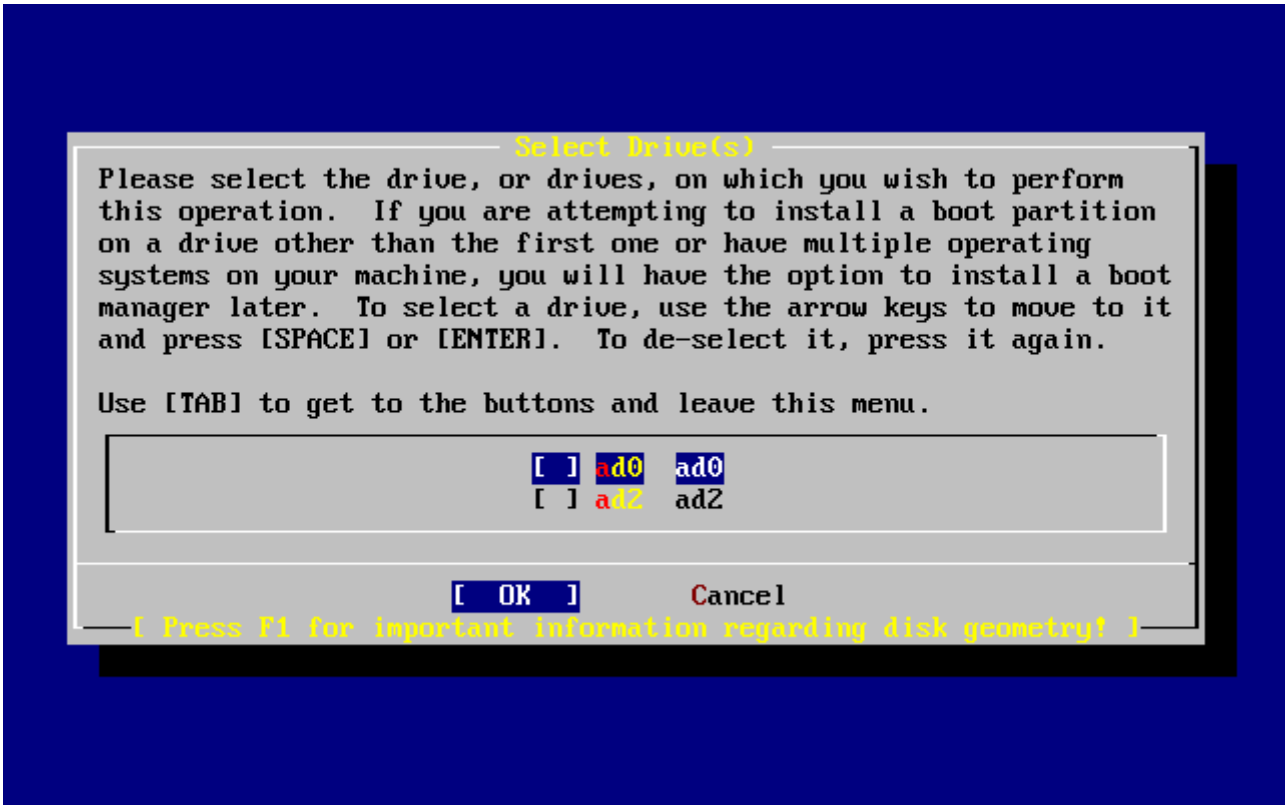
Стандарт суулгацыг сонгосны дараа sysinstall танд дараах мэдэгдлийг харуулна:

```
                                Message
In the next menu, you will need to set up a DOS-style ("fdisk")
partitioning scheme for your hard disk. If you simply wish to devote
all disk space to FreeBSD (overwriting anything else that might be on
the disk(s) selected) then use the (A)ll command to select the default
partitioning scheme followed by a (Q)uit. If you wish to allocate only
free space to FreeBSD, move to a partition marked "unused" and use the
(C)reate command.
                               [  OK  ]

                     [ Press enter or space ]
```

Энэ хэсгийг орчуулбал:

```
      Мэдэгдэл
      Дараагийн хэсэгт та өөрийнхөө дискэнд DOS-маягийн ("fdisk")
      диск хуваалт хийх хэрэгтэй. Хэрэв та дискнийхээ бүх хэмжээг
      FreeBSD-д зориулна гэж бодож байвал (дискэн дээр байгаа бүх өгөгдлийг
      дарж бичнэ) (A)ll тушаалыг сонгоод дараа нь (Q)uit цэсийг сонгож
      гараарай. Хэрэв та зөвхөн сул чөлөөтэй байгаа хэсгийг FreeBSD-д
      зориулна гэж бодож байгаа бол "unused" буюу хэрэглэгдээгүй гэсэн хэсгийг
      сонгоод (C)reate буюу үүсгэ гэсэн тушаалыг сонгоорой.
      [  OK  ]

      [ enter товч эсвэл зай авагч товч дарна уу]
```

Энд голдуу `Enter` товч дардаг. Ингэсний дараа цөмд эхлэх үед таньж туршсан хатуу дискнүүдийн жагсаалтыг танд харуулах болно. FDisk-н хэрэглэх дискийг сонгох дээр IDE дисктэй системийн жишээ харуулав. Тэдгээр нь ad0 болон ad2 гэсэн нэртэй буй.




*Зураг 12. FDisk-н хэрэглэх дискийг сонгох*

Та магадгүй яагаад ad1 дискийг энд харуулсангүй вэ? гэж гайхаж мадагүй.

Танд хоёр ширхэг IDE хатуу диск байна гэж үзье. Нэг нь нэг IDE залгуур дээр мастер диск болж залгагдсан бөгөөд нөгөөх нь хоёр дахь IDE залгуур дээр суугдсан байг. Хэрэв FreeBSD нь тэдгээрийг ad0 ба ad1 гэж дугаарласан бол бүх юм зүгээр л ажиллах байсан.

Гэвч, хэрэв та гурав дахь дискийг эхний IDE залгуур дээрх мастер дисктэй боол болгож залгавал уг диск ad1 гэж дугаарлагдах бөгөөд өмнө нь ad1 гэж дугаарлагдсан диск ad2 гэж нэрлэгдэнэ. Дискэнд өгсөн нэрийг ашиглан (жишээ нь ad1s1a) файл системийг хайхад хэрэглэдэг учраас гэнэт таны дискнүүд өөр харагдаж эхлэх бөгөөд та FreeBSD-н тохируулгыг дахин хийх шаардлага гарна.

Ийм хүндрэлийг арилгахын тулд цөм нь IDE дээр залгагдсан дискнүүдийг таньсан дарааллаар нь биш харин залгагдсан байрлалаас нь хамааран дугаарладаг. Ийм учраас хэрэв IDE-н хоёр дахь залгуур дээр залгагдсан мастер диск нь *үргэлж* ad2 гэж нэрлэгдэх бөгөөд ad0 эсвэл ad1 дискнүүд бүр байхгүй байсан ч энэ нэрээрээ л байх болно.

Энэ нь FreeBSD-н цөмийн анхны тохируулга бөгөөд, ийм шалтгааны улмаас ad0 болон ad2 гэж харуулж байна. Тэгэхээр энэ зурган дээр байгаа машины IDE 2 залгуур дээр хоёр мастер диск залгагдсан бөгөөд ямар ч боол диск байхгүй байна гэж харуулж байна.

Та аль диск дээр нь FreeBSD-г суулгахаа сонгоод **[ OK ]** дээр дарах хэрэгтэй. FDisk эхлэх бөгөөд дэлгэц дээр Засварлаж эхлэхээс өмнөх `fdisk`-н диск хуваалтын жишээ-тэй төстэй зураг харуулагдах болно.

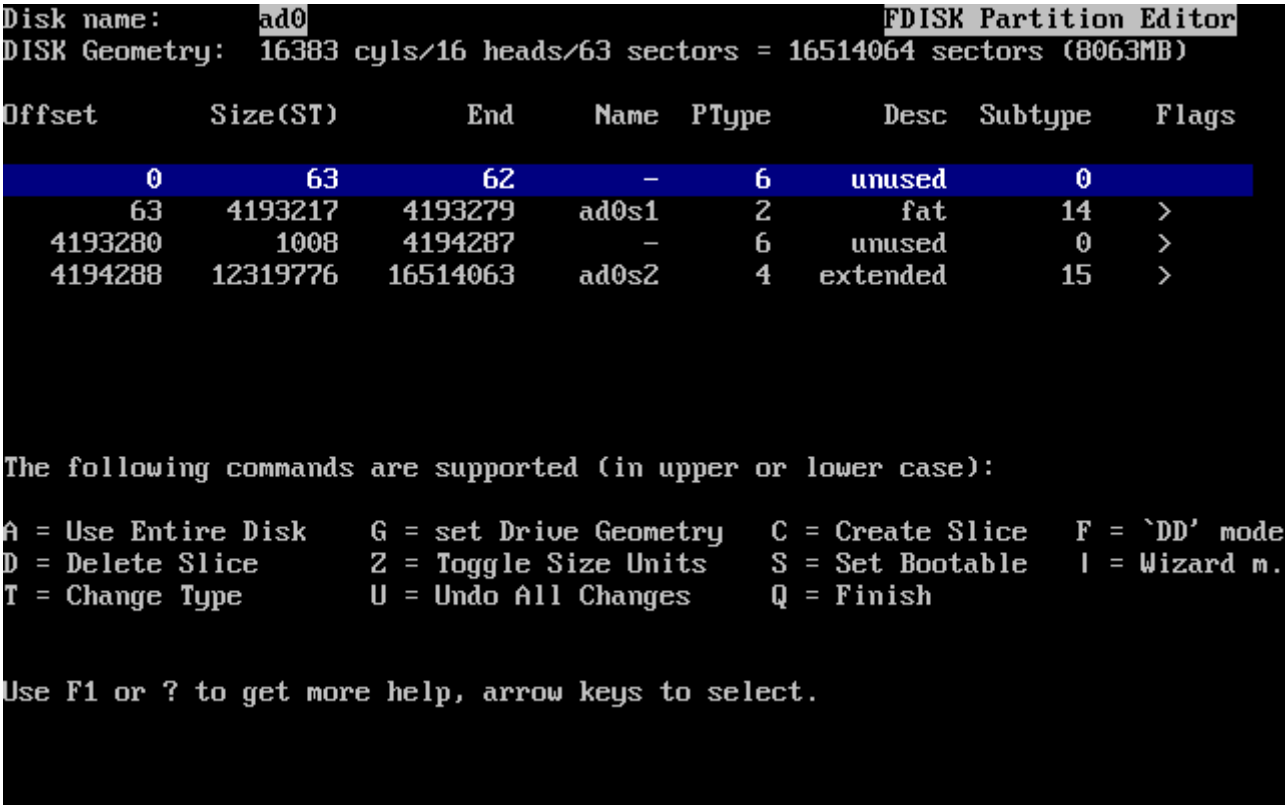
FDisk нь дэлгэц дээр гурван хэсэгт хуваагдаж харуулагддаг.

Эхний хэсэгт нь дээд хоёр мөр хамаарагддаг бөгөөд сонгогдсон дискний нарийвчилсан

мэдээллүүдийг харуулдаг. Энэ нь FreeBSD-н өгсөн нэр, дискний зохион байгуулалт мөн дискний нийт хэмжээ зэргийг харуулдаг.

Хоёр дахь хэсэгт дискэн дээр байгаа зүсмэлүүдийг харуулдаг ба хаанаас эхлээд хаана дууссан, ямар хэмжээгээр зүсэгдсэн бөгөөд FreeBSD хэрхэн нэр өгсөн, бас уг зүсмэлийг тодорхойлсон тодорхойлолт болон дэд төрлийг нь харуулдаг. Энэ жишээ дээр, компьютер дээрх дискний хэрэглэгдээгүй хоёр зүсмэлийг харуулсан байна. Уг зурган дээр бас нэг том FAT зүсмэл байгааг харуулсан байгаа бөгөөд магадгүй энэ нь MS-DOS® / Windows® системийн C: диск байж болзошгүй. Мөн уг зурган дээр бас нэг өргөтгөсөн зүсмэл байгааг харуулсан байгаа ба, бас энэ нь MS-DOS® / Windows® систем дээр нэг өргөтгөсөн диск байж магадгүй.

Гурав дахь хэсэг нь FDisk дээр хэрэглэж болох тушаалуудын жагсаалт байна.




*Зураг 13. Засварлаж эхлэхээс өмнөх `fdisk`-н диск хуваалтын жишээ*

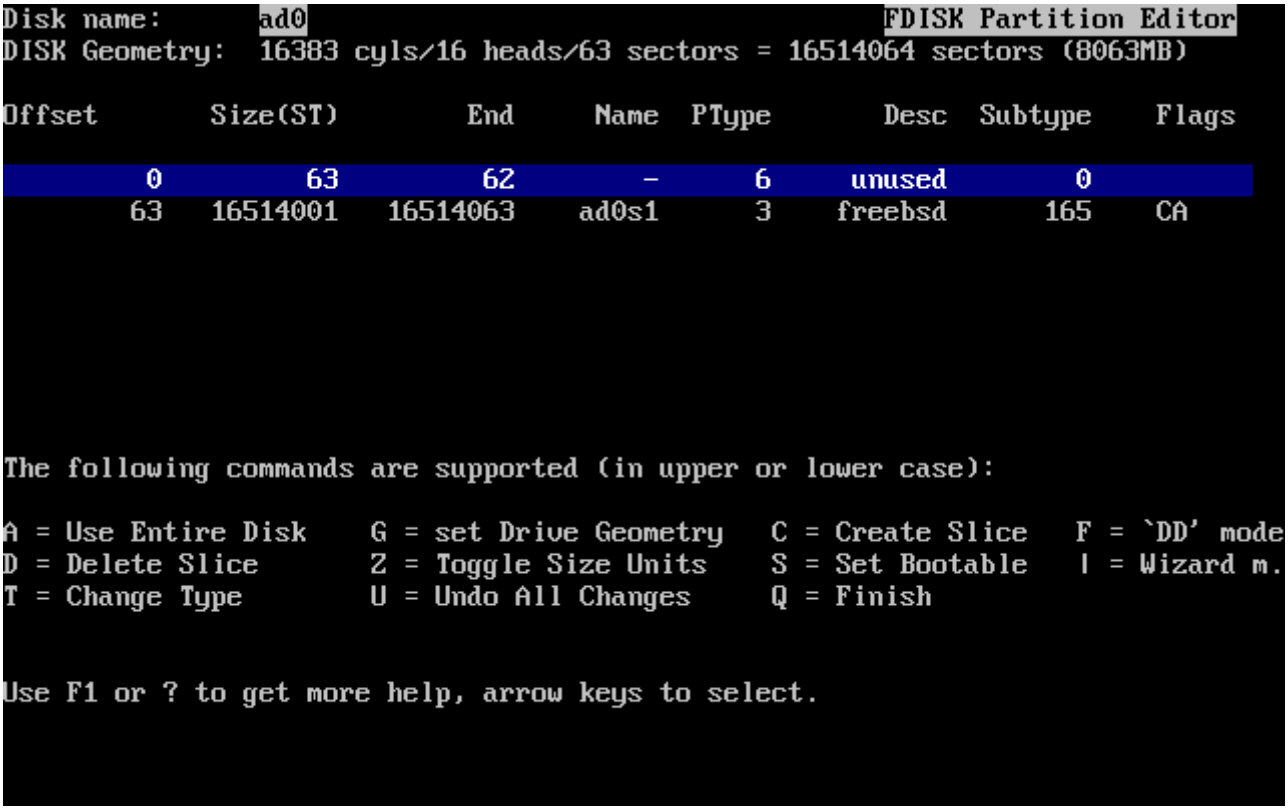
Одоо таны хийх алхам дискээ хэрхэн зүсэж хуваахаас их хамаарна.

Хэрэв та дискээ бүхлээр нь FreeBSD-д зориулна гэж бодож байвал (дискэн дээр байгаа бүх өгөгдлийг дарж суугдах бөгөөд sysinstall танаас суулгацын явцад лавлаж асуух болно) Use Entire Disk буюу дискийг бүхлээр нь хэрэглэ гэсэн заалтад буй `A` товч дарах хэрэгтэй. Байсан бүх зүсмэлүүд арилах бөгөөд жижигхэн хэсэг нь `unused буюу хэрэглэгдээгүй` гэж хуваагдаад (компьютер диск шалгахад зориулагддаг хэсэг) үлдсэн том хэсэг нь FreeBSD-д зориулж хуваагддаг. Ингэж сонгосны дараа сумтай товч хэрэглэн шинээр үүссэн FreeBSD-н зүсмэлийг сонгоод `S` товч дарж уг зүсмэлээс эхлэн ачаалагддаг болгох хэрэгтэй. Таны дэлгэц Fdisk дискийг бүхлээр нь хэрэглэж буй жишээ-тэй төсөөтэй зураг харуулах ёстой. Тэмдэглэж хэлэхэд, `Flags` баганад буй `A` үсэг нь уг зүсмэлийг *active буюу идэвхтэй* гэдгийг илэрхийлж байгаа бөгөөд энэ зүсмэлээс эхлэж ачаалагдах болно гэдгийг харуулж байгаа юм.

Хэрэв та өмнө нь хэрэглэгдэж байсан зүсмэлүүдээс нэгийг нь FreeBSD-д зориулна гэж бодсон

бол уг зүсмэлийг сонгоод D товч дарж устгах хэрэгтэй. Дараа нь та C товч дарвал үүсгэх зүсмэлийн хэмжээг танаас асуудаг. Хүссэн хэмжээгээ оруулаад Enter товч дарах хэрэгтэй. Уг лавлаж асуусан цонх дээр буй анхны тоо бол уг зүсмэл дээр хэрэглэж болох хамгийн их хэмжээг зааж байдаг тул уг зүсмэлийг бүхлээр нь хэрэглэнэ гэж бодвол шууд уг тоог өөрчлөлгүй хэрэглэх хэрэгтэй.

Хэрэв та урьдчилан FreeBSD -д зориулан дискээ суллачихсан байвал (магадгүй PartitionMagic® гэх мэтийн програм ашиглан дискээ бэлтгэсэн бол) дараа нь C товч дарж шинэ зүсмэл үүсгээрэй. Дахин сануулахад, зүсмэл үүсгэх үед хуваах зүсмэлийн хэмжээг танаас асуух болно.




*Зураг 14. Fdisk дискийг бүхлээр нь хэрэглэж буй жишээ*

Хувааж дуусаад Q товч дарж гарна. Таны өөрчилсөн өөрчлөлтүүд sysinstall дээр хадгалагдах бөгөөд диск уруу одоохондоо бичигдээгүй байгаа.

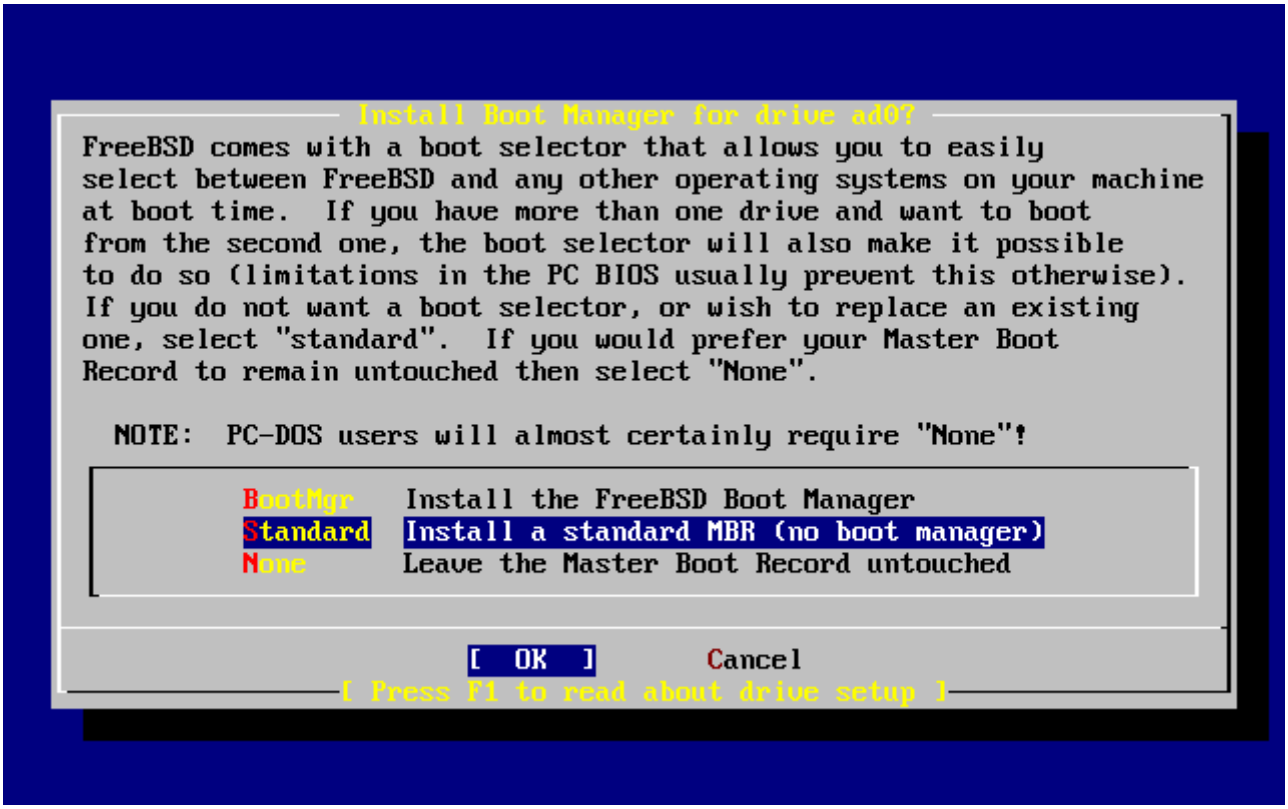
### 2.6.3. Boot Manager буюу эхлэн ачаалалт зохицуулагчийг суулгах

Танд одоо эхлэн ачаалалт зохицуулагчийг суулгах эсэхээ сонгох боломж гардаг. Хэрэв танд дараах нөхцлүүд биелж байвал голдуу FreeBSD boot manager-ийг сонгох нь элбэг:

- Танд олон дискнүүд байгаа бөгөөд FreeBSD-г эхнийх дээр нь суулгаагүй бол.
- Та FreeBSD-г өөр үйлдлийн системтэй хамт нэг диск дээр суулгаад компьютер эхлэх үед FreeBSD-г эхлүүлэх үү эсвэл нөгөө үйлдлийн системийг эхлүүлэх үү гэж сонголт хиймээр байгаа бол.

Хэрэв FreeBSD нь уг машин дээр байгаа цорын ганц үйлдлийн систем байхаар бол Standard -г сонгоход хангалттай. Хэрэв та FreeBSD-г эхлүүлж чадах өөр ямар нэгэн програм суулгасан бол None -г сонгоорой.

Сонголтоо хийгээд `Enter` товчийг дар.



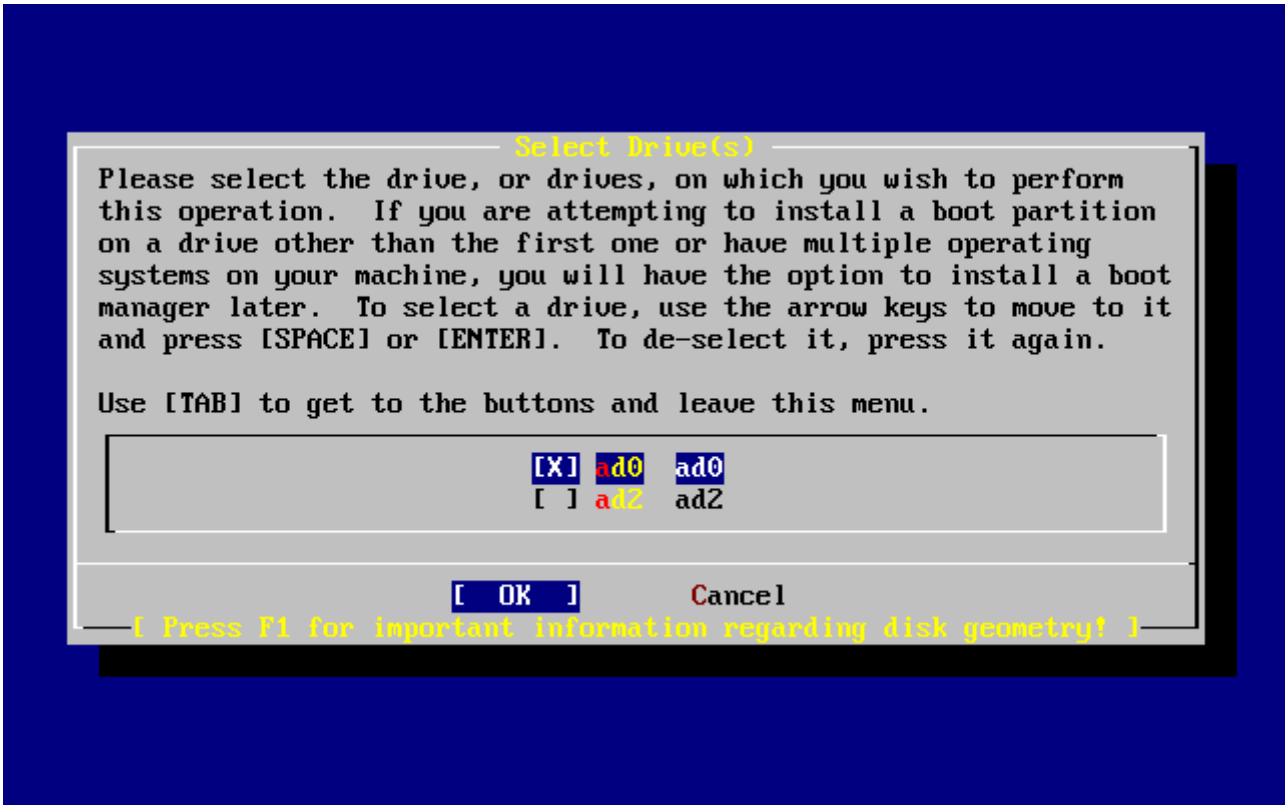

*Зураг 15. Sysinstall-н эхлэн ачаалалт зохицуулагчийн цэс*

`F1` товч дарвал тусламжийн цэс харуулагдах бөгөөд өөр үйлдлийн системтэй хамтарч хэрэглэх үед ямар хүндрэл үүсэж болох талаар тайлбарлаж өгдөг.

### 2.6.4. Бусад диск дээр зүсмэл үүсгэх

Хэрэв танд нэгээс илүү олон дискнүүд байгаа бол эхлэн ачаалалтын зохицуулагчийг сонгосны дараа диск сонгох цэсэнд буцаж очдог. Хэрэв та FreeBSD-г олон диск хэрэглэж суулгахыг хүсвэл, FDisk ашиглан цааш нь зүсэж хуваах хэрэгтэй.

Хэрэв та FreeBSD-г эхний дискнээс өөр диск дээр суулгаж байгаа бол FreeBSD-н эхлэн ачаалалтыг зохицуулагчийг хоёулан дээр нь суулгах хэрэгтэй.




*Зураг 16. Диск сонгох цэснээс гарах*

`Tab` товч хэрэглэн дискнүүд болон **[ OK ]**, эсвэл **[ Cancel ]** зэрэг тушаалууд уруу сэлгэж болно.

`Tab` товч хэрэглэн **[ OK ]**, дээр сонгоод `Enter` товч дарж суулгацыг цааш нь үргэлжлүүлнэ.

### 2.6.5. Disklabel буюу дискэнд нэр өгч хуваалт үүсгэх

Та одоо үүсгэсэн зүсмэл дотроо хуваалт үүсгэх ёстой. Хуваалт болгон `a` үсгээр эхлээд `h` хүртэл нэр авдаг бөгөөд `b`, `c` болон `d` гэсэн нэрнүүд нь тусгай зөвшлийн дагуу өөр зориулалтаар хэрэглэгддэг тул та үүнийг хүлээн зөвшөөрөх хэрэгтэй.

Хэрэв олон диск дээр хуваалт үүсгэж байгаа бол зарим програмууд нь зорилгоосоо хамааралд онцгой хуваалтын загвар хэрэглэдэг. Гэхдээ та одоогоор FreeBSD-г анх удаагаа суулгаж байгаа диск дээр хэрхэн хуваалт үүсгэх тухай нэг их бодох шаардлага байхгүй. Хамгийн чухал нь FreeBSD-г суулгаад хэрхэн хэрэглэдэг талаар сурах явдал юм. Та үйлдлийн системд нэлээн гаршиж сайжирсан үедээ FreeBSD-г дахин шинээр хэдийд ч суулгаж болно.

Энд үзүүлж буй хуваалтын загвар нь дөрвөн хуваалттай байна. Нэг нь swap хэмжээнд, бусад нь файлын системд зориулагдсан байна.

*Хүснэгт 2. Эхний дискний хуваалт*

| Хуваалт | Файл систем | Хэмжээ | Тодорхойлолт |
|---|---|---|---|
| a | / | 1 GB | Энэ бол root буюу эх файл систем юм. Бусад бүх файл системүүд ямар нэгэн аргаар энд танигдаж хэрэглэгддэг. 1 GB хэмжээ бол тухайн зорилгодоо таарсан хэмжээ. Та энд тийм их өгөгдөл хадгалахгүй бөгөөд ердийн FreeBSD-н суулгац нь энд 128 MB-г хэрэглэдэг. / дээр үлдсэн зай нь дараа нь хэрэглэгдэх зорилгоор юм уу эсвэл түр зуурын өгөгдөл зэрэгт зориулагддаг. |
| b | N/A | 2-3 x RAM | Системийн swap зай нь b хуваагдал дээр байрладаг. swap-д зориулж хэмжээгээ тохируулах нь бас чухал. Хамгийн сайн хэрэглэгддэг арга бол, байгаа санах ойныхоо (RAM) хэмжээнээс хоёр юм уу гурав дахин их хэмжээтэй байхад болно. Хэрэв танд 32 MB хэмжээтэй RAM санах ой байгаа бол ядаж 64 MB хэмжээтэй swap бэлдэх хэрэгтэй.<br><br>Хэрэв та нэгээс их дисктэй бол диск болгонд зориулж swap зай үүсгэж болно. Ингэвэл, FreeBSD нь диск болгонд буй swap зайг хэрэглэснээр илүү үр дүнтэй ажилладаг. Энэ тохиолдолд хэрэглэх нийт swap-нхаа хэмжээг (жишээ нь, 128 MB) байгаа дискнийхээ тоонд хувааж (жишээ нь хоёр дисктэй гэж үзье) гарсан хэмжээг дискэн дээр swap-д зориулж бэлдэх хэрэгтэй бөгөөд жишээний дагуу бол 64 MB диск болгонд ногдож байна. |
| e | /var | 512 MB-аас 4096 MB хүртэл | /var санд байнга өөрчлөгдөж байдаг файлууд байрладаг; бүртгэл файл, мөн бусад удирдах болон хянах файлууд энэ төрлийн файлд ордог. Эдгээр файлын ихэнх нь FreeBSD дээр ажилладаг програмуудаар өдрийн турш уншигдаж бас бичигдэж байдаг. Иймэрхүү файлуудыг нэг дор байрлуулснаар FreeBSD нь өөр файл систем уруу хандаж цаг заралгүй идэвхитэй ажиллаж чаддаг. |
| f | /usr | Дискний үлдсэн хэсэг (хамгийн багадаа 8 GB) | Бараг бусад бүх файлууд нь /usr санд болон түүн дотор буй дэд сангуудад байрладаг. |

Дээр дурдсан утганууд нь жишээ болон өгөгдсөн бөгөөд зөвхөн туршлагатай хэрэглэгчид эдгээрийг ашиглах хэрэгтэй. Хэрэглэгчид нь FreeBSD-ийн хуваалт засварлагчийн `Auto Defaults` гэж хэлэгддэг автомат хуваалтын хэлбэрийг ашиглах нь зүйтэй юм.

Хэрэв та FreeBSD-г олон дискнүүд дээр дамнан суулгаж байгаа бол бусад дискнүүд дээр үүсгэсэн зүсмэл дотроо хуваалт үүсгэх хэрэгтэй. Хамгийн амархан арга нь диск болгонд хоёр хуваагдал үүсгээд нэгийг нь swap зай болгоод нөгөөх нь ямар нэгэн файл систем болгох арга

байдаг.

*Хүснэгт 3. Бусад дискэн дээрх дискний хуваалт*

| Хуваалт | Файл систем | Хэмжээ | Тодорхойлолт |
|---|---|---|---|
| b | N/A | Тодорхойлолтоос уншина уу | Өмнө дурьдсанчлан, swap хэмжээг диск болгон дээр үүсгэж болдог. Хэдийгээр a хуваалт сул байсан ч гэсэн зарчмын дагуу swap зай нь b хуваалт дээр байрладаг. |
| e | /disk*n* | Дискний үлдсэн хэмжээ | Дискний үлдсэн хэмжээ нь нэг бүхэл хуваалт болдог. Энэ нь e хуваалт дээр биш харин a хуваалт дээр байрлаж болох байсан ч зарчмын дагуу a хуваагдал дээр root буюу эх файл систем (/) суугддаг. Та энэ зарчмыг дагахгүй байж болох боловч sysinstall харин дагадаг: Энэ зарчмыг дагаснаар суулгацыг цэвэрхэн болгодог. Та энэ файл системийг хаана ч холбож болох бөгөөд энэ жишээн дээр бол уг файл системийг /diskn гэсэн сан дотор холбосон байна. *n* үсэг нь дискний дугаарыг илэрхийлж байна. Гэхдээ та хүсвэл өөр газар холбож болно. |

Хуваагдлынхаа загвараа ингэж хийж дуусаад sysinstall-г ашиглаж үүсгэх хэрэгтэй. Үүсгэх үед дараах мэдэгдлийг танд харуулдаг:

```
                                Message
Now, you need to create BSD partitions inside of the fdisk
```

```
partition(s) just created. If you have a reasonable amount of disk
space (1GB or more) and don't have any special requirements, simply
use the (A)uto command to allocate space automatically. If you have
more specific needs or just don't care for the layout chosen by
(A)uto, press F1 for more information on manual layout.

                                [  OK  ]
                           [ Press enter or space ]
```

Үүний хөрвүүлбэл:

```
                                  Мэдэгдэл
Та одоо, fdisk ашиглан бий болгосон хуваалтаар  BSD хуваалт үүсгэх хэрэгтэй.
Хэрэв танд тодорхой хэмжээний дискний зай (200MB эсвэл түүнээс дээш) байгаа
бөгөөд ямар нэгэн онцгой шаардлага тавигдаагүй бол (A)uto тушаалыг ашиглан
дискний зайг автоматаар тохируулж болно. Хэрэв танд онцгой шаардлага бий юм уу
эсвэл (A)uto тушаалаар үүсгэх байрлуулалт тийм чухал биш бол
F1 товч дарж гарын авлага дээрх нэмэлт мэдээллийг харж болно

                                [  OK  ]
                           [ enter товч эсвэл зай авагч товч дарна уу]
```

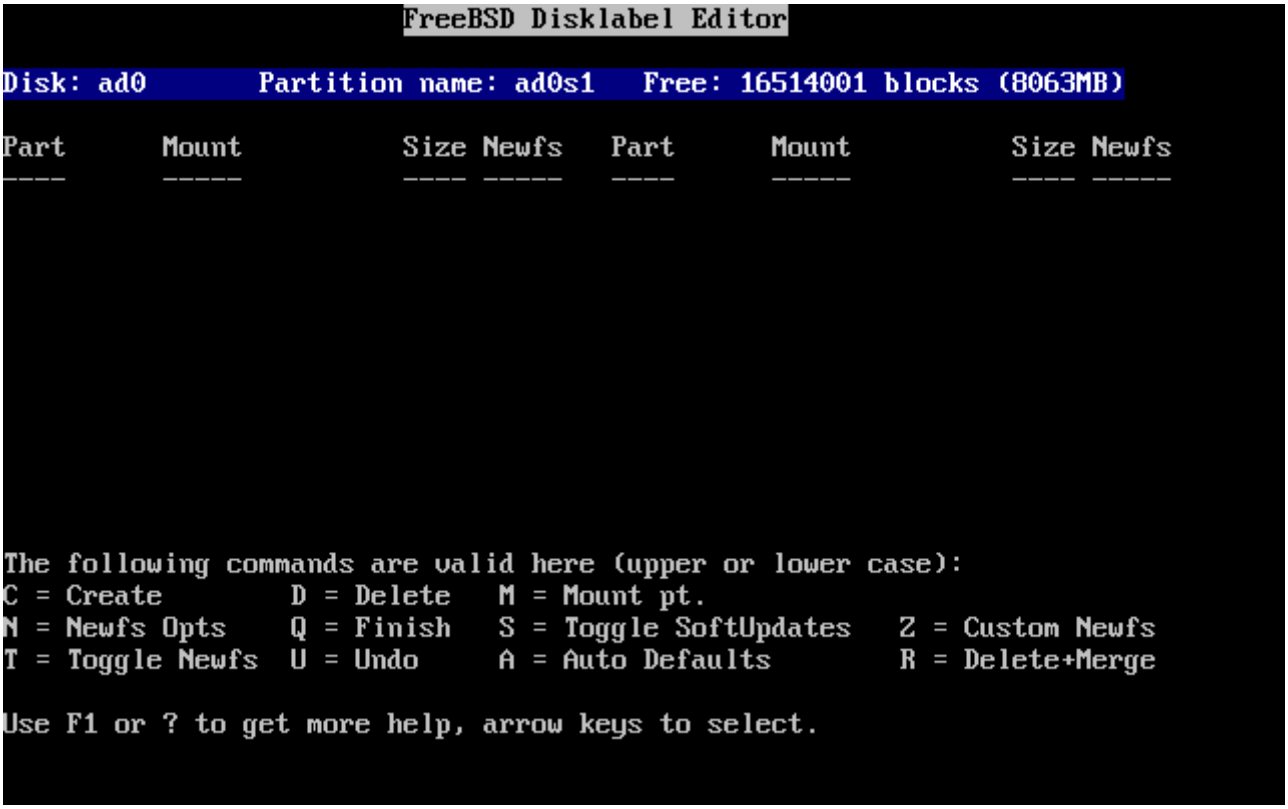
`Enter` товч дарж FreeBSD-н диск хуваалтыг үүсгэж бичдэг Disklabel нэртэй програм харуулагдана.

Sysinstall-н Disklabel буюу дискэнд нэр өгөн засварлагч дээр Disklabel -г анх эхлүүлэх үеийг харуулсан. Дэлгэц гурван хэсэгт хуваагдсан байгаа.

Эхний хэдэн мөрөнд, таны ажиллаж байгаа дискний нэрийг харуулсан бөгөөд мөн хуваагдал агуулсан зүсмэлийг (энд Disklabel зүсмэл гэж нэрлэлгүй харин `Partition name буюу хуваалтын нэр` гэж нэрлэсэн байна) харуулсан байна. Энэ хэсэгт мөн зүсмэлд буй сул хэсгийн хэмжээг харуулдаг бөгөөд уг зураг дээр бол уг хэмжээ нь одоогоор ямар ч хуваалтад хэрэглэгдээгүй байна.

Дэлгэцийн дунд хэсэг үүсгэгдсэн хуваалтуудыг харуулдаг бөгөөд үүнд, хуваалтын агуулж байгаа файл системийн нэр, түүний хэмжээ, мөн файл системд хамаатай нэмэлт сонголтуудыг харуулдаг.

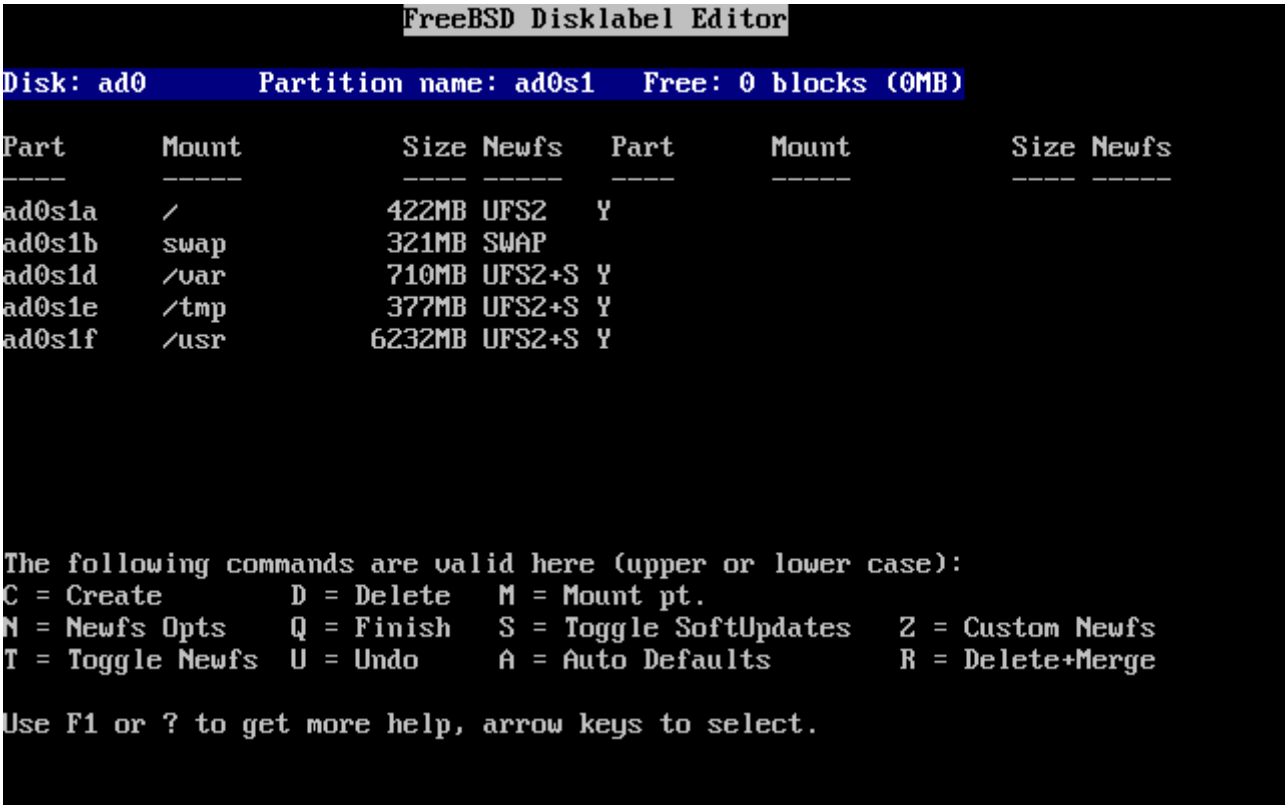
Дэлгэцийн доод гурав дахь хэсэгт Disklabel дээр хэрэглэж болох гарын товчлууруудыг харуулдаг юм.




*Зураг 17. Sysinstall-н Disklabel буюу дискэнд нэр өгөн засварлагч*

Disklabel нь мөн автоматаар хуваалтуудыг үүсгээд анхдагч хэмжээг нь үүсгэж чаддаг. Анхдагч хэмжээ нь хуваалтын хэмжээг тогтоох дотоод алгоритмийн тусламжтайгаар дискийн хэмжээн дээр тулгуурлан тооцоологддог. Үүнийг туршихын тулд `A` товчийг дарж үзээрэй. Тэгвэл танд Sysinstall-н Disklabel хэрэгслийн автомат тохируулалт-тай төстэй зураг харуулагдана. Таны хэрэглэж байгаа дискнээс хамаараад анхны зааж өгсөн хэмжээ нь өөр байж болох юм. Хэрэв та анхны хэмжээг нь хүлээн зөвшөөрч л байвал энэ тийм чухал биш.

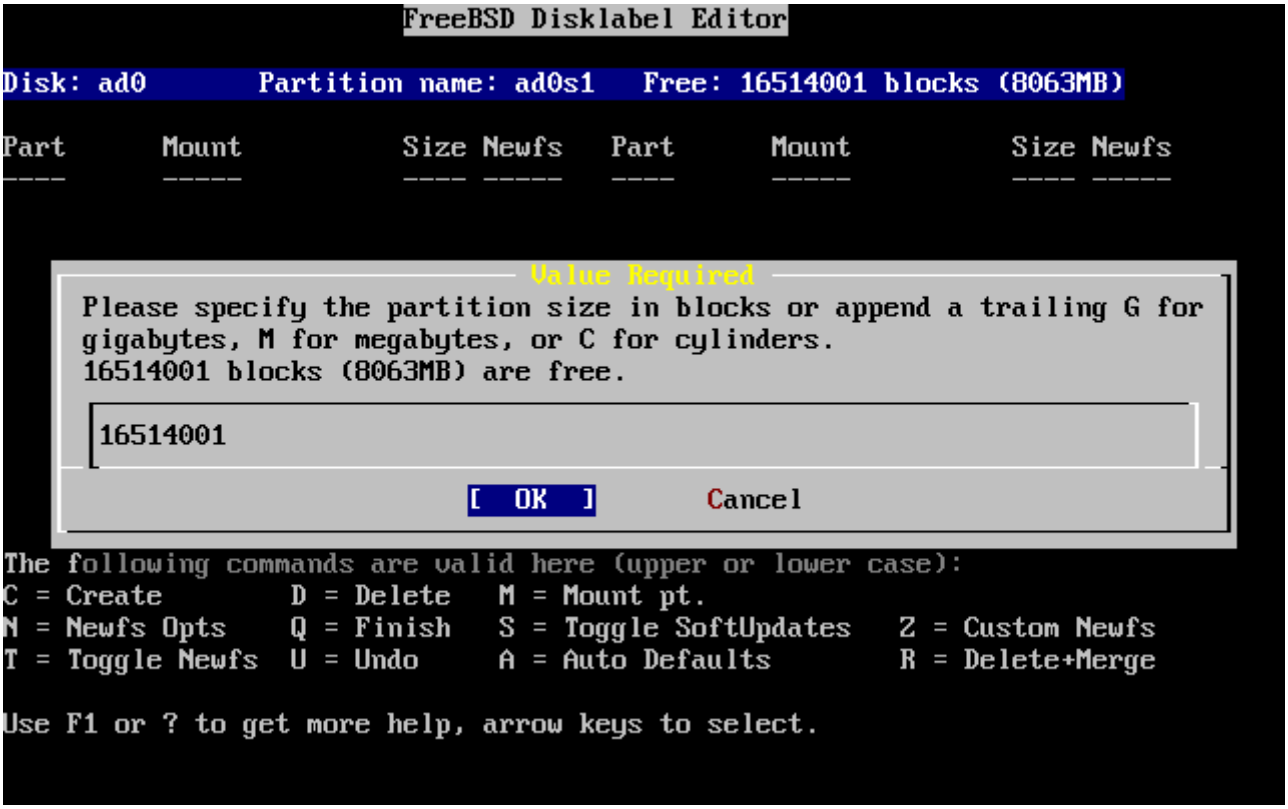
Анхдагч хуваалтад /tmp санг / хуваалтаас тусад нь өөр хуваалтад үүсгэдэг бөгөөд ингэснээр / хуваалтыг түр зуурын файлуудаар түргэн дүүргэхээс сэргийлж өгдөг.




*Зураг 18. Sysinstall-н Disklabel хэрэгслийн автомат тохируулалт*

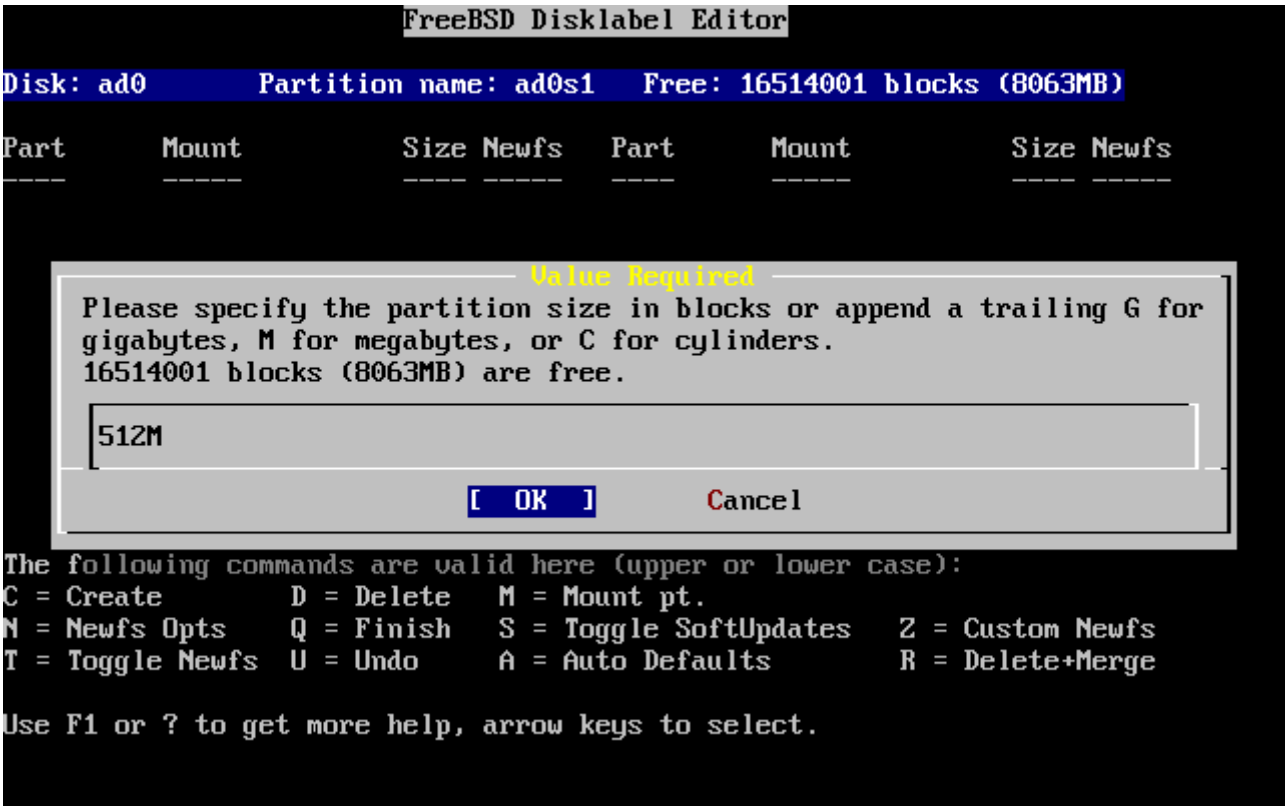
Хэрэв та энэ автомат анхдагч хуваалтыг хүсэлгүй, өөрийнхөөрөө хуваахыг хүсвэл, сумтай товч хэрэглэж сонгоод `D` товч дарж устгаарай. Уг товчийг дахин дахин дарж бүх санал болгосон хуваалтыг устгана.

Эхний хуваалтыг үүсгэхийн тулд (`a` үсэг нь / - root буюу эх файлын систем болдог), дэлгэцийн дээд хэсэгт буй зүсмэлийг сонгож байгаад `C` товч дарах хэрэгтэй. Лавлаж асуух цонх гарч ирэх бөгөөд уг цонхонд шинэ үүсгэх хуваалтын хэмжээг (Root буюу эх хуваалт дээрх сул зай дээр харуулсан шиг) шаарддаг. Та хэрэв хүсвэл, уг талбарт хуваалтын хэмжээг дискний блок хэмжээг, эсвэл тоо оруулаад ард нь `M` үсэг тавьж мегабайтаар, `G` үсэг тавьж гигабайтаар, эсвэл `C` үсэг тавьж цилиндрийн тоогоор илэрхийлж өгч болдог.




*Зураг 19. Root буюу эх хуваалт дээрх сул зай*

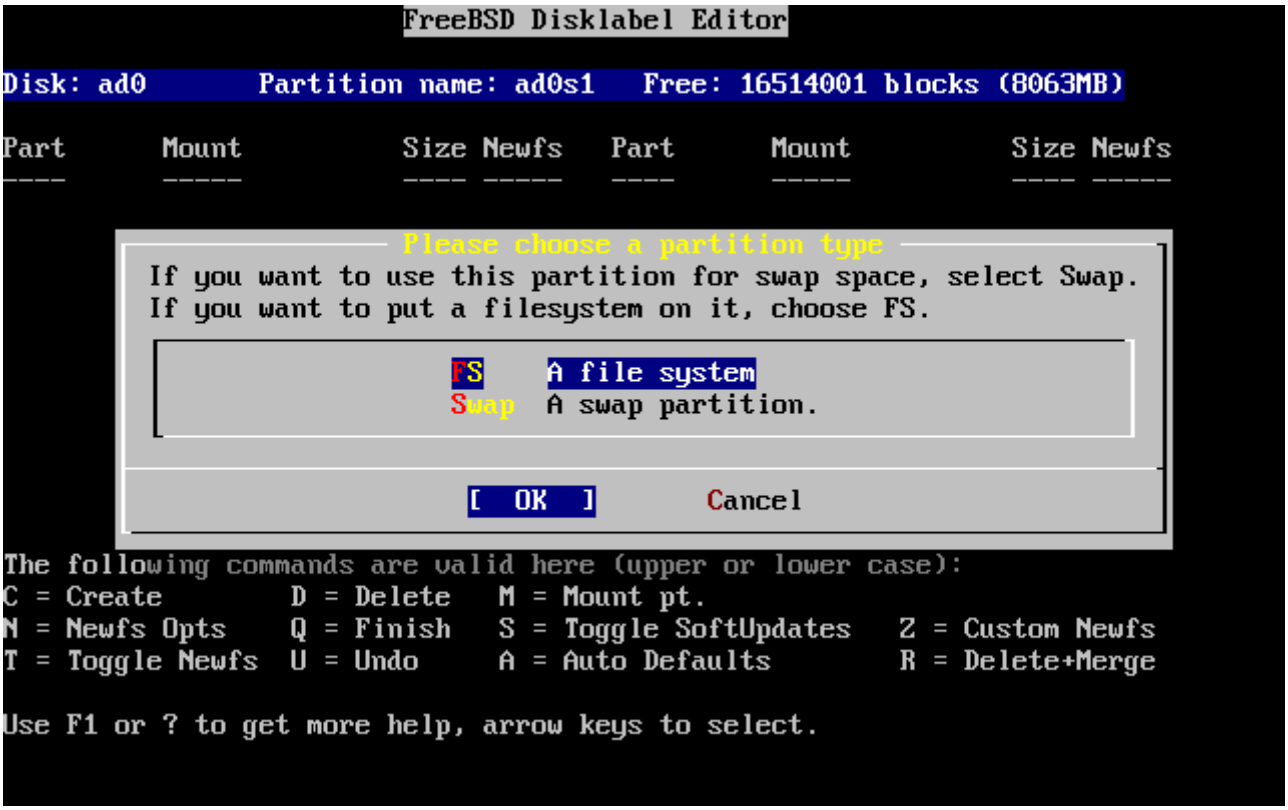
Анхны харуулж байгаа хэмжээ нь зүсмэл дээр үлдсэн хэмжээг зааж харуулдаг. Хэрэв та өмнө харуулсан хуваалтын хэмжээг харж байгаа бол `Backspace` товч дарж устгаад Root partition буюу эх хуваалтын хэмжээ дээр харуулсан шиг `512M` гэж оруулаад дараа нь **[ OK ]** дээр дарах хэрэгтэй.




*Зураг 20. Root partition буюу эх хуваалтын хэмжээ*

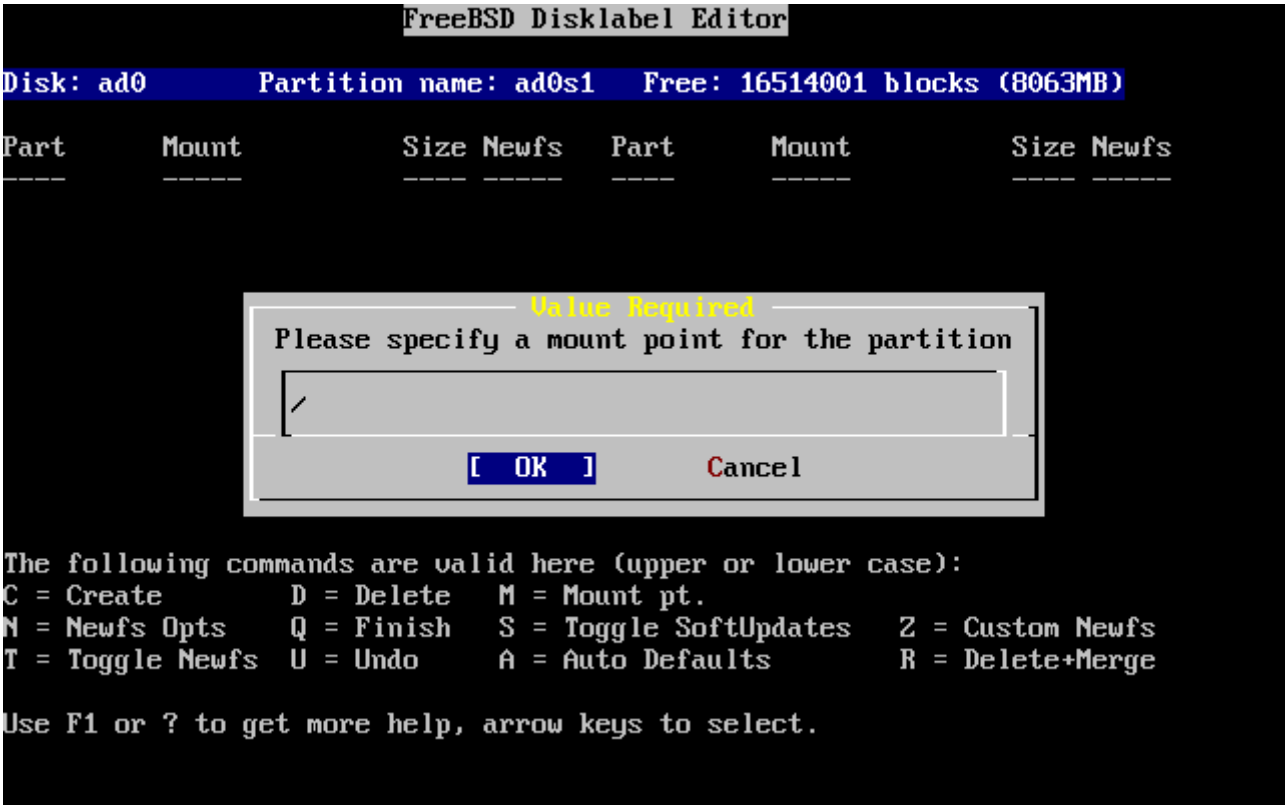
Хуваалтад зориулсан хэмжээг зааж өгсний дараа танаас уг хуваалт дээр файл систем

байрлуулах уу эсвэл swap зай байрлуулах уу гэж асуудаг. Уг асуух цонхыг Root Partition буюу эх хуваалтын төрлийг сонгох дээр харуулав. Эхний хуваалт заавал файл систем байх ёстой учир энэ удаад FS гэсэн сонголтыг сонгоод `Enter` дээр дарах хэрэгтэй.




*Зураг 21. Root Partition буюу эх хуваалтын төрлийг сонгох*

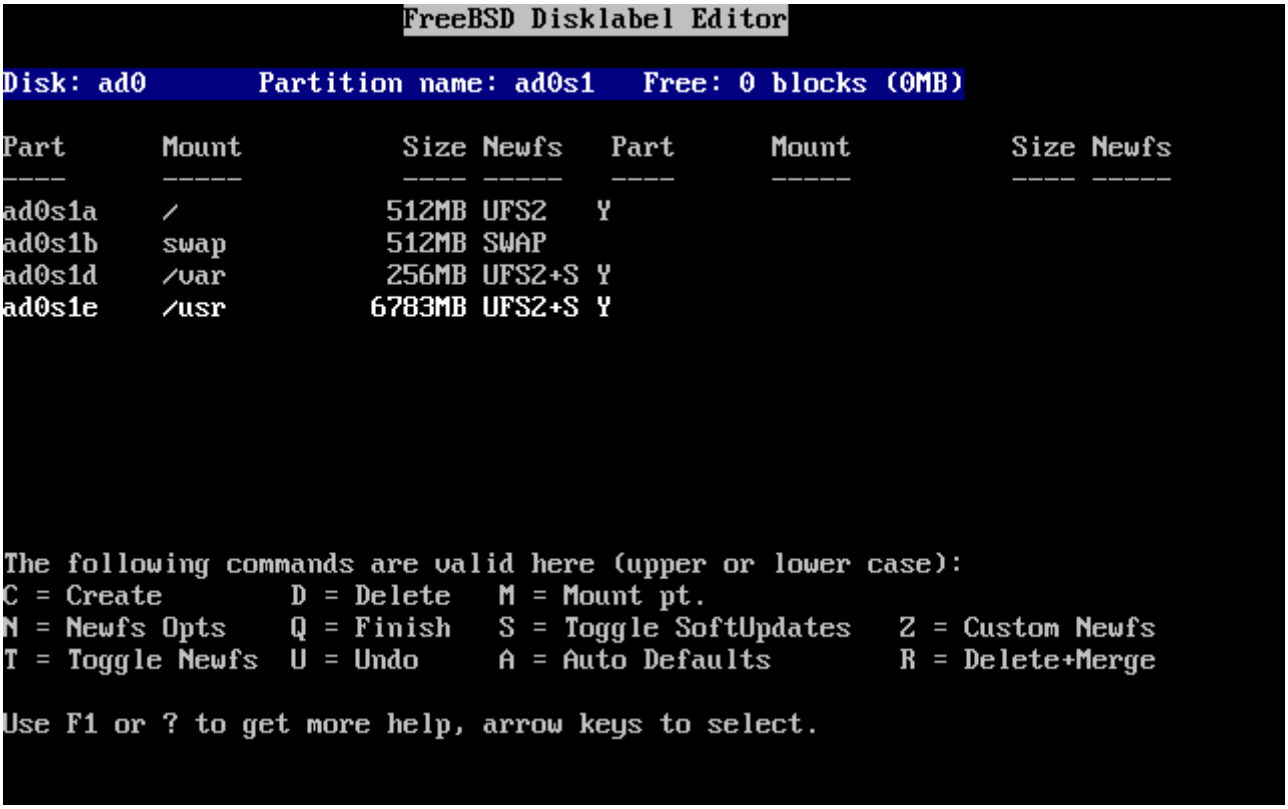
Сүүлд нь, та файл систем үүсгэж байгаа болохоор хаана таниулж холбохыг Disklabel дээр зааж өгөх хэрэгтэй. Уг заалтыг оруулдаг цонхыг Root-г холбох цэг дээр харуулав. root буюу эх файл системийн холбох цэг бол / болохоор та / гэж бичээд `Enter` дараарай.




*Зураг 22. Root-г холбох цэг*

Дэлгэцэн дээр үүсгэсэн хуваалтуудыг шинэчилж харуулах болно. Та өмнө хийсэн үйлдлээ бусад хуваалт дээр хийх хэрэгтэй. Хэрэв та swap хуваалт үүсгэх болонгуут swap хуваалтыг холбох шаардлага байдаггүй учир танаас файл системийн холбох цэг гэж асуухгүй. Сүүлийн хуваалт /usr-г үүсгэх үед санал болгосон хэмжээг өөрчлөлгүй тэр чигээр нь авч хэрэглэснээр зүсмэлийн үлдсэн бүх хэмжээг ашиглаж дуусах нь тэр билээ.

Таны FreeBSD DiskLabel дээрх сүүлчийн харуулалт нь Sysinstall Disklabel засварлагч зурагтай төстэй байх болов уу. Гэхдээ таны сонгосон хэмжээнээс мэдээж өөр байж болно. `Q` товч дарж уг хэсгийг дуусгана.




*Зураг 23. Sysinstall Disklabel засварлагч*

# 2.7. Юу суулгахаа сонгох

## 2.7.1. Суулгах түгээлтийн төрлөө сонгох

Ямар түгээлтийн төрөл суулгах нь системийг ямар зорилгоор хэрэглэх болон дискэнд байгаа сул зайнаас маш их хамаарна. Суулгацад зориулж урьдчилан бэлдсэн суулгах хэмжээ нь суулгаж болох хамгийн бага хэмжээнээс эхлүүлээд бүгдийг суулгах хүртэл боломж өгнө. UNIX® ба/эсвэл FreeBSD системийг шинээр сурч байгаа хүмүүст эдгээр сонголтоос нэгийг нь сонгох хэрэгтэй байх. Урьдчилан бэлдсэн түгээлтийн төрлийг өөрчилж суулгах нь дадлагажсан туршлагатай хэрэглэгчдэд илүүтэй зориулагдсан байдаг.

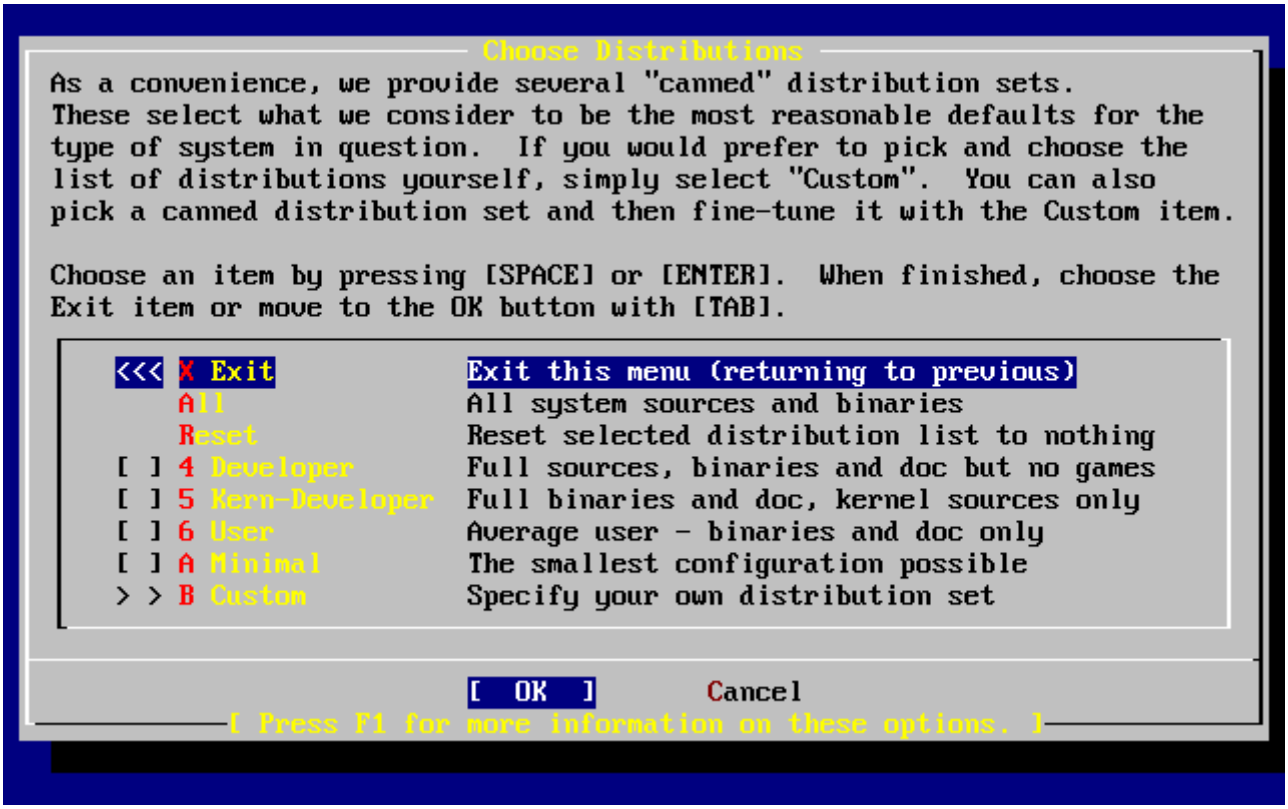
`F1` товчийг түгээлтийн төрөл бүр дээр дарж юу агуулсан болохыг нь харж болно. Тусламж файлыг харсны дараа `Enter` товч дээр дарвал Select Distributions буюу түгээлтийн төрөл сонгох цэсэнд буцаж очдог.

Хэрэв та график горимд ажиллана гэж бодож байвал X серверийн тохиргоо болон анхдагч график горимын орчны сонголтыг FreeBSD-г суулгасны дараа хийх ёстой. X серверийг суулгаж тохируулахтай холбоотой нэмэлт мэдээллийг X Цонхот систем хэсгээс уншаарай.

Хэрэв та цаашдаа өөртөө тохируулсан цөм эмхэтгэж бэлдэнэ гэж бодож байгаа бол source code буюу эх бичлэг агуулсан сонголтыг сонгох хэрэгтэй. Яагаад өөрчилж тохируулсан цөм хэрэгтэй тухай нэмэлт мэдээллийг FreeBSD цөмийг тохируулах нь хэсгээс харна уу.

Мэдээж, элдэв ид шидтэй, уян хатан систем бол юм болгоныг л агуулдаг. Хэрэв хангалттай дискний хэмжээ танд байгаа бол Суулгах түгээлтийн төрлөөс сонгох зураг харуулсны дагуу All гэдгийг сонгоод `Enter` товч дарах хэрэгтэй. Хэрэв танд дискний сул зай тийм чухал бол өөртөө тохирсон түгээлтийн төрлийг нь сонгоорой. Суулгацын дараа бусад түгээлтийн

төрлөөс нэмж болох учраас та төгс сонголт байхгүй байна гэж бүү цухалдаарай.




*Зураг 24. Суулгах түгээлтийн төрлөөс сонгох*

### 2.7.2. Портын цуглуулгыг суулгах

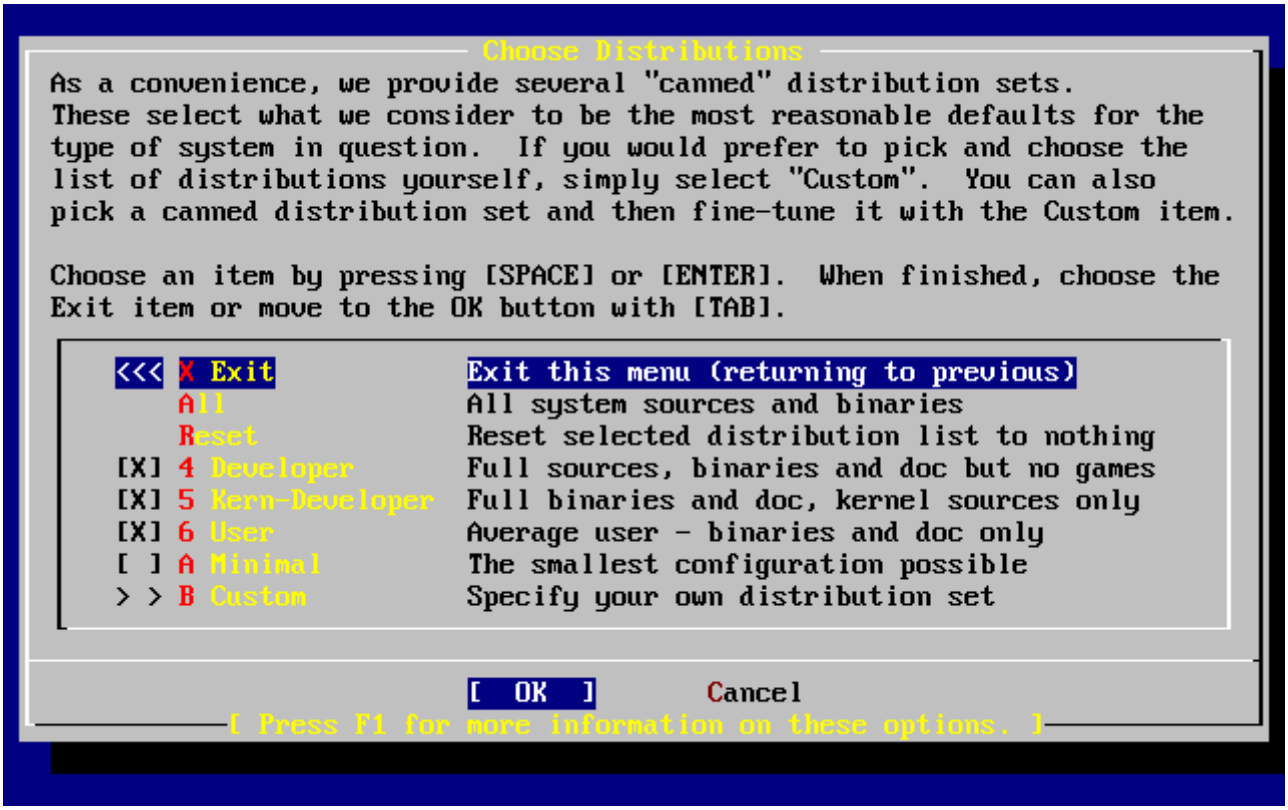
Хүссэн түгээлтийн төрлөө сонгосны дараа FreeBSD-н портын цуглуулгыг суулгах боломж гардаг. Портын цуглуулга гэдэг нь програм суулгах амарчилсан арга юм. Портын цуглуулга нь шаардлагатай програмын эх бичлэгийг агуулдаггүй, харин гуравдагч хөгжүүлэгчдийн бүтээсэн програмыг татаж аваад хөрвүүлж суулгах автоматжуулсан үйлдлүүд байдаг. Програм суулгах. Багцууд болон портууд дээр портын цуглуулгыг хэрэглэх талаар дурьдсан буй.

Порт суулгах програм нь таны дискний зай хангалттай эсэхийг шалгадаггүй. Тийм болохоор дискэнд тань хангалттай зай байгаа тохиолдолд энэ сонголтыг хийх хэрэгтэй. FreeBSD 12.0 хувилбарын байдлаар бол портын цуглуулга нь ойролцоогоор 3 GB хэмжээг дискэн дээр эзэлдэг. FreeBSD-н хувилбар шинэчлэх тутамд энэ хэмжээ ихсэнэ гэж тооцох хэрэгтэй.

```
                        User Confirmation Requested
Would you like to install the FreeBSD ports collection?

This will give you ready access to over 24,000 ported software packages,
at a cost of around 500 MB of disk space when "clean" and possibly much
more than that if a lot of the distribution tarballs are loaded
(unless you have the extra CDs from a FreeBSD CD/DVD distribution
available and can mount it on /cdrom, in which case this is far less
of a problem).

The Ports Collection is a very valuable resource and well worth having
```

```
on your /usr partition, so it is advisable to say Yes to this option.

For more information on the Ports Collection & the latest ports,
visit:
    http://www.FreeBSD.org/ports

                              [ Yes ]     No
```

Портын цуглуулга суулгана гэвэл **[ yes ]** гэдгийг сонгоно, хэрэв суулгахгүй гэвэл **[ no ]** гэдгийг сонгоод `Enter` товч дээр дарж цааш нь үргэлжлүүлнэ. Choose Distributions буюу суулгах түгээлтийн төрөл сонгох цэс дахин гарч ирэх болно.




*Зураг 25. Сонгосон суулгах түгээлтийн төрлөө лавлах*

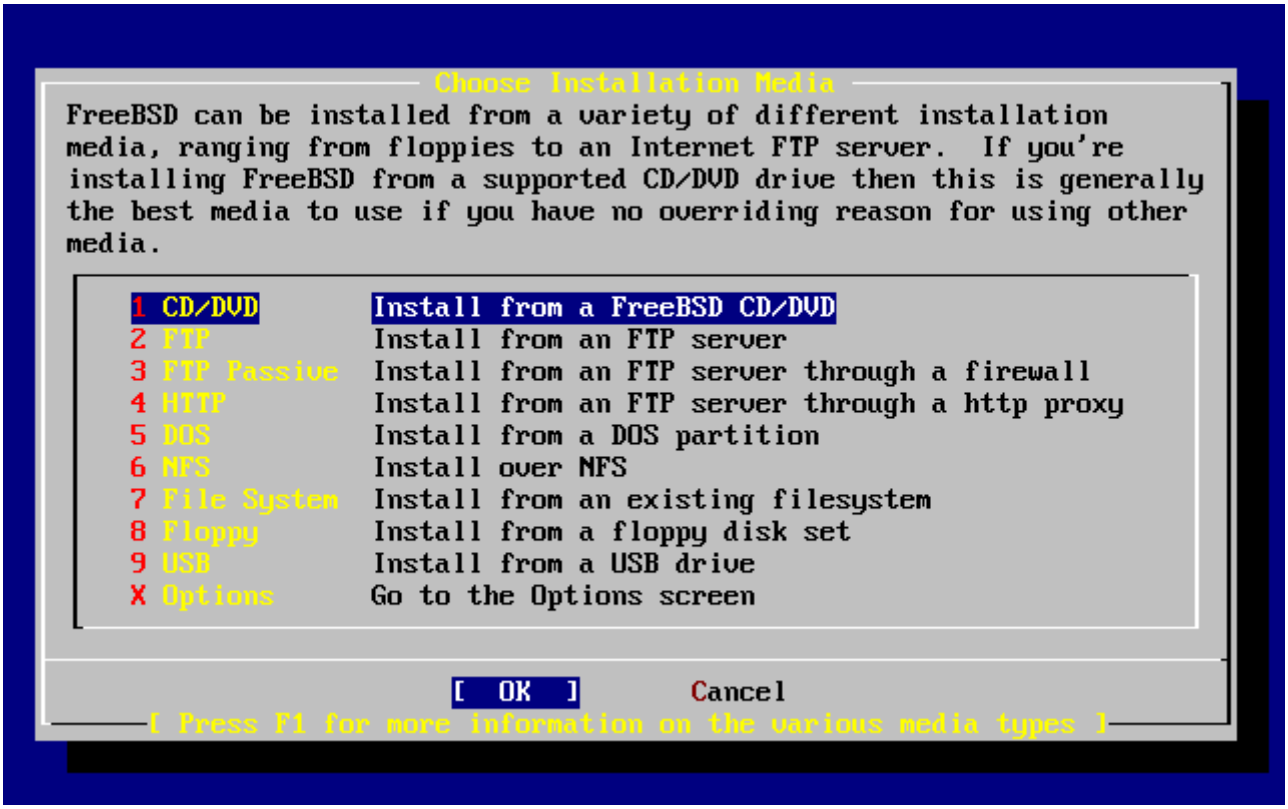
Хэрэв сонгосон түгээлтийн төрөлдөө та сэтгэл хангалуун байгаа бол сумтай товч ашиглан Exit цэсийг сонгоод дараа нь **[ OK ]** сонголт идэвхитэй байх үед `Enter` дээр дарж цааш нь үргэлжлүүлнэ.

## 2.8. Суулгацын төхөөрөмжөө сонгох

Хэрэв CDROM эсвэл DVD-нээс суулгахаар бол сумтай товчийг ашиглан Install from a FreeBSD CD/DVD буюу FreeBSD-г CD/DVD-нээс суулга гэдгийг сонгоно. Дараа нь **[ OK ]** товчийг сонгосны дараа `Enter` товч дарж суулгацыг үргэлжлүүлнэ.

Суулгацын бусад төрлийг сонгохоор бол тухайн тохирсон сонголтыг сонгож харгалзах зааврыг нь дагах хэрэгтэй.

`F1` товч дарж суулгацын төхөөрөмжийн тухай тусламжийг үзэж болно. `Enter` товч дарж тусламжаас гаран суулгацын төхөөрөмж сонгох цэс рүү буцна.




*Зураг 26. Суулгацын төхөөрөмж сонгох*

*FTP суулгацын горим*

Таны сонгож болох гурван төрлийн FTP суулгацын горим бол: active FTP буюу идэвхитэй FTP, passive FTP буюу идэвхгүй FTP, эсвэл HTTP proxy буюу HTTP прокси.

**FTP Active (идэвхитэй): Install from an FTP server (FTP серверээс суулгах)**

Энэ сонголт нь бүх FTP дамжуулалтыг "Active буюу идэвхитэй" горим ашиглаж гүйцэтгэдэг. Энэ холболт нь галт ханаар дамжиж ажиллахгүй бөгөөд харин идэвхгүй горимыг дэмждэг хуучин FTP серверүүдтэй ихэвчлэн ажилладаг. Хэрэв таны холболт идэвхгүй горимд (анхдагч горим) гацаж байвал идэвхитэй болгоод үзэх хэрэгтэй!

**FTP Passive (идэвхгүй): Install from an FTP server through a firewall (галт ханаар дамжиж FTP серверээс суулгах)**

Энэ сонголт нь sysinstall-г бүх FTP үйлдлийг "Passive буюу идэвхгүй" горимд ажиллана гэж тохируулдаг. Энэ нь дурын TCP порт дээр ирж байгаа холболтыг зөвшөөрдөггүй галт ханаар дамжиж ажиллах боломж өгдөг.

**FTP via a HTTP proxy (HTTP проксигоор дамжиж): Install from an FTP server through a http proxy (HTTP проксигоор дамжин FTP серверээс суулгах)**

Энэ сонголт нь sysinstall-г HTTP протокол ашиглан (вэб хөтлөгч шиг) прокситой холбогдож бүх FTP үйлдлийг гүйцэтгэхээр тохируулж өгдөг. Прокси нь ирсэн хүсэлтийг хөрвүүлээд цааш нь FTP сервер уруу дамжуулдаг. Ингэснээр хэрэглэгчид бүх FTP холболтыг хориод HTTP

> холболтыг зөвшөөрсөн галт ханын дундуур дамжих боломж олгодог. Энэ тохиолдолд та FTP серверийг зааж өгөхөөс гадна мөн проксигийн нэрийг зааж өгдөг.
>
> Прокси FTP серверийн тохируулгад жинхэнэ холбогдох серверийнхээ нэрийг хэрэглэгчийн нэрийн хэсэг мэт "@" тэмдгийн араас оруулж өгөх хэрэгтэй. Ингэснээр прокси серверийг жинхэнэ сервер мэт "хуурч" ажиллуулдаг. Жишээ нь та `ftp.FreeBSD.org` гэсэн серверээс суулгах хэрэгтэй бөгөөд 1234 порт дээр буй `foo.example.com` гэсэн FTP проксигоор дамжих ёстой байсан гэж авч үзье.
>
> Энэ тохиолдолд та сонгох цэс рүү очоод, FTP хэрэглэгчийн нэрийг `ftp@ftp.FreeBSD.org` гэж оруулаад нууц үгэнд нь захианыхаа хаягийг бичих хэрэгтэй. Суулгацын төхөөрөмжөө FTP (эсвэл прокси нь дэмждэг бол идэвхгүй FTP) гэж сонгоод URL хаягийг `ftp://foo.example.com:1234/pub/FreeBSD` гэж оруулна.
>
> `ftp.FreeBSD.org`-д буй /pub/FreeBSD хаяг нь `foo.example.com` гэсэн нэрээр дамжигдах бөгөөд та суулгацыг *энэ* машинаас (уг машин таны файлуудыг `ftp.FreeBSD.org` хаягнаас танд зуучилж өгнө) татаж авч өгдөг.

## 2.9. Суулгацыг баталж гүйцээх

Хэрэв хүсвэл, одоо суулгацыг гүйцэтгэж болно. Энэ нь мөн хатуу дискэнд өөрчлөлт оруулахаас сэргийлж цуцалж болох сүүлийн боломж юм.

```
                      User Confirmation Requested
Last Chance! Are you SURE you want to continue the installation?

If you're running this on a disk with data you wish to save then WE
STRONGLY ENCOURAGE YOU TO MAKE PROPER BACKUPS before proceeding!

We can take no responsibility for lost disk contents!

                           [ Yes ]    No
```

**[ yes ]** товчийг сонгоод `Enter` товч дарж суулгацыг гүйцэтгэнэ.

Сонгосон түгээлт, суулгацын төхөөрөмж, компьютерийн хурд зэргээс хамааараад суулгах хугацаа нь янз бүр. Суулгацын үед явцын төлөв байдлыг илэрхийлсэн хэд хэдэн бичиглэл харуулагддаг.

Дараах бичиглэл харуулагдсан үед суулгац гүйцсэн байдаг:

```
                              Message

Congratulations! You now have FreeBSD installed on your system.
```

```
We will now move on to the final configuration questions.
For any option you do not wish to configure, simply select No.

If you wish to re-enter this utility after the system is up, you may
do so by typing: /usr/sbin/sysinstall.

                                [ OK ]

                        [  Press enter or space  ]
```

`Enter` товч дээр дарж суулгацын дараах тохируулгыг хийх шатанд ордог.

**[ no ]** товчийг сонгоод `Enter` товч дарвал суулгац цуцлагдах бөгөөд системд ямар ч өөрчлөлт хийгдэхгүй. Тэгээд дараах мэдээллийг харуулдаг:

```
                                Message
Installation complete with some errors.  You may wish to scroll
through the debugging messages on VTY1 with the scroll-lock feature.
You can also choose "No" at the next prompt and go back into the
installation menus to retry whichever operations have failed.

                                [ OK ]
```

Юу ч суулгаагүй тохиолдолд энэ мэдээлэл харуулагддаг. `Enter` товч дарж суулгацаас гарч болох Суулгацын үндсэн цэсэнд буцаж очдог.

# 2.10. Суулгацын дараах тохиргоо

Амжилттай суулгацын дараа маш олон тохируулгын хэсэг эхэлдэг. Уг тохируулгыг, шинээр FreeBSD-г ачаалахын өмнө хийж болох бөгөөд эсвэл суулгацын дараа `sysinstall`-г ажиллуулан Configure-г сонгож тохиргоог дахин хийж болдог.

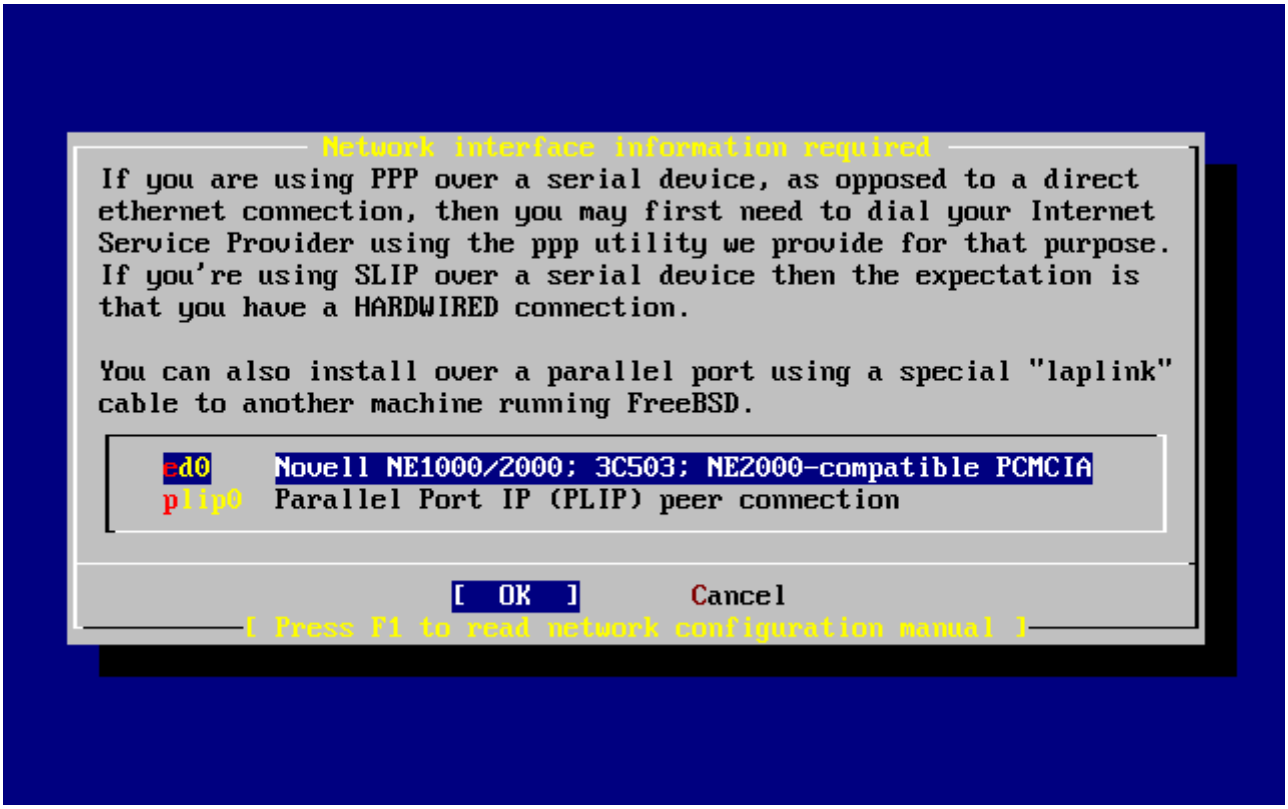
## 2.10.1. Сүлжээний тохиргоо

Хэрэв та өмнө нь FTP суулгац хийхдээ PPP тохируулга хийчихсэн бол, энэ дэлгэц танд харуулагдахгүй бөгөөд хэрэв дахин тохируулах шаардлагатай бол өмнө бичсэний дагуу үйлдэх боломж буй.

Дотоод сүлжээний тухай нарийвчилсан мэдээлэл мөн FreeBSD-г сүлжээний gateway/router буюу хаалга/дамжуулагч (гарц/чиглүүлэгч) хэрхэн болгох талаар Нэмэлт сүлжээ хэсгээс харна уу.

```
                    User Confirmation Requested
   Would you like to configure any Ethernet or PPP network devices?

                            [ Yes ]   No
```

Сүлжээний төхөөрөмжийг тохируулахыг хүсвэл **[ yes ]** гэж сонгоод `Enter` товч дээр дарах хэрэгтэй. Үгүй бол **[ no ]** гэдгийг сонгоод цааш нь үргэлжлүүлээрэй.




*Зураг 27. Сүлжээний төхөөрөмж сонгох нь*

Тохируулах төхөөрөмжөө сумтай товч хэрэглэж сонгоод `Enter` товч дээр дарна.

```
                  User Confirmation Requested
      Do you want to try IPv6 configuration of the interface?

                              Yes   [ No ]
```

Энэ жишээ болгож авсан хувийн локал сүлжээнд одоо сонгосон байгаа интернэт төрлийн протокол нь (IPv4) бүрэн хангалттай болохоор **[ no ]** гэж сонгогдоод `Enter` товч дарагдсан байна.

Хэрэв та өмнө нь байсан IPv6 сүлжээгээр RA сервер уруу холбогдсон байгаа бол **[ yes ]** гэж сонгоод `Enter` дарах хэрэгтэй. Ингэсний дараа хэсэг хугацааны турш RA серверийг хайдаг.

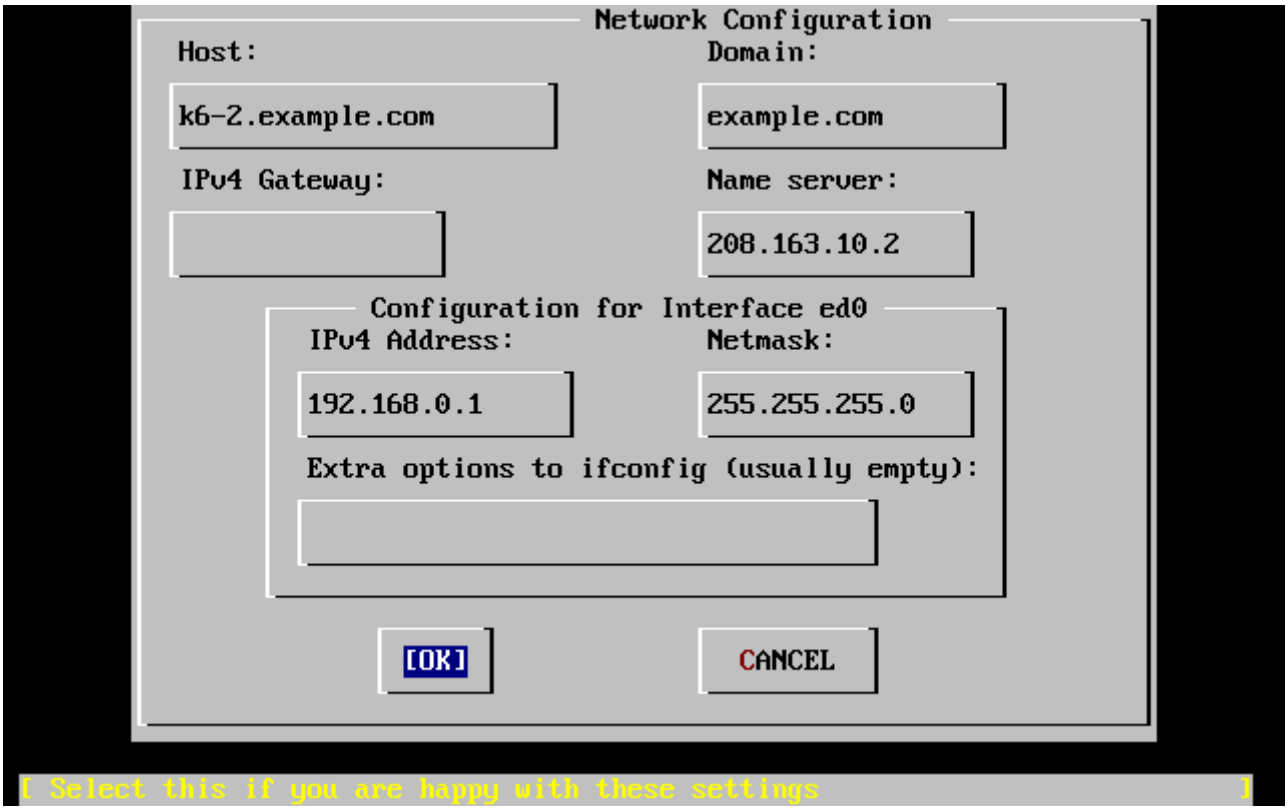
```
                      User Confirmation Requested
      Do you want to try DHCP configuration of the interface?

                              Yes   [ No ]
```

Хэрэв DHCP (Dynamic Host Configuration Protocol буюу Динамикаар компьютерийг тохируулах протокол) шаардлагагүй бол **[ no ]** гэж сонгоод `Enter` дээр дарах хэрэгтэй.

**[ yes ]** гэж сонгосноор dhclient хэрэгслийг ажиллуулдаг бөгөөд хэрэв энэ нь амжилттай

болвол сүлжээний тохиргоо автоматаар хийгдсэн байдаг. Нэмэлт мэдээлэл авахыг хүсвэл Автомат Сүлжээний Тохиргоо (DHCP) хэсгээс харна уу.

Дараах сүлжээг тохируулах цонхонд, уг системийг дотоод сүлжээний gateway буюу хаалга болгон тохируулж байгааг харуулж байна.




*Зураг 28. ed0-д сүлжээний тохиргоог хийх нь*

`Tab` товч хэрэглэн тохиргооны талбаруудад сэлгэж шаардлагатай тохируулгыг нь оруулаарай:

**Host буюу уг компьютерийн нэр**

Уг компьютерийн бүрэн нэр. Жишээ нь энэ тохиолдолд `k6-2.example.com`.

**Domain буюу домэйн**

Таны компьютерийн ашиглаж буй домэйн нэр. Энэ тохиолдолд `example.com` болж байна.

**IPv4 Gateway буюу хаалга**

Дотоод биш компьютер уруу өгөгдөл цааш дамжуулан илгээгч компьютерийн IP хаяг. Хэрэв таны тохируулж байгаа машин тань сүлжээндээ ийм дамжуулагчийн үүрэг гүйцэтгэгч биш, харин тийм дамжуулагчийг ашигладаг бол, энэ талбарт бөглөөрэй. Хэрэв таны машин интернэт уруу оруулдаг gateway буюу хаалганы үүрэг гүйцэтгэдэг бол, энэ талбарыг *хоосон орхих ёстой*. IPv4 Gateway буюу хаалга нь анхдагч хаалга юм уу эсвэл анхдагч route буюу зам заагч гэж нэрлэгддэг.

**Name server буюу Нэрийн сервер**

Дотоод DNS серверийн IP хаяг. Хэрэв дотоод хувийн сүлжээнд DNS сервер гэж байхгүй бол интернэтээр хангагч байгууллагын DNS серверийн хаягийг оруулж өгдөг. (энэ жишээнд `208.163.10.2`).

**IPv4 хаяг**

Энэ сүлжээний төхөөрөмжид өгөгдсөн IP хаяг нь `192.168.0.1` гэж тохируулагдаж байна

**Netmask буюу сүлжээний ангилагч**

Энэ дотоод сүлжээний хаягийн хувьд ашиглагдах хаягийн блок нь `255.255.255.0` бүхий сүлжээний ангилагчтай байх бөгөөд ингэснээр хаяглалт нь `192.168.0.0` - `192.168.255.255` хүртэл байх болно.

**Extra options to ifconfig буюу ifconfig-н нэмэлт тохиргоонууд**

`ifconfig` дээр нэмэгдэх сүлжээний төхөөрөмжтэй холбоотой нэмэлт тохиргоонууд энд бичигдэнэ. Энэ жишээн дээр нэмэх зүйл байхгүй байна.

Хэрэв дууссан бол `Tab` товч ашиглан **[ OK ]** цэсийг сонгоод `Enter` товч дарна.

```
                    User Confirmation Requested
       Would you like to bring the ed0 interface right now?

                           [ Yes ]   No
```

**[ yes ]** гэж сонгоод `Enter` товч дарснаар уг машины сүлжээний холболтыг идэвхжүүлдэг. Нэгэнт машиныг дараа нь дахин ачаалах хэрэг гардаг болохоор энэ холболт нь нэг их ашиглагдаад байдаггүй билээ.

## 2.10.2. Gateway буюу сүлжээний хаалганы тохиргоо

```
                    User Confirmation Requested
      Do you want this machine to function as a network gateway?

                           [ Yes ]    No
```

Хэрэв уг машин нь, дотоод сүлжээнд хаалгачийн үүрэг гүйцэтгэж машинуудын хооронд багц мэдээллүүдийг дамжуулах үүрэгтэй бол **[ yes ]** гэдгийг сонгоод `Enter` дээр дараарай. Хэрэв зөвхөн сүлжээнд холбогдох машин бол **[ no ]** гэж сонгоод `Enter` дээр дарж үргэлжлүүлнэ.

## 2.10.3. Интернэт үйлчилгээнүүдийг тохируулах

```
                    User Confirmation Requested
Do you want to configure inetd and the network services that it provides?

                           Yes   [ No ]
```

Хэрэв **[ no ]** гэж сонговол, telnetd гэх мэт төрөл бүрийн үйлчилгээнүүдийг хорьдог. Ингэсэн үед алсаас холбогдох хэрэглэгч нь telnet мэтийн програм хэрэглэж уг машин уруу холбогдож чадахгүй болно гэсэн үг. Харин дотоод хэрэглэгчид бол гадаад машин уруу telnet-г ашиглан холбогдож чадсаар байх болно.

Эдгээр үйлчилгээнүүдийг суулгацын дараа /etc/inetd.conf файлыг дуртай завсарлагчаараа нээж засварласнаар идэвхжүүлж болдог. Энэ тухай дэлгэрэнгүй мэдээллийг Ерөнхий агуулга хэсгээс харна уу.

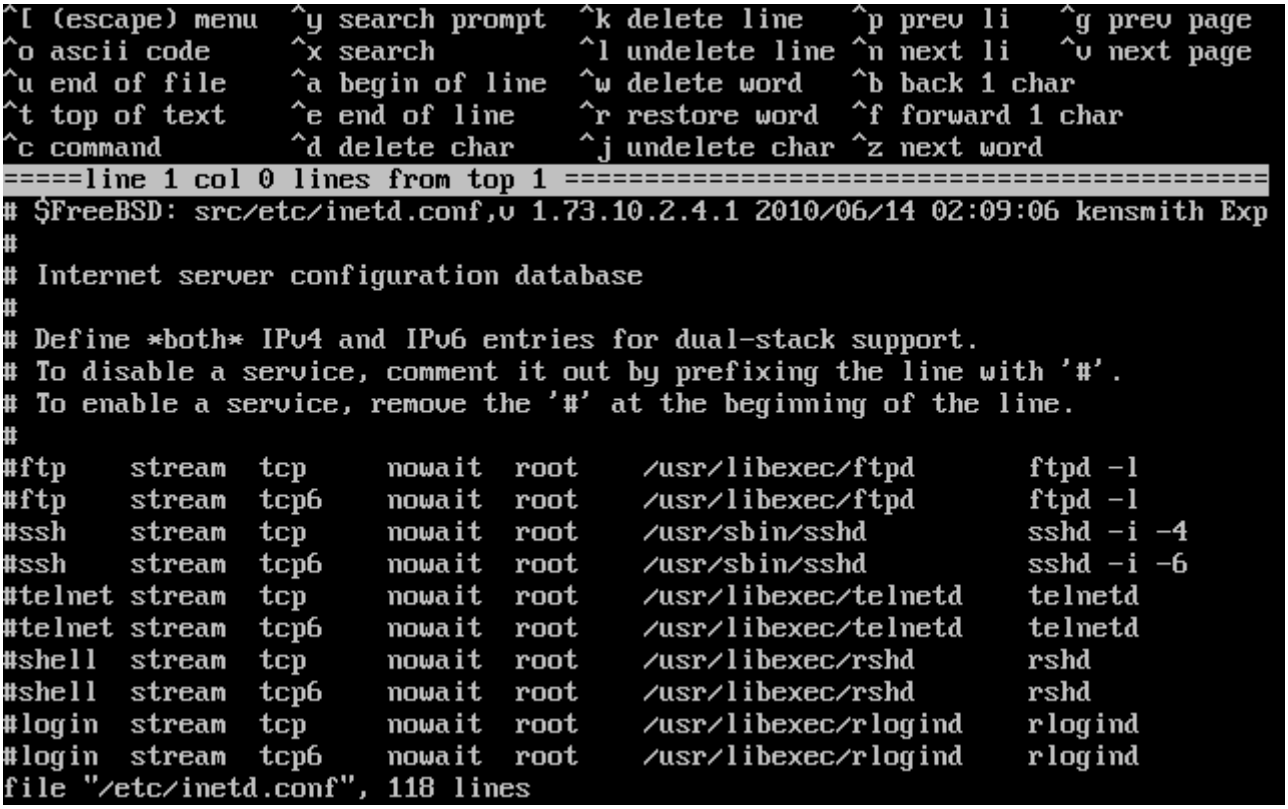
Хэрэв та суулгаж байх явцад эдгээр үйлчилгээнүүдийг тохируулахыг хүсвэл **[ yes ]** гэж сонгоно. Нэмэлт лавлаж асуусан цонх харуулагдах болно:

```
                      User Confirmation Requested
The Internet Super Server (inetd) allows a number of simple Internet
services to be enabled, including finger, ftp and telnetd.  Enabling
these services may increase risk of security problems by increasing
the exposure of your system.

With this in mind, do you wish to enable inetd?

                             [ Yes ]   No
```

**[ yes ]** дээр дарж үргэлжлүүлээрэй.

```
                      User Confirmation Requested
inetd(8) relies on its configuration file, /etc/inetd.conf, to determine
which of its Internet services will be available.  The default FreeBSD
inetd.conf(5) leaves all services disabled by default, so they must be
specifically enabled in the configuration file before they will
function, even once inetd(8) is enabled.  Note that services for
IPv6 must be separately enabled from IPv4 services.

Select [Yes] now to invoke an editor on /etc/inetd.conf, or [No] to
use the current settings.

                             [ Yes ]   No
```

**[ yes ]** гэж сонгосноор тохируулгын файлыг засварлагч файл дээр нээдэг бөгөөд мөрний урд нь буй # тэмдгийг авсанаар тухайн үйлчилгээг идэвхжүүлдэг.




*Зураг 29. inetd.conf файлыг засварлах нь*

Хүссэн үйлчилгээгээ нэмсний дараа `Esc` товч дарвал, хийсэн өөрчлөлтөө хадгалаад гарах сонголттой цэс харуулагддаг.

### 2.10.4. SSH нэвтрэлтийг идэвхжүүлэх нь

```
                    User Confirmation Requested
                 Would you like to enable SSH login?
                       Yes        [  No  ]
```

**[ yes ]**-г сонгосноор OpenSSH-ийн демон програм болох sshd(8)-г идэвхжүүлэх болно. Энэ нь таны машин руу алсаас аюулгүйгээр хандах боломжийг олгоно. OpenSSH-ийн талаар дэлгэрэнгүй мэдээллийг OpenSSH-с үзнэ үү.

### 2.10.5. Anonymous буюу нэр нь үл мэдэгч FTP үйлчилгээ

```
                    User Confirmation Requested
  Do you want to have anonymous FTP access to this machine?

                              Yes    [ No ]
```

#### 2.10.5.1. Anonymous буюу нэр нь үл мэдэгдэгч FTP хэрэглэгчийг хориглох

Анхдагч сонгогдсон байгаа **[ no ]** гэдэг дээр `Enter` дарвал нэр нь үл мэдэгдэгчээр FTP үйлчилгээг хориглодог бөгөөд харин FTP хандах эрхтэй ба нууц үгтэй хэрэглэгчид хандаж болдог.

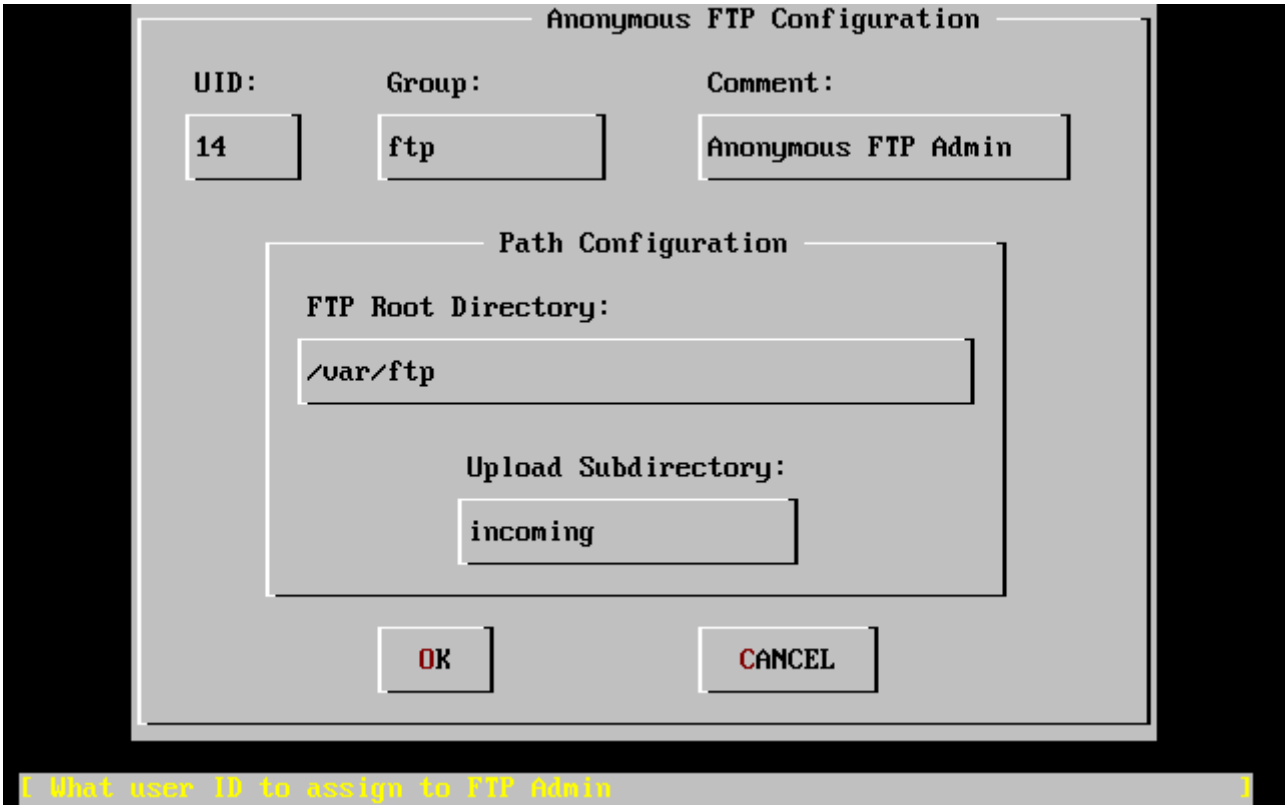
#### 2.10.5.2. Anonymous буюу нэр нь үл мэдэгдэгч FTP хэрэглэгчийг зөвшөөрөх

Энэ тохиолдолд хүн болгон таны машин уруу нэр нь үл мэдэгдэгч болж FTP холболт хэрэглэн хандаж болдог. Нууцлал болон аюулгүй байдлынхаа талаар сайн бодсоны дараа энэ үйлчилгээг хэрэглэх эсэхээ шийдэх хэрэгтэй. Нууцлал болон аюулгүй байдлын талаар Аюулгүй байдал хуудсанд бичсэн буй.

anonymous буюу нэр нь үл мэдэгдэгч FTP хэрэглэгчийг зөвшөөрөхийн тулд, сумтай товч ашиглан **[ yes ]** гэж сонгоод `Enter` дараарай. Нэмэлт батлалт хүлээх дэлгэц харуулагдах болно:

```
                      User Confirmation Requested
Anonymous FTP permits un-authenticated users to connect to the system
FTP server, if FTP service is enabled.  Anonymous users are
restricted to a specific subset of the file system, and the default
configuration provides a drop-box incoming directory to which uploads
are permitted.  You must separately enable both inetd(8), and enable
ftpd(8) in inetd.conf(5) for FTP services to be available.  If you
did not do so earlier, you will have the opportunity to enable inetd(8)
again later.

If you want the server to be read-only you should leave the upload
directory option empty and add the -r command-line option to ftpd(8)
in inetd.conf(5)

Do you wish to continue configuring anonymous FTP?

                          [ Yes ]         No
```

Энэ мэдэгдэл нь хэрэв та нэр нь үл мэдэгдэгч FTP холболтуудыг зөвшөөрөхийг хүсэж байгаа бол FTP үйлчилгээг /etc/inetd.conf файлд бас идэвхжүүлэх ёстойг мэдээлж байна, Интернэт үйлчилгээнүүдийг тохируулах-г үзнэ үү. Үргэлжлүүлэхийн тулд **[ yes ]**-г сонгож `Enter`-г дарна; доор үзүүлсэн дэлгэц гарах болно:




*Зураг 30. Anonymous нэр нь үл мэдэгдэгч FTP үйлчилгээний анхдагч тохиргоо*

Мэдээллийн талбаруудыг сонгохын тулд `Tab` ашиглаж тохирох мэдээллийг оруулах хэрэгтэй:

**UID**

Нэр нь үл мэдэгдэх FTP хэрэглэгчид өгөхийг хүссэн хэрэглэгчийн ID. Хуулагдсан бүх файлуудыг энэ ID эзэмших болно.

**Group**

Нэр нь үл мэдэгдэх FTP хэрэглэгчийг аль бүлэгт байхыг заана.

**Comment**

/etc/passwd файл дахь энэ хэрэглэгчийн тайлбарласан мөр.

**FTP Root Directory**

Нэр нь үл мэдэгдэх FTP-д зориулсан файлууд хаана байхыг заана.

**Upload Subdirectory**

Нэр нь үл мэдэгдэх FTP хэрэглэгчдийн хуулсан файлууд байх сан.

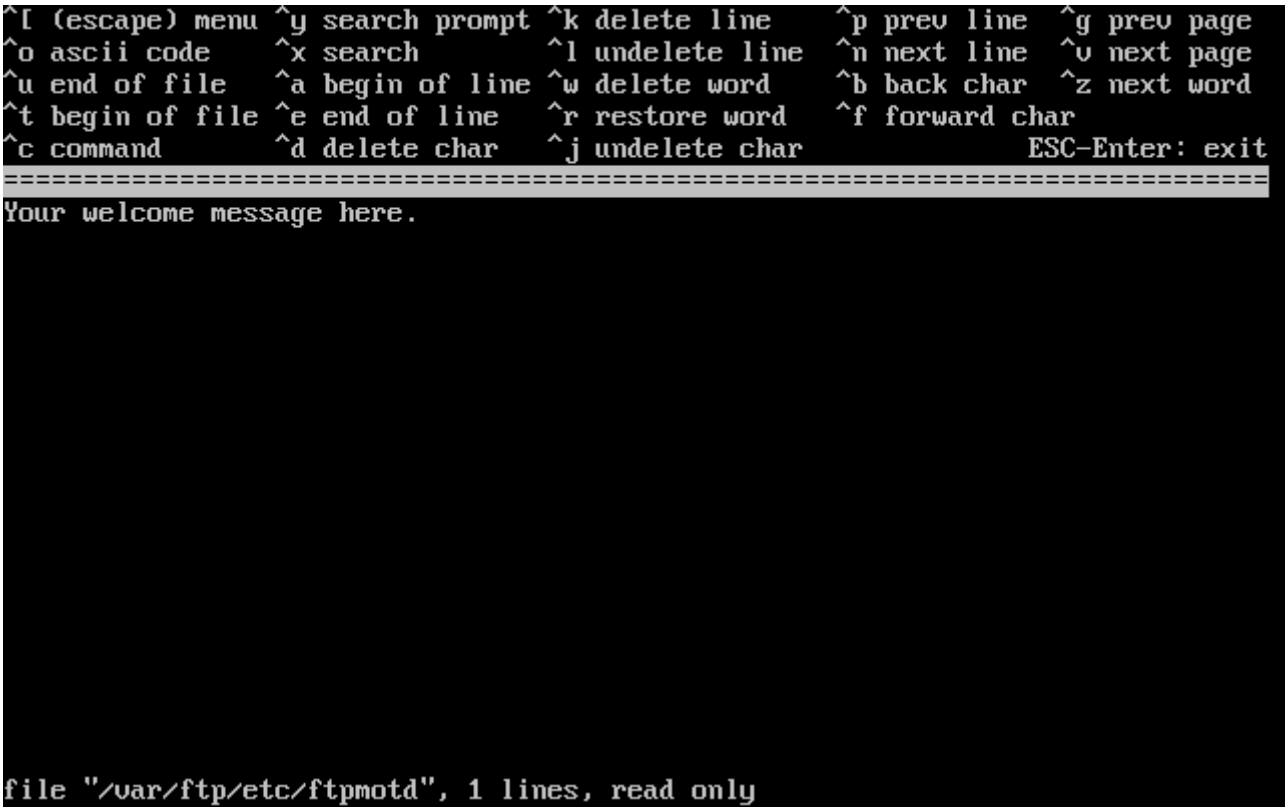
FTP-н root буюу эх сан нь анхдагч тохиргоогоор /var санд байрлагддаг. Хэрэв тэнд хангалттай зай байхгүй бол /usr санг ашиглаж FTP эх сангаа /usr/ftp гэж тохируулж болох юм.

Хэрэв та оруулсан утгуудыг зөв болсон гэж бодож байгаа бол `Enter` дээр дарж үргэлжлүүлээрэй.

```
                    User Confirmation Requested
      Create a welcome message file for anonymous FTP users?
```

```
                              [ Yes ]    No
```

Хэрэв та **[ yes ]** гэж сонгоод `Enter` дээр дарвал, танаас текст оруулуулах зорилгоор засварлагч програм автоматаар ажилладаг.




*Зураг 31. FTP мэндчилгээний бичлэгийг засварлах нь*

Энэ бол `ee` гэгч засварлагч юм. Зааврын дагуу мэндчилгээний захиаг өөрчилж болох бөгөөд, эсвэл дараа нь өөрийн дуртай засварлагчаар нээж уг бичлэгийг өөрчилж бас болно. Тэмдэглэж хэлэхэд, уг файлын нэр болон байрлал нь програмын доод хэсэгт харуулагдаж байгааг анзаарна уу.

`Esc` дээр дарвал, цэстэй жижиг цонх үүсэх бөгөөд түүн дотор a) leave editor буюу засварлагчаас гарах цэс анхдагчаар сонгогдсон байдаг. `Enter` дарж гараад үргэлжлүүлж болно. Эсвэл `Enter` дахин дарж оруулсан өөрчлөлтүүдээ хадгалдаг.

## 2.10.6. Network File System буюу сүлжээний файл системийг тохируулах

Network File System (NFS) нь сүлжээнд буй файлуудыг хуваалцаж хэрэглэх боломж олгодог. Машин нь сервер, хэрэглэгч эсвэл хоёуланг нь болж тохируулагдаж болдог. Сүлжээний Файлын Систем (NFS) хэсгээс нэмэлт мэдээллийг харна уу.

### 2.10.6.1. NFS сервер

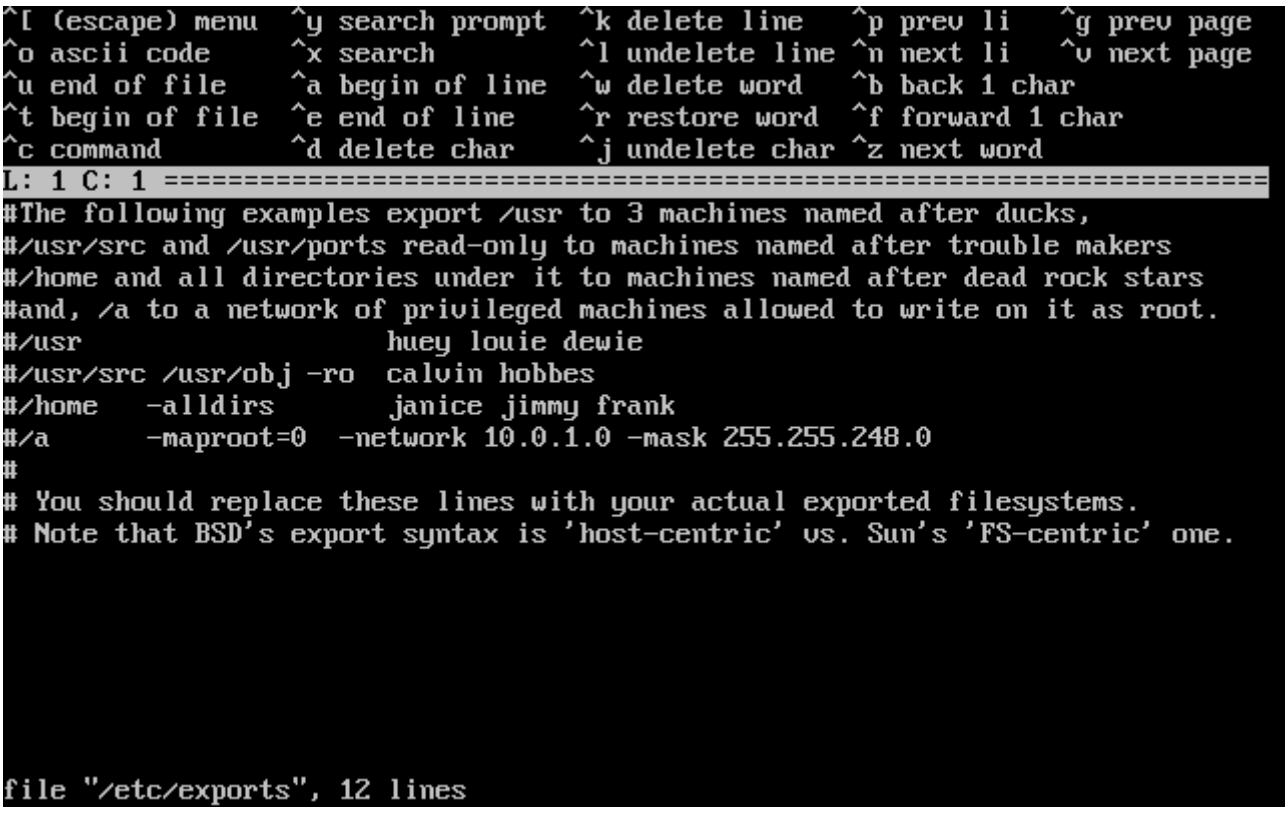
```
                      User Configuration Requested
  Do you want to configure this machine as an NFS server?

                              Yes    [ No ]
```

Хэрэв танд сүлжээний файл системийн сервер шаардлагагүй бол **[ no ]** гэж сонгоод `Enter` дараарай.

Хэрэв сервер хэрэгтэй гэвэл **[ yes ]** гэж сонгох хэрэгтэй бөгөөд танд жижиг цонхон дээр exports файл үүсгэгдэх ёстой гэсэн сануулга гарч ирдэг.

```
                              Message
Operating as an NFS server means that you must first configure an
/etc/exports file to indicate which hosts are allowed certain kinds of
access to your local filesystems.
Press [Enter] now to invoke an editor on /etc/exports
                              [ OK ]
```

`Enter` дарж үргэлжлүүлдэг. Текст засварлагч ажиллаж, түүнд exports файлуудыг үүсгэх ба засварлах үйлдлүүдийг гүйцэтгэдэг.




*Зураг 32. exports файлыг засварлах нь*

Заавар ашиглан, одоо байгаа файл системийг нэмэх эсвэл сүүлд дуртай засварлагчаараа нээж засварлаж болох юм. Файлын нэр болон байршлыг дэлгэцийн доод хэсэгт харуулсан байгааг анзаарна уу.

`Esc` дарахад цэстэй цонх үүсэх бөгөөд a) leave editor буюу засварлагчийг орхи цэс анхлан сонгогдсон байдаг. `Enter` дарж гараад цааш нь үргэлжлүүлнэ.

#### 2.10.6.2. NFS Client буюу хэрэглэгч

NFS хэрэглэгчид нь NFS сервер уруу холбогддог.

```
                    User Confirmation Requested
 Do you want to configure this machine as an NFS client?

                              Yes   [ No ]
```

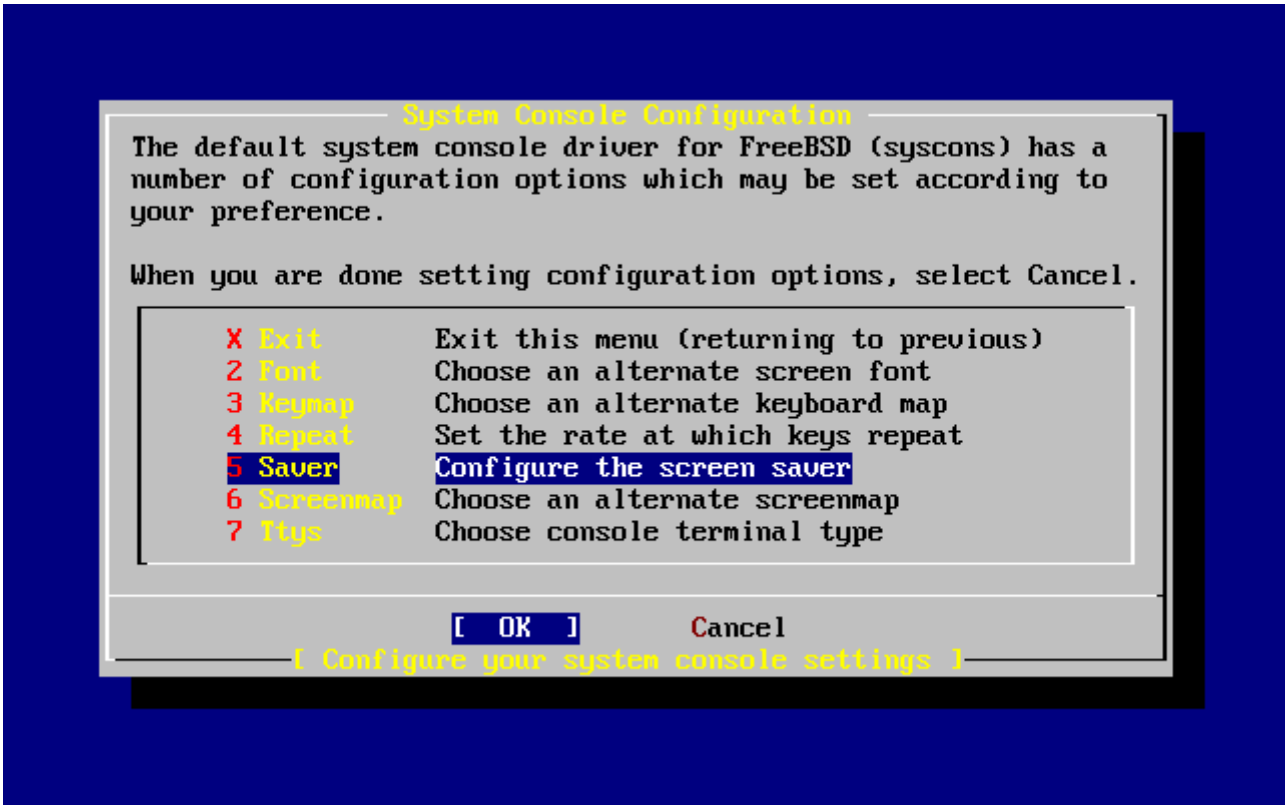
Сумтай товч ашиглан **[ yes ]** эсвэл **[ no ]** сонголтыг шийдвэрээсээ шалтгаалан сонгоод `Enter` дээр дараарай.

### 2.10.7. Системийн консол тохиргоо

Системийнхээ консолыг өөрчлөх хэд хэдэн тохиргоо байдаг.

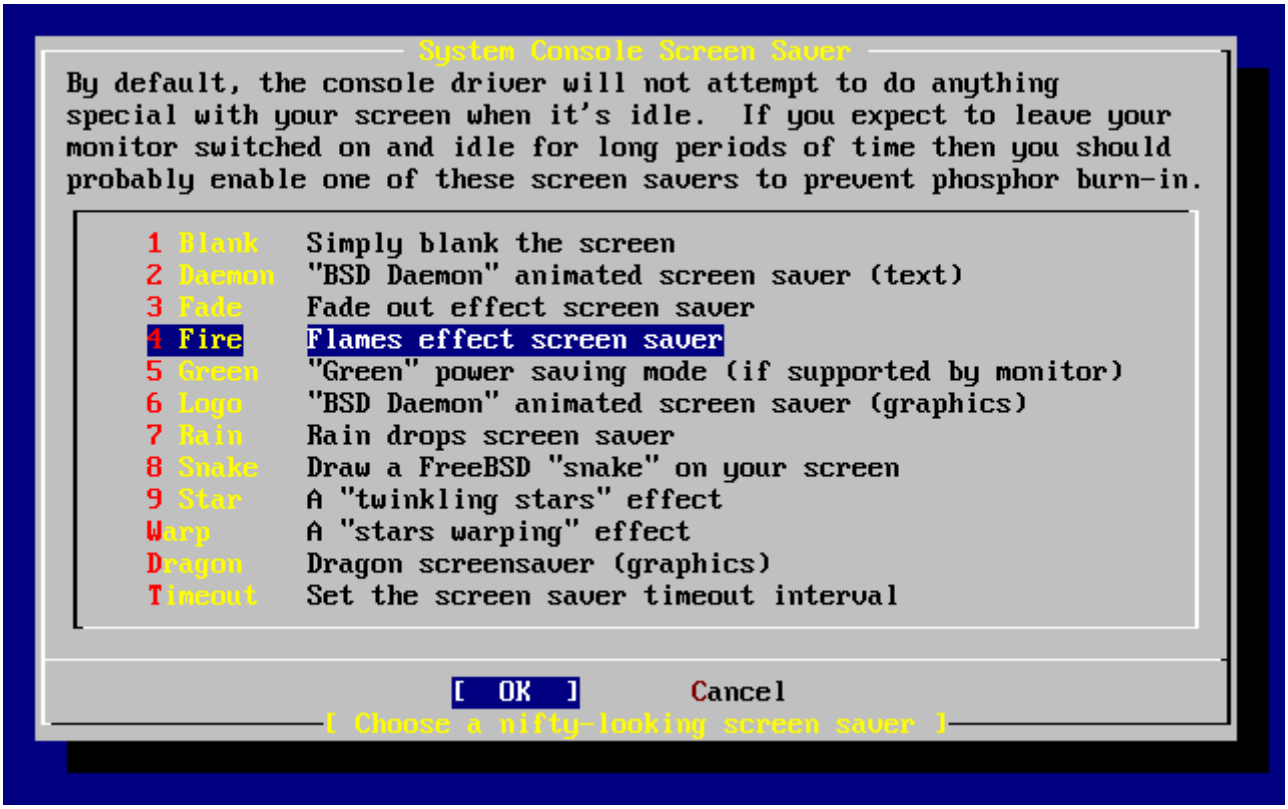
```
                    User Confirmation Requested
       Would you like to customize your system console settings?

                              [ Yes ]  No
```

Тохиргооны сонголтуудыг харах юм уу тохируулахын тулд **[ yes ]** гэж сонгоод `Enter` дарна.




*Зураг 33. Системийн консолын тохируулгын сонголтууд*

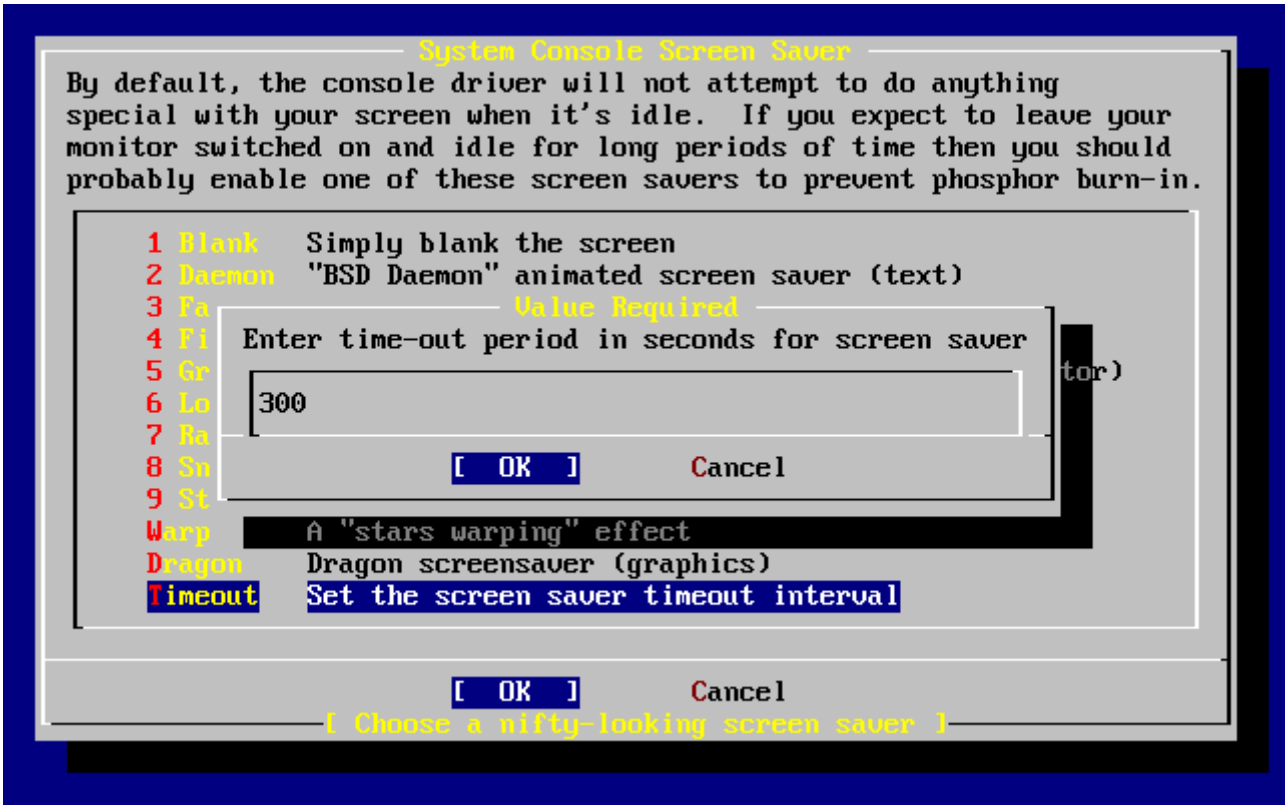
Дэлгэц амраах тохиргоо байнга хэрэглэгддэг. Сумтай товчоор Saver гэж сонгоод `Enter` дараарай.




*Зураг 34. Дэлгэц амраах тохиргоо*

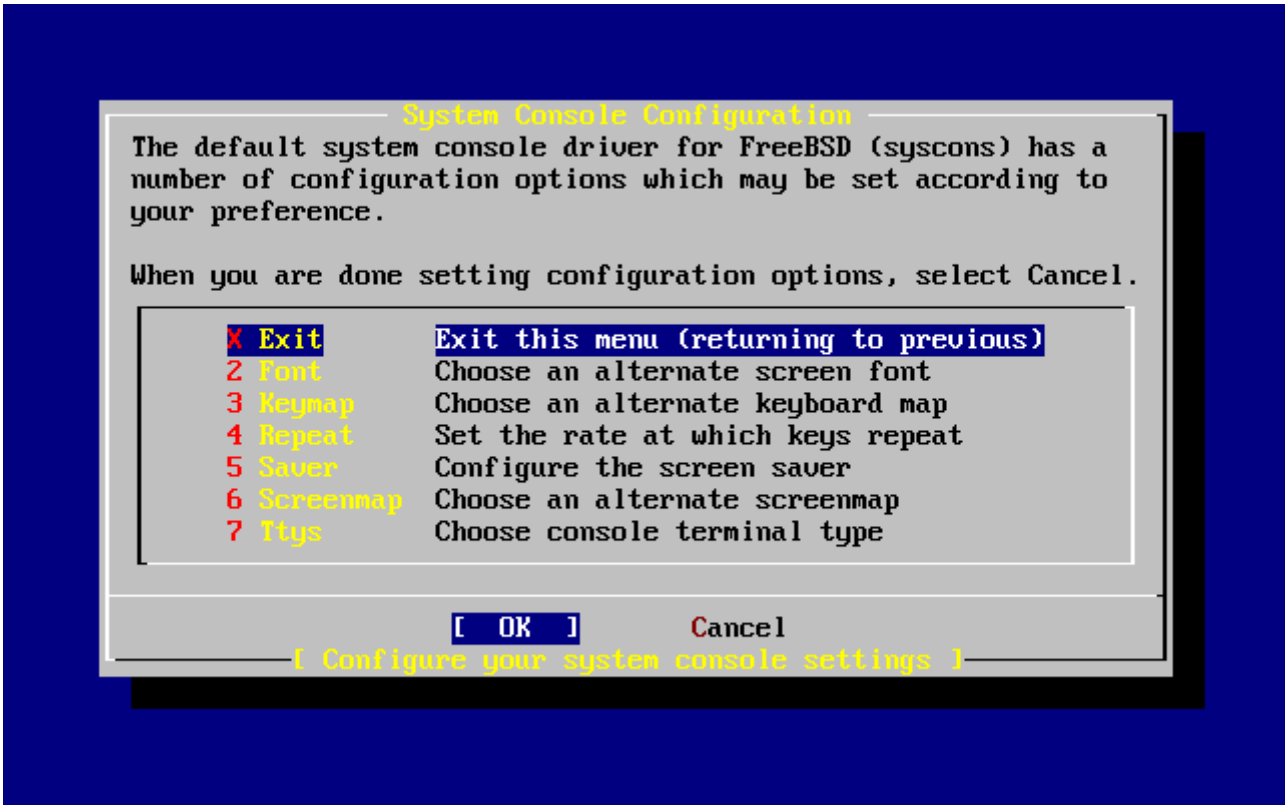
Сум товч ашиглаж хүссэн дэлгэц амраах хөтөлбөрөө сонгоод `Enter` дарна. Системийн консол тохируулах цонх буцаж гарч ирдэг.

Дэлгэц амраах хөтөлбөрийн ажиллах анхдагч хугацаа нь 300 секунд байдаг. Цагийн энэ хугацааг өөрчлөхийнх тулд Saver гэдгийг дахин сонгоно. Түүн дотор байгаа сонголтуудаас Timeout гэдгийг сумтай товч ашиглаж сонгоод `Enter` дээр дарна. Ингэхэд танд жижиг цонх харуулагддаг:




*Зураг 35. Дэлгэц амраагчийн ажиллах хугацаа*

Утгыг өөрчлөөд **[ OK ]** гэж сонгон `Enter` дээр дарж буцаад системийн консолын тохиргооны цэсэнд очдог.




*Зураг 36. Системийн консолын тохиргооноос гарах*

Exit гэдгийг сонгоод `Enter` дээр дарвал суулгацын дараах тохиргоогоо үргэлжлүүлэх хэсэгт очино.

### 2.10.8. Цагийн бүсийг тохируулах

Машиныхаа цагийн бүсийг тохируулснаар тухайн бүсийн цагтай холбогдолтой өөрчлөлтүүд автоматаар хийгдэх давуу талтай байдаг.

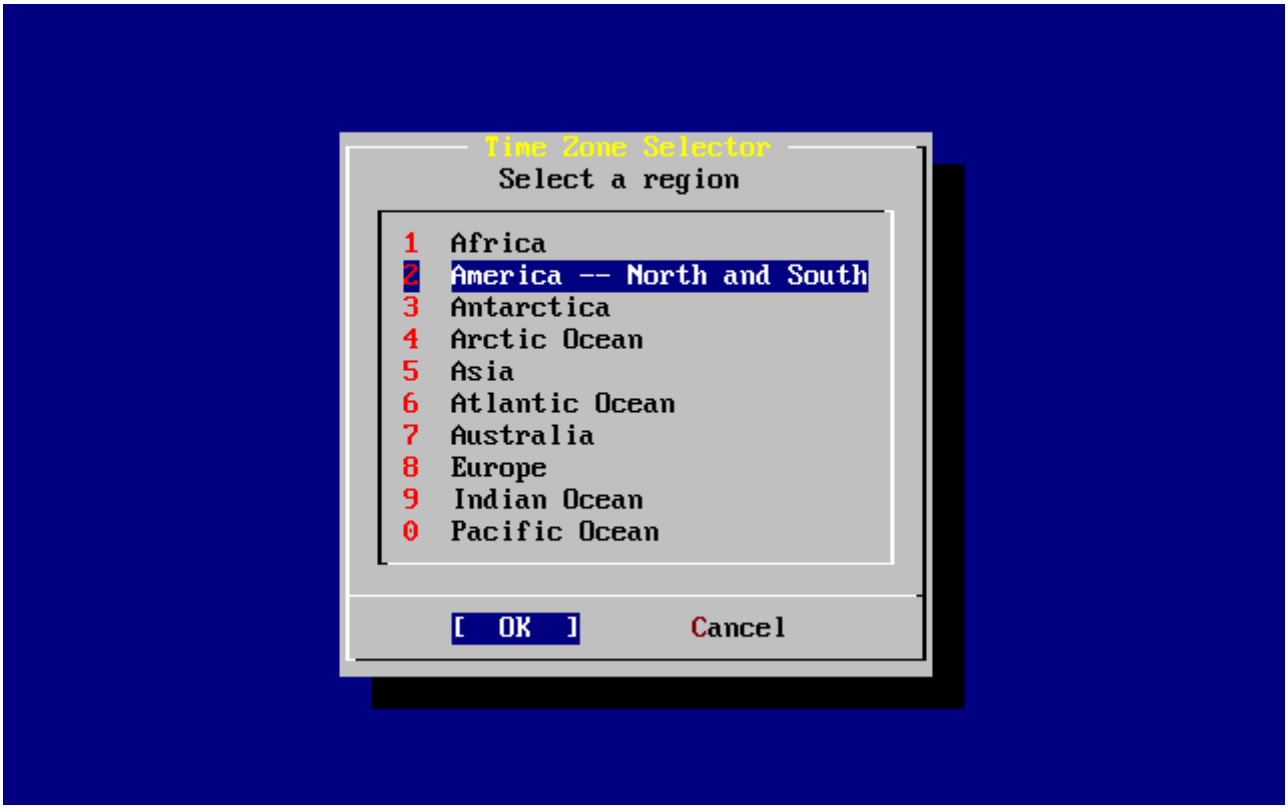
Жишээн дээр, Америкийн Нэгдсэн Улсын Eastern буюу зүүн хэсгийн цагийн бүсийг тохируулж байна. Таны тохируулга байгаа газраасаа шалтгаалж өөр байх болно.

```
                  User Confirmation Requested
        Would you like to set this machine's time zone now?

                           [ Yes ]   No
```

**[ yes ]** гэж сонгоод `Enter` дарвал цагийн бүсийг тохируулах болно.

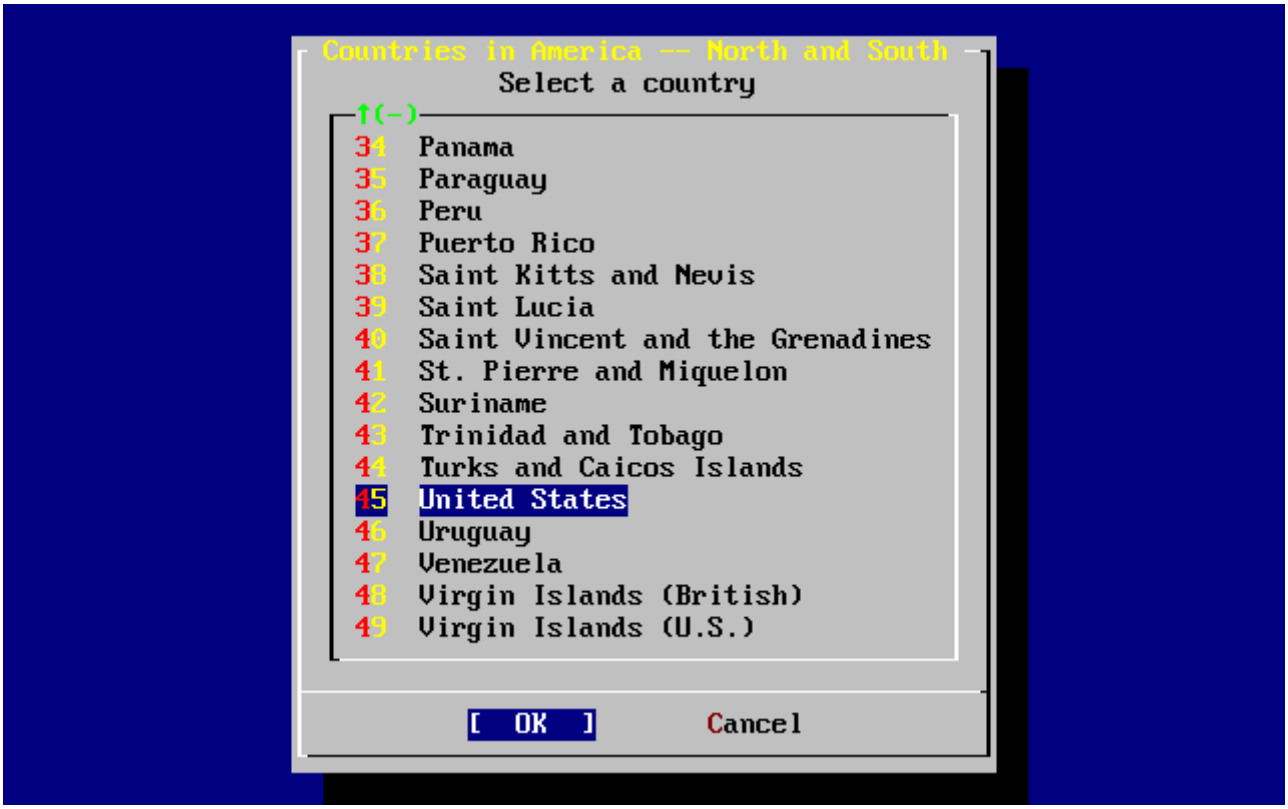
```
                    User Confirmation Requested
Is this machine's CMOS clock set to UTC? If it is set to local time
or you don't know, please choose NO here!

                              Yes   [ No ]
```

Машиныхаа цагийн тохируулгаасаа хамаарч **[ yes ]** эсвэл **[ no ]** гэдгийг сонгоод `Enter` дарна ( мэдэхгүй бол **[ no ]** гэдгийг сонгоорой ) .



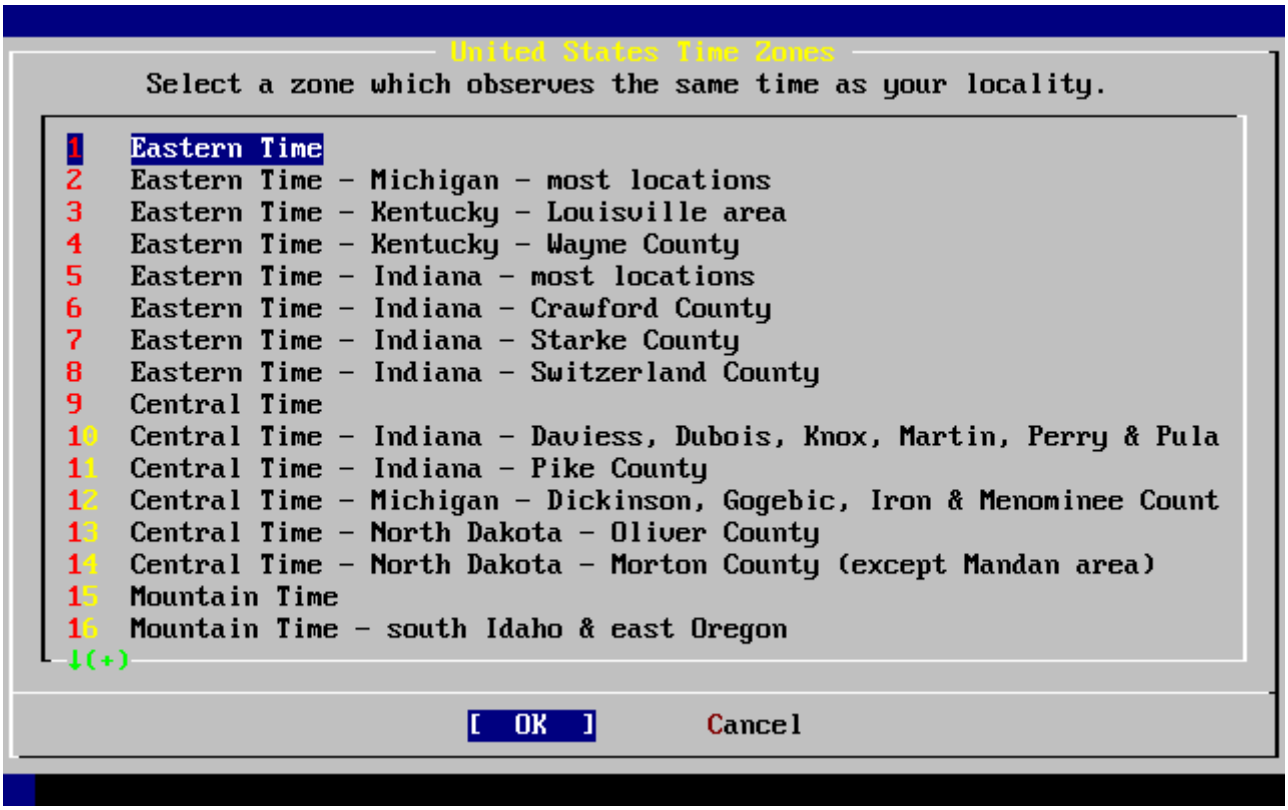

*Зураг 37. Оршин буй бүсээ сонгох*

Тохирсон бүсээ сумтай товч хэрэглэж сонгоод `Enter` дээр дарна.




*Зураг 38. Оршин буй улсаа сонгох*

Тохирох улсаа сонгоод `Enter` дээр дарна.




*Зураг 39. Цагийн бүсээ сонгох*

Сумтай товч хэрэглэж тохирсон цагийн бүсээ сонгоод `Enter` дээр дарна.

```
                    Confirmation
        Does the abbreviation 'EDT' look reasonable?
```

```
                        [ Yes ]   No
```

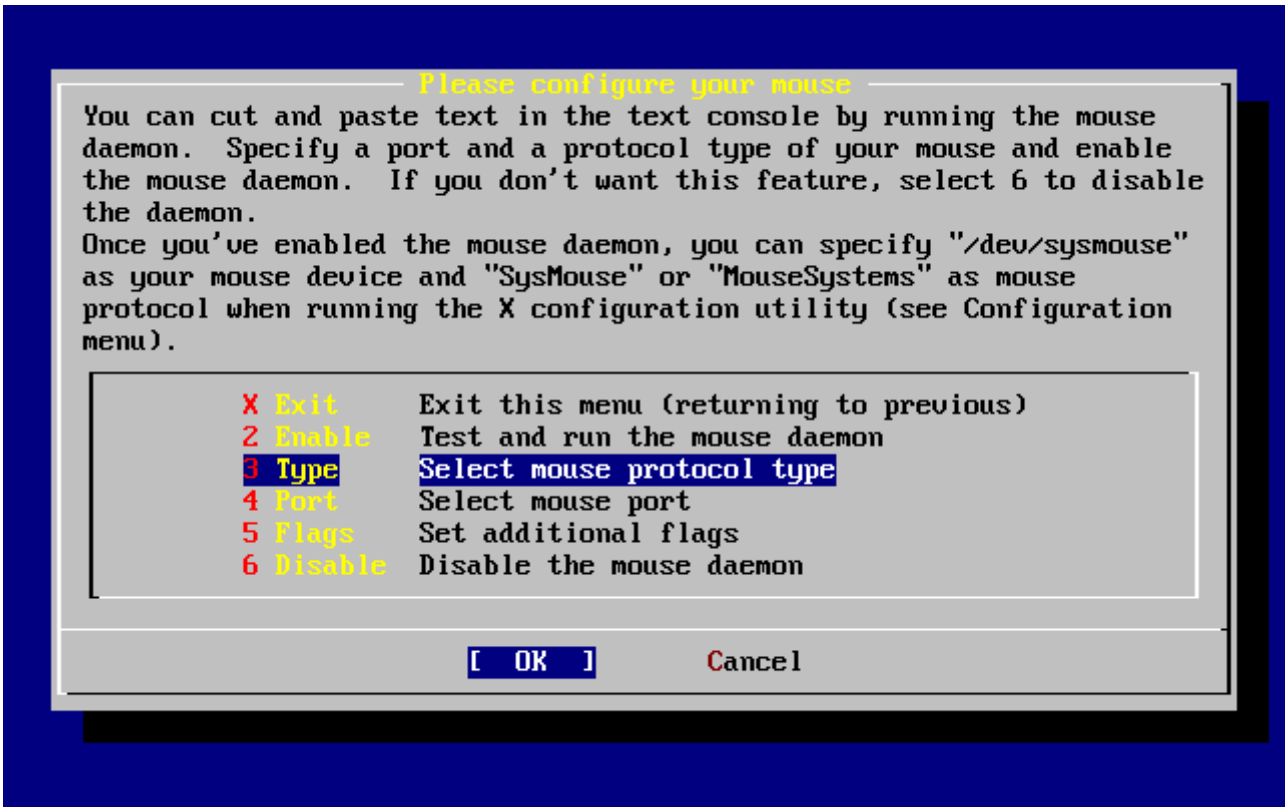
Цагийн бүсийн товчлол нэрийг лавлаж асуух хэсэг гарч ирдэг. Хэрэв уг товчлол нь танд тохирч байвал `Enter` дарж суулгацын дараах тохиргоог цааш нь үргэлжлүүлээрэй.

### 2.10.9. Хулганы тохиргоо

Энэ тохиргоо нь таныг 3 товчтой хулганаар, програм болон консолд текст тасдаж сануулах болон буулгах боломж өгдөг. Хэрэв 2 товчит хулгана хэрэглэж байгаа бол moused(8) гарын авлагаас лавлаж хэрхэн гурван товчтой хулгана болгон ажиллуулж болдгийг харж болно. Энэ жишээн дээр USB биш төрлийн хулганы тохируулгыг харуулж байна (PS/2 эсвэл COM портоор холбогддог хулгана):

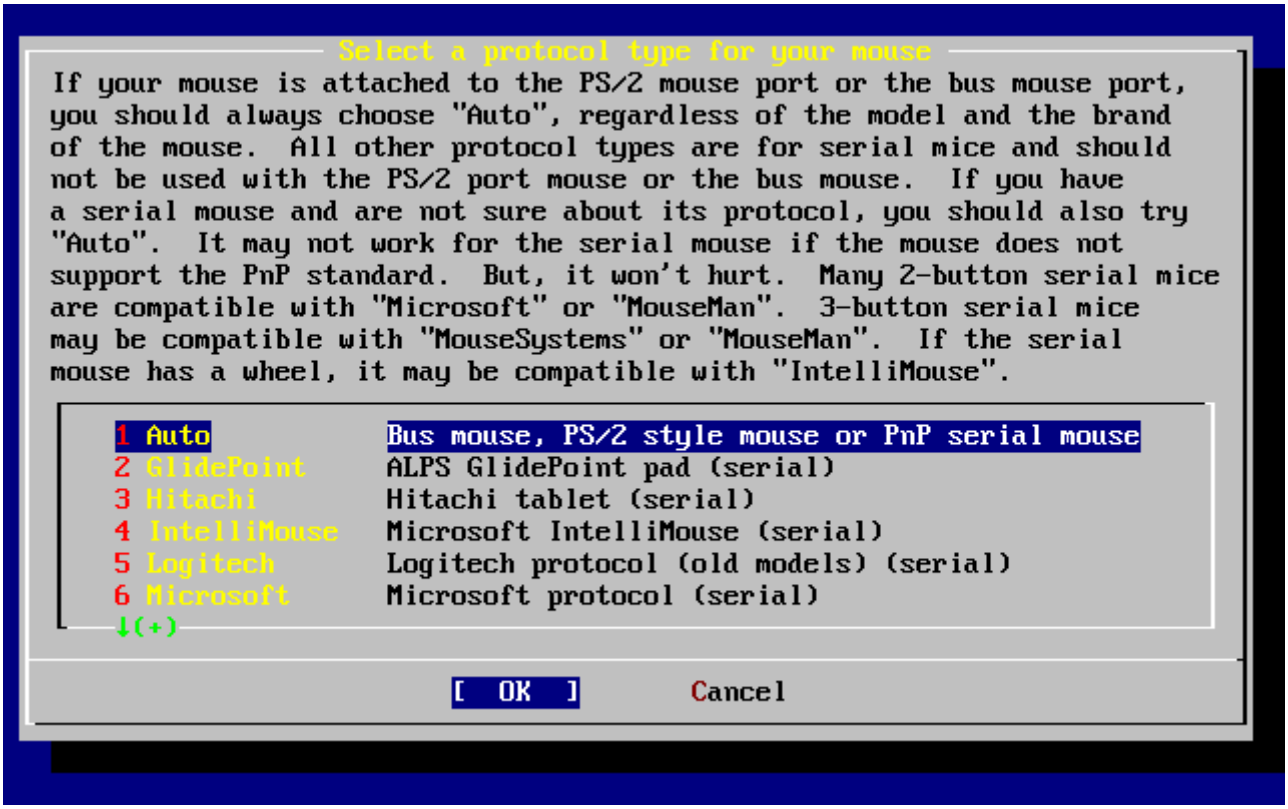
```
                    User Confirmation Requested
          Does this system have a PS/2, serial, or bus mouse?

                        [ Yes ]    No
```

**[ yes ]** гэж PS/2, цуваа эсвэл bus буюу шугаман төрлийн хулганыг сонгох юм уу эсвэл **[ no ]** гэж USB төрлийн хулганыг сонгоод `Enter` товч дээр дараарай.



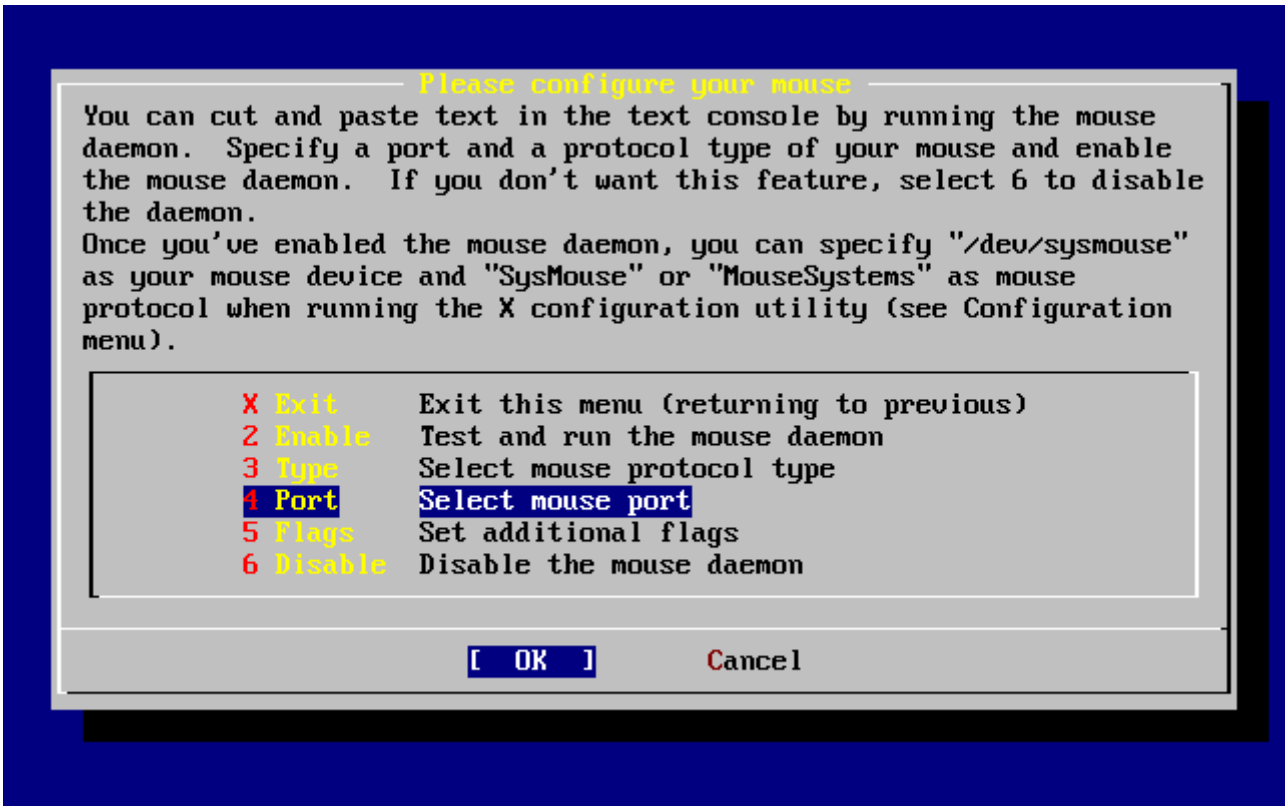

*Зураг 40. Хулганы холбогдох төрлийг сонгох нь*

Сумтай товч ашиглан Type гэж сонгоод `Enter` дарна.



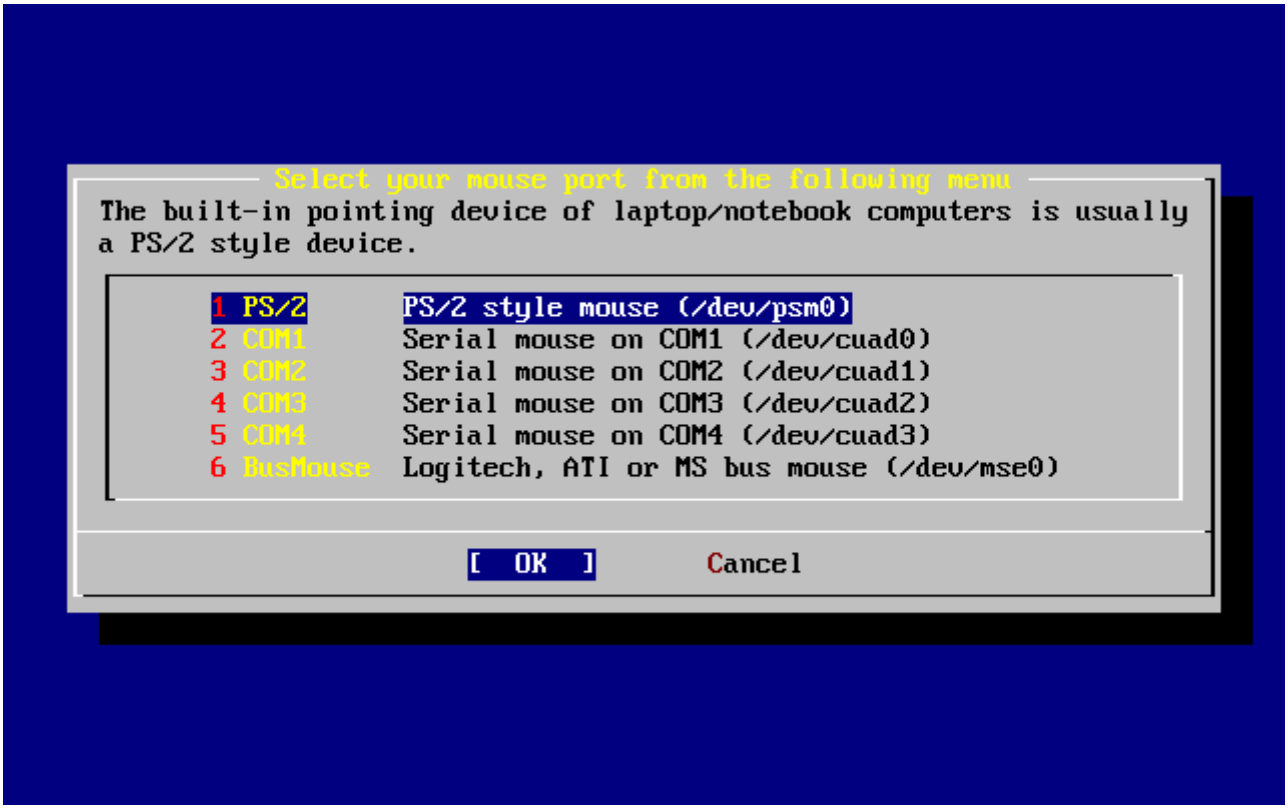

*Зураг 41. Хулганы холбогдох төрлийг тохируулах*

Энэ жишээнд хэрэглэгдэж байгаа хулгана PS/2 төрлийнх бөгөөд анхдагч Auto буюу автомат гэсэн тохируулга таарч байдаг. Энэ холбогдох төрлийг нь өөрчлөнө гэвэл сумтай товч ашиглан сонгоорой. Дараа нь **[ OK ]**-г гэрэлтүүлж сонгогдсон эсэхийг магадлаад `Enter` дээр дарж энэ цэснээс гарах болно.



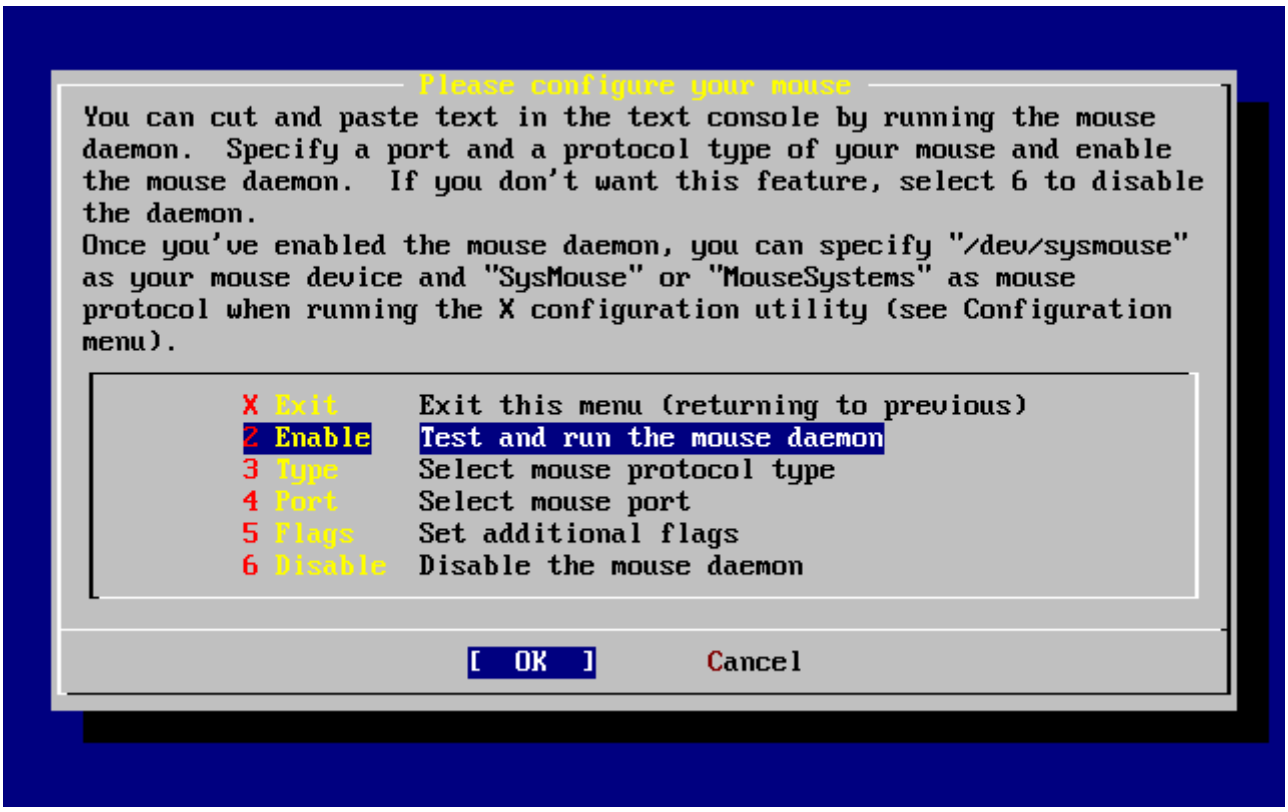

*Зураг 42. Хулганы холбогдох портыг нь сонгох*

Сумтай товч ашиглан Port цэсийг сонгоод `Enter` дээр дарна.



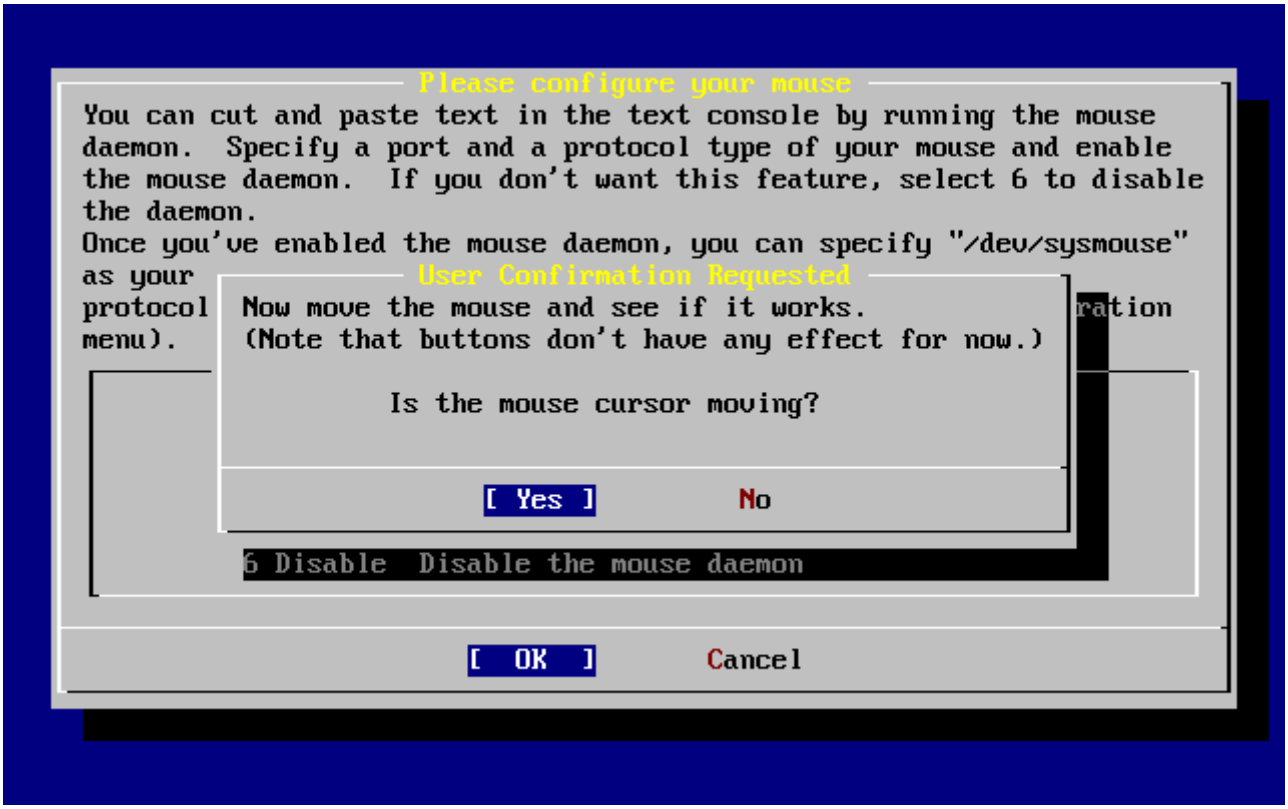

*Зураг 43. Хулганы портыг тохируулах*

Энэ системд PS/2 төрлийн хулгана хэрэглэгдэж байгаа болохоор анхдагч тохируулга болох PS/2 сонгогдсон байдаг. Портыг солихын тулд сумтай товч ашиглаж сонгоод `Enter` дээр дараарай.




*Зураг 44. Хулганы Daemon буюу далд чөтгөрийг идэвхжүүлэх нь*

Эцэст нь сумтай товч ашиглаад Enable буюу идэвхжүүл гэж сонгоод `Enter` товч дээр дарж хулганыг ажиллуулагч далд чөтгөрийг ажиллуулж эхэлдэг.




*Зураг 45. Хулганы далд чөтгөрийг шалгах*

Хулганаа дэлгэцэн дээр хөдөлгөж шалгаж үзэх хэрэгтэй. Хэрэв зүгээр байх юм бол **[ yes ]** гэж сонгоод `Enter` дарна. Хэрэв үгүй бол хулгана зөв тохируулагдаагүй байна гэсэн үг - тийм болохоор **[ no ]** гэж сонгоод өөр тохируулга хийж турших хэрэгтэй.

Exit буюу гарах гэдгийг сумтай товчоор очиж сонгоод `Enter` дээр дарж суулгацын дараах тохируулга уруу буцаж очиж үргэлжлүүлнэ.

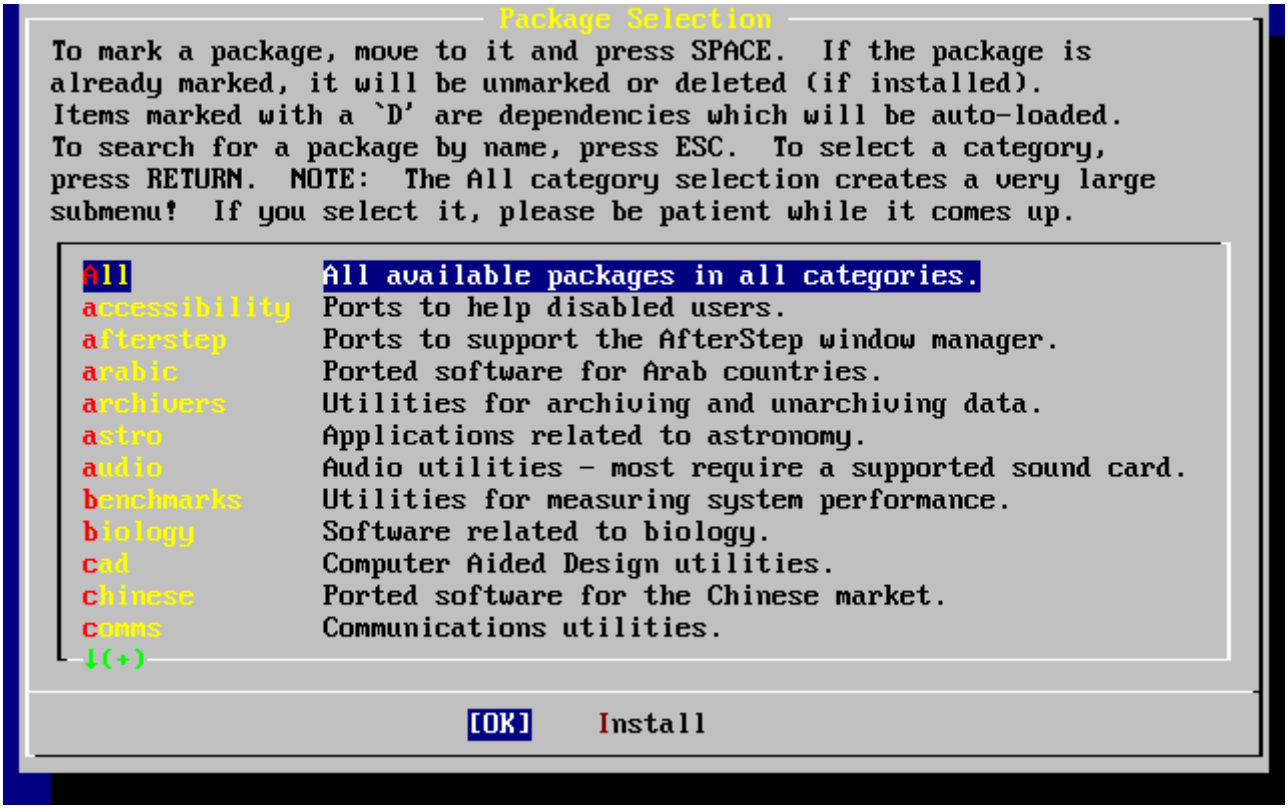
## 2.10.10. Програмын багц суулгах

Багцууд нь урьдчилан хөрвүүлэгдсэн програмууд бөгөөд програм суулгах хамгийн эвтэйхэн арга юм.

Үзүүлэх журмаар нэг багцыг суулгах явцыг энд харуулав. Хэрэв шаардлагатай бол нэмэлт багцууд мөн давхар суугддаг. Суулгасны дараа `sysinstall`-г нэмэлт багц суулгахад хэрэглэж болно.

```
                    User Confirmation Requested
The FreeBSD package collection is a collection of hundreds of
ready-to-run applications, from text editors to games to WEB servers
and more. Would you like to browse the collection now?

                             [ Yes ]   No
```

**[ yes ]** гэж сонгоод `Enter` дарвал багц програмуудыг сонгох цонхонд очих болно:

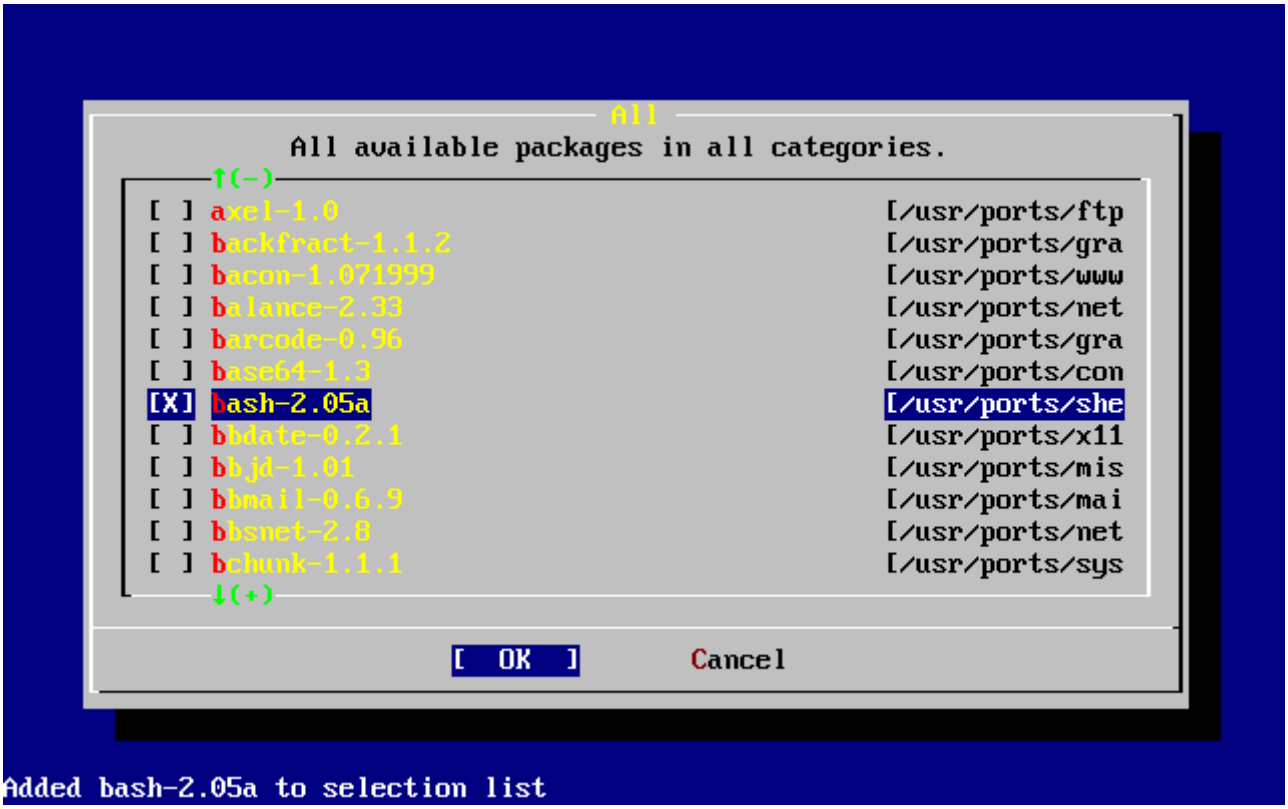



*Зураг 46. Багцын төрлийг сонгох*

Зөвхөн тухайн үед сонгогдсон байгаа суулгацын төрөл дээр байгаа багцууд харуулагдах болно.

Хэрэв All гэж сонговол, байгаа бүх багцыг харж болно. Сумтай товч ашиглан сонгоод `Enter` дээр дараарай.

Байгаа бүх багцууд сонгогдож болохоор танд харуулагдах болно:



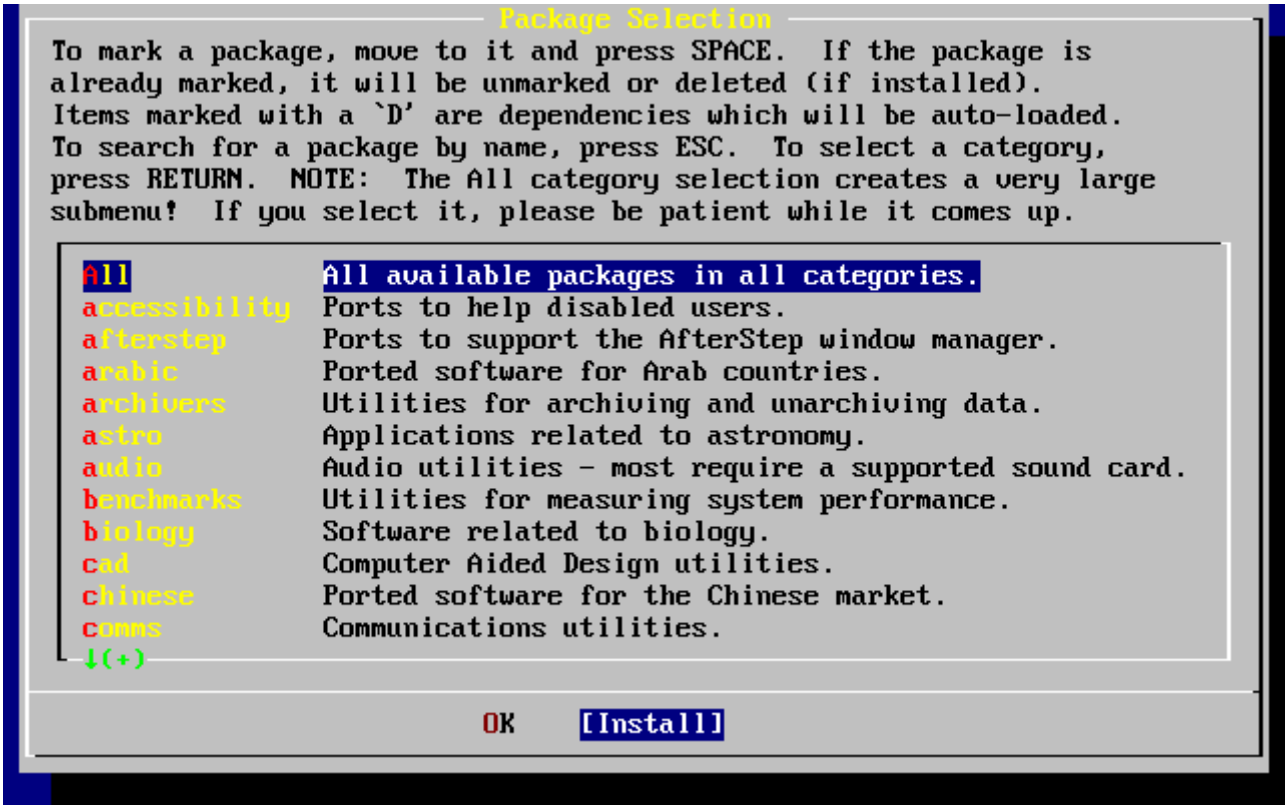

*Зураг 47. Багц сонгох*

Энд харуулснаар bash бүрхүүл сонгогдсон байна. `Space` товч ашиглаж суулгахыг хүссэн бүх багцаа сонгоорой. Сонгох бүрд дэлгэцийн доод хэсэгт багцны товч тайлбар гардаг.

`Tab` товчоор сонгогдсон багц болон **[ OK ]**, ба **[ Cancel ]** сонголтуудын хооронд дамжиж болно.

Суулгахаар хүссэн багцаа сонгож дууссаны дараа `Tab` товч дарж **[ OK ]** гэж сонгоод `Enter` дээр дарж багц сонгох цэс рүү буцаж очдог.

Зүүн ба баруун товчоор бас **[ OK ]** ба **[ Cancel ]**-н хооронд сэлгэж болно. Энэ аргаар та бас **[ OK ]** гэж сонгоод `Enter` дээр дарж багц сонгох хэсэгт буцаж очиж бас болно.




*Зураг 48. Багц суулгах*

`Tab` юм уу эсвэл сумтай товч ашиглан **[ Install ]** гэж сонгоод `Enter` дарах хэрэгтэй. Таны сонгосон багцуудыг танд харуулж суулгахыг хүсэж байгааг тань лавлаж асуух болно:

*Зураг 49. Багц суулгалтын лавлалт*

**[ OK ]** гэж сонгоод `Enter` дарж багц суулгалтыг гүйцэтгэдэг. Суулгах явцын мэдэгдлүүд суулгаж дуустал танд харуулагддаг. Хэрэв ямар нэг алдаа гарвал тэмдэглэж авах хэрэгтэй.

Багц суулгасны дараа сүүлийн тохиргоог хийдэг. Та багц суулгаж дууссан ч гэсэн сүүлийн тохируулгыг хийхийн тулд **[ Install ]** гэдгийг бас дарж болно.

### 2.10.11. Хэрэглэгч/Бүлэг нэмэх

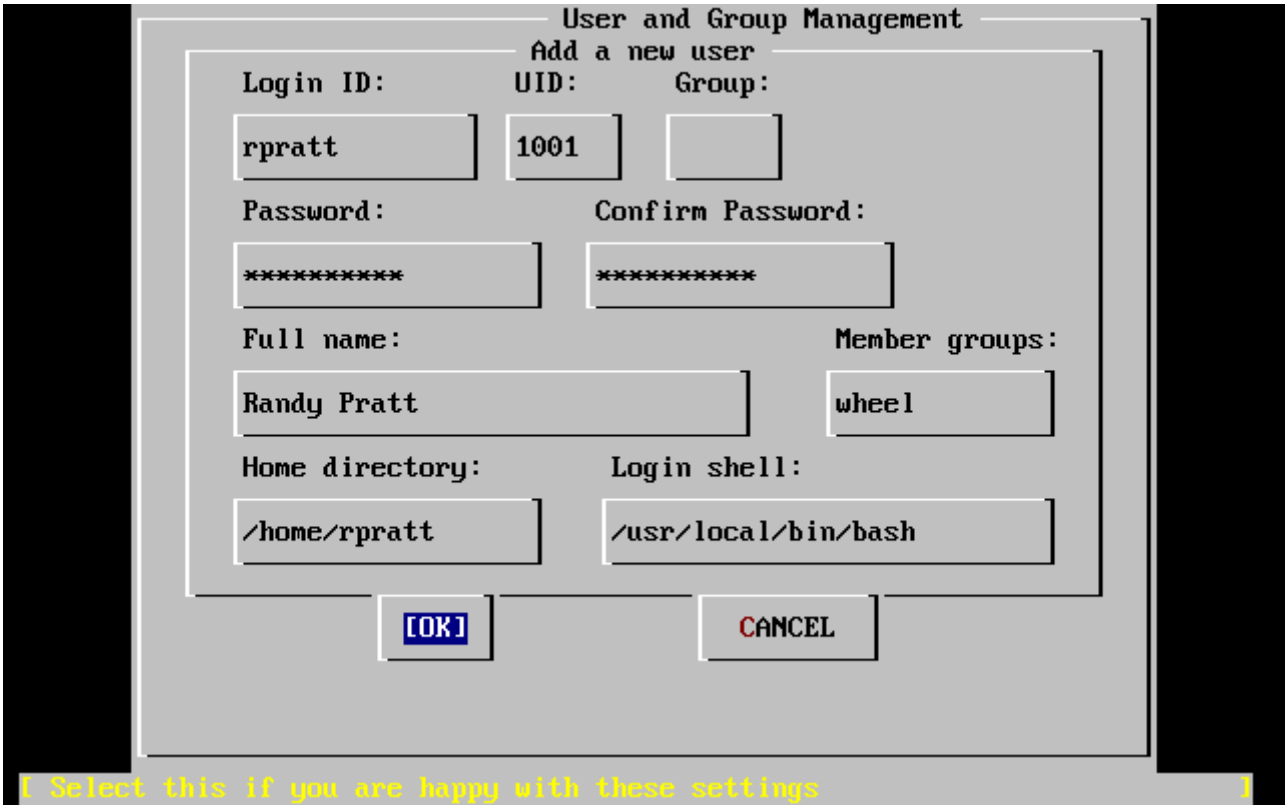
Суулгацын явцад та хамгийн багадаа нэг хэрэглэгчийн эрх нээх хэрэгтэй бөгөөд, энэ эрхээрээ `root` эрх хэрэглэлгүйгээр системдээ нэвтэрч болно. root хуваалт нь голдуу тун бага хэмжээтэй байдаг болохоор `root` эрхээр програм ажиллуулбал түргэн дүүрдэг. Хамгийн хэцүү хүндрэлийг доор мэдүүлж байна:

```
                    User Confirmation Requested
Would you like to add any initial user accounts to the system? Adding
at least one account for yourself at this stage is suggested since
working as the "root" user is dangerous (it is easy to do things which
adversely affect the entire system).

                              [ Yes ]   No
```

**[ yes ]** гэж сонгоод `Enter` дарж шинэ хэрэглэгч нэмэх болно.

*Зураг 50. User буюу хэрэглэгч гэж сонгоно*

Сумтай товч хэрэглэн User гэж сонгоод `Enter` дээр дарна.




*Зураг 51. Хэрэглэгчийн мэдээллийг оруулах*

`Tab` товч хэрэглэн талбаруудад дамжин оруулга хийх бүр доор харуулсан тайлбарууд талбар бүрд харуулагдах болно:

**Login ID**

Хэрэглэгчийн нэвтрэх шинэ нэр(заавал шаардлагатай).

**UID**

Уг хэрэглэгчийн ID дугаар (хоосон орхивол автоматаар сонгогдох болно).

**Group**

Уг хэрэглэгчийн хамрагдах бүлгийн нэр (хоосон орхивол автоматаар сонгогдох болно).

**Password**

Тухайн хэрэглэгчийн нууц үг (энэ талбарт тун анхааралтай оруулах хэрэгтэй!).

**Full name**

Хэрэглэгчийн бүтэн нэр (дэлгэрэнгүй мэдээлэл).

**Member groups**

Энэ хэрэглэгчийн хамрагдаж болох бүлгүүд (өөрөөр хэлбэл хандаж болох бүлгүүд).

**Home directory**

Хэрэглэгчийн эхлэл сан (хоосон орхивол автоматаар сонгогдох болно ).

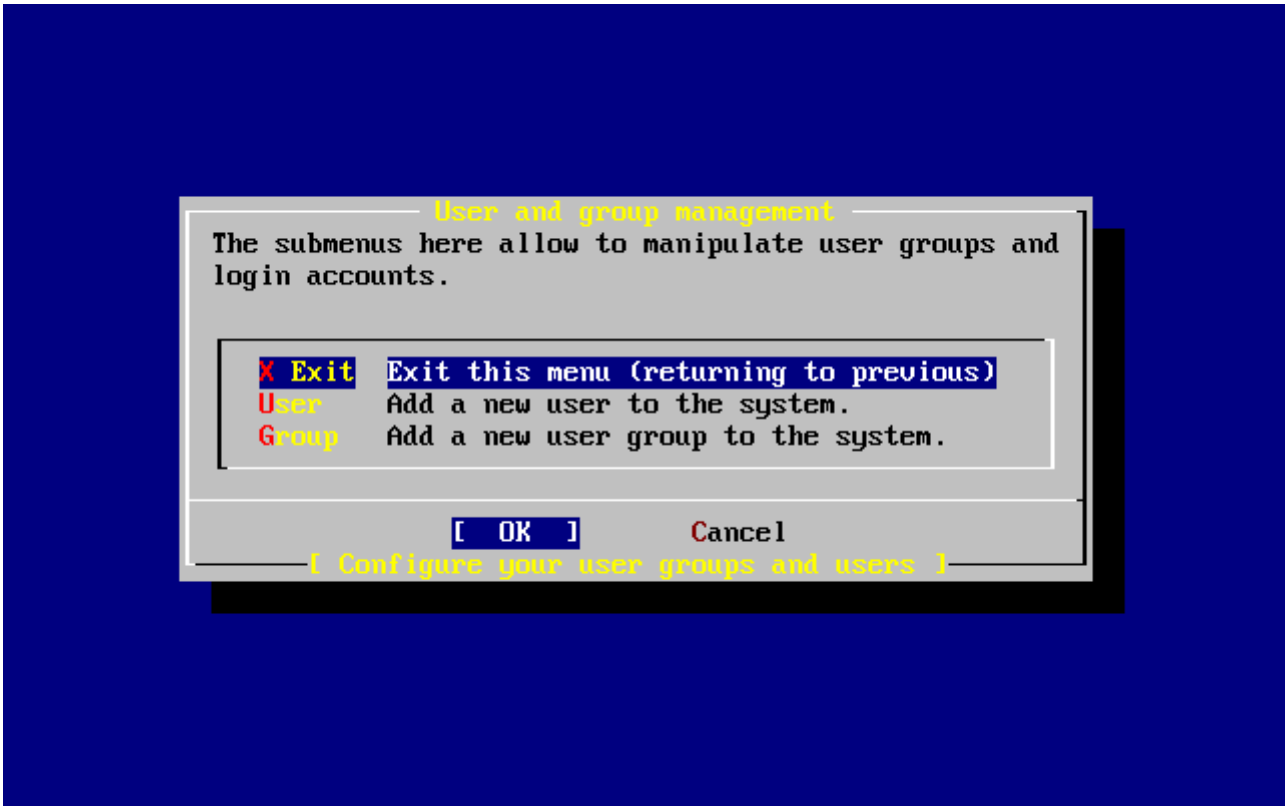
**Login shell**

Хэрэглэгчийн холбогдох бүрхүүл (хоосон орхивол анхдагч бүрхүүл сонгогдох болно. Өөрөөр хэлбэл, /bin/sh).

Жишээн дээр bash-г суулгасан болохоор, /bin/sh гэдгийг /usr/local/bin/bash гэж орлуулж болно. Суугдаагүй бүрхүүл хэрэглэвэл систем уруугаа нэвтэрч чадахгүй болно. BSD ертөнцөд хамгийн өргөн хэрэглэдэг бүрхүүл бол C бүрхүүл бөгөөд /bin/tcsh гэж олддог.

Мөн энэ хэрэглэгч нь `wheel` бүлэг рүү нэмэгдсэн бөгөөд ингэснээр `root` эрхүүдтэй супер хэрэглэгч болох боломжтой болно.

Хэрэв та оруулсан мэдээллээ зөв боллоо гэж бодож байгаа бол **[ OK ]** дээр дарахад хэрэглэгч ба бүлэг нэмэх цонх дахин харуулагддаг:




*Зураг 52. Хэрэглэгч ба бүлэг зохицуулах хэсгээс гарах*

Хэрэгцээтэй бол энэ үед бүлэг нэмж болно. Хэрэв шаардлага гарвал суулгацын дараа `sysinstall` ашиглан нэмж болно.

Хэрэв та хэрэглэгч нэмж дууссан бол сумтай товч ашиглан Exit гэж сонгоод `Enter` дээр дарж суулгацаа үргэлжлүүлээрэй.

## 2.10.12. `root` нууц үг оруулах

```
                      Message
Now you must set the system manager's password.
This is the password you'll use to log in as "root".

                       [ OK ]

              [ Press enter or space ]
```

`Enter` дээр дарж `root` нууц үг оруулна.

Нууц үгийг хоёр удаа зөв оруулах ёстой. Мартахааргүй нууц үг оруулаарай гэж сануулах нь илүүдэх байх. Таны оруулсан нууц үгийг давтаж харуулах юм уу эсвэл одоор дүрсэлж харуулахгүй болохыг сануулж байна.

```
New password:
Retype new password :
```

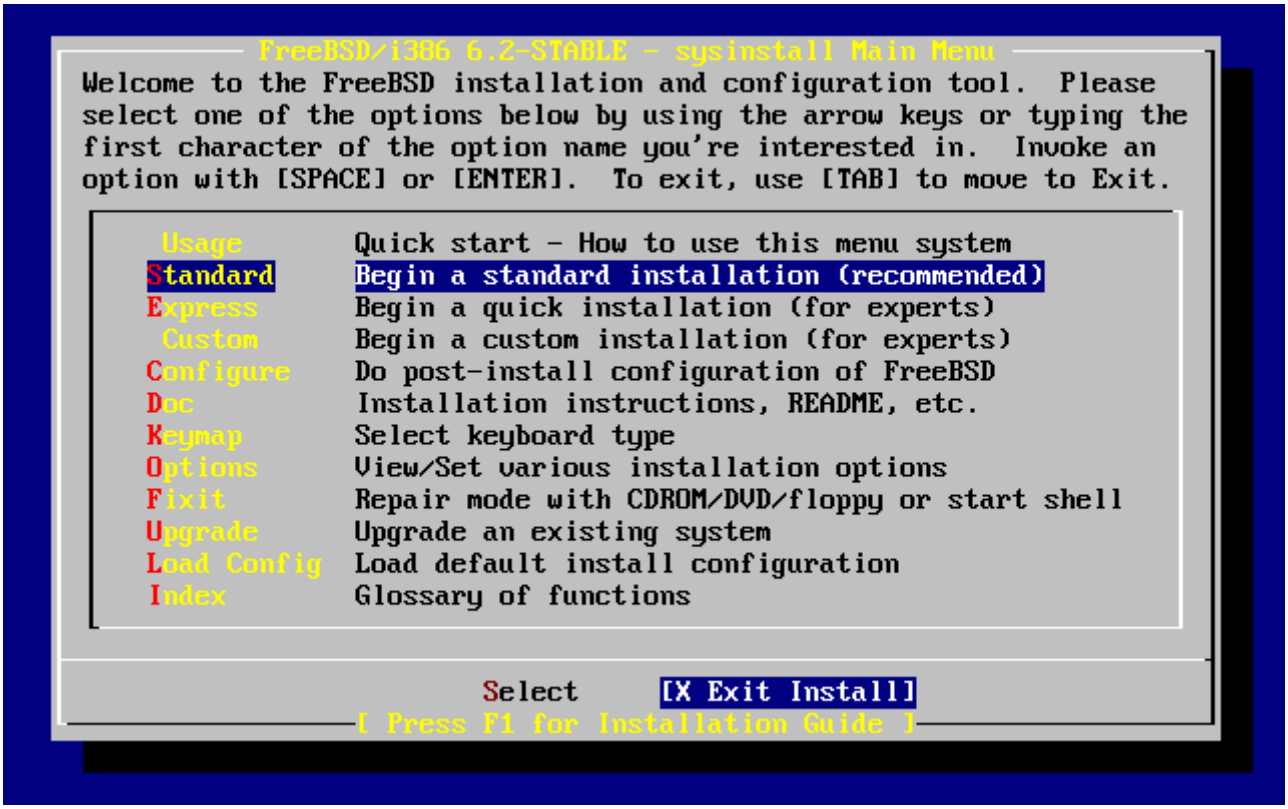
Нууц үгийг амжилттай оруулсны дараа суулгац цаашаагаа үргэлжлэх болно.

### 2.10.13. Суулгацаас гарах

Хэрэв та одоо нэмэлт сүлжээний тохиргоо юм уу эсвэл ямар нэгэн өөр тохиргоо хийхийг хүсвэл, та яг одоо юм уу эсвэл суулгасны дараа `sysinstall` ашиглаж суулгана.

```
                    User Confirmation Requested
 Visit the general configuration menu for a chance to set any last
 options?

                                Yes   [ No ]
```

Сумтай товчоор **[ no ]** гэж сонгоод `Enter` дээр дарж үндсэн суулгацын цэс рүү буцах болно.




*Зураг 53. Суулгацаас гарах*

Сумтай товчоор **[ X Exit Install ]** гэдгийг сонгон `Enter` дээр дарна. Танаас гарах гэж буйг тань лавлаж асуух болно:

```
                    User Confirmation Requested
Are you sure you wish to exit? The system will reboot.

                             [ Yes ]   No
```

**[ yes ]** гэж сонгоно. Хэрэв та CDROM хөтлөгчөөс ачаалж байгаа бол дараах мэдэгдэл дискийг хөтлөгчөөс авахыг танаас хүсэх болно:

```
                    Message
Be sure to remove the media from the drive.

                    [ OK ]
          [ Press enter or space ]
```

Машин дахин ачаалж эхэлтэл CDROM цоожлогдох бөгөөд тэр үед та (хурдан) сугалж авч болно. **[ OK ]** дарж дахин ачаална.

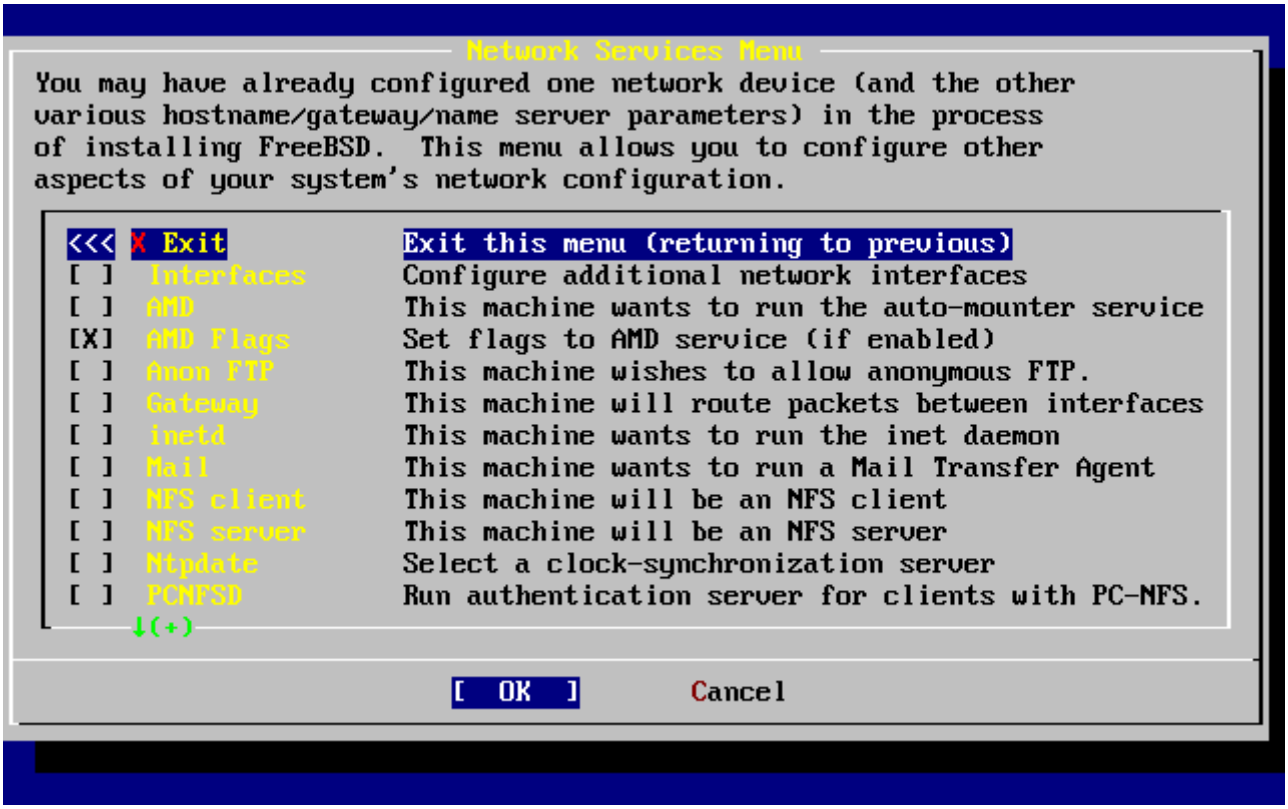
Систем одоо шинээр ачаалагдах бөгөөд ямар нэгэн алдаа гарахыг ажиглах хэрэгтэй. Дэлгэрэнгүйг FreeBSD эхлэн ачаалалт-с үзнэ үү.

## 2.10.14. Сүлжээний үйлчилгээний нэмэлт тохиргоо

Сүлжээний үйлчилгээнүүдийг тохируулах нь, энэ талын мэдлэггүй шинэ хэрэглэгчдэд нэлээн хатуу боорцог мэт санагддаг. Сүлжээний тохиргоог хийж интернэт хэрэглэнэ гэдэг бол FreeBSD-г оруулаад орчин үеийн үйлдлийн системүүдэд маш чухал үйлдэл бөгөөд ингэж сурах явцдаа FreeBSD-н сүлжээтэй холбоотой нэмэлт боломжуудыг ойлгож авдаг. Суулгацын явцад энэ үйлдлийг хийхээсээ өмнө, хэрэглэгч нь хэрэглэх үйлчилгээнийхээ тухай тодорхой ойлголттой байх ёстой.

Сүлжээний үйлчилгээнүүд нь, сүлжээгээр ямар нэгэн оруулга хүлээж авдаг програмуудыг хэлж байгаа юм. Эдгээр програмуудыг " эвдэлж сөнөөдөг " төрлийнх биш байлгах гэж маш их хичээсэн байдаг. Харамсалтай нь програмчид төгс биш байдаг болохоор сүлжээний үйлчилгээнд байгаа ямар нэг цоорхой нүх хорхойг довтлогчид ашиглан муу юманд хэрэглэдэг. Та зөвхөн өөрийнхөө мэддэг болон хэрэгтэй сүлжээний үйлчилгээгээ идэвхжүүлэх нь чухал. Хэрэв танд эргэлзээтэй байгаа бол, уг үйлчилгээг хэрэгцээтэй болох хүртлээ хорьсон байх нь дээр байдаг. Та үүнийг сүүлд sysinstall ашиглаад юм уу эсвэл /etc/rc.conf файлын тусламжтайгаар хэзээд идэвхжүүлж болно.

**Networking** тохируулгыг сонгоход доор дурдсантай төстэй цэсийг үзүүлэх болно:




*Зураг 54. Сүлжээний дээд түвшний тохиргоо*

Эхний тохиргоо Interfaces буюу сүлжээний төхөөрөмжийн талаар бид өмнө нь Сүлжээний тохиргоо хэсэгт үзсэн болохоор энд алгаслаа.

AMD сонголтыг сонгосноор BSD automatic mount буюу автоматаар таньж холбох хэрэгслийг нэмж өгдөг. Энэ нь голдуу NFS холболт (өмнө тайлбарласан буй) ашиглах үед, алсад буй файл системийг таньж холбоход хэрэглэгддэг. Ямар нэгэн онцгой тохируулга энд шаардлагагүй.

Дараагийн тохируулга, AMD Flags сонголт. Хэрэв энэ сонгогдвол жижиг цэстэй цонх үсэрч гарч ирдэг бөгөөд ямар AMD тохиргоо хийхийг зааж өгдөг. Уг цэсэнд анхдагч тохируулгууд нь заагдчихсан байдаг:

```
-a /.amd_mnt -l syslog /host /etc/amd.map /net /etc/amd.map
```

`-a` сонголт нь анхдагч таньж холбох байршлыг зааж өгдөг бөгөөд энэ тохиолдолд /.amd_mnt гэсэн байна. `-l` сонголт анхдагч log буюу бүртгэл файлыг зааж өгдөг боловч хэрэв `syslogd` хэрэглэгдэж байгаа бол бүх бүртгэж бичих үйлдлүүд системийн log daemon буюу далд ажилладаг бүртгэл бичигч чөтгөр уруу илгээгддэг. /host сан нь алсад буй компьютерийн экспортолсон файлуудыг таньж холбох санг илэрхийлж байхад, /net сан нь IP хаягаас экспортлогдсон файлуудыг таньж холбох санг зааж өгч байна. /etc/amd.map файл нь AMD-н экспортын анхдагч тохируулгуудыг заасан байна.

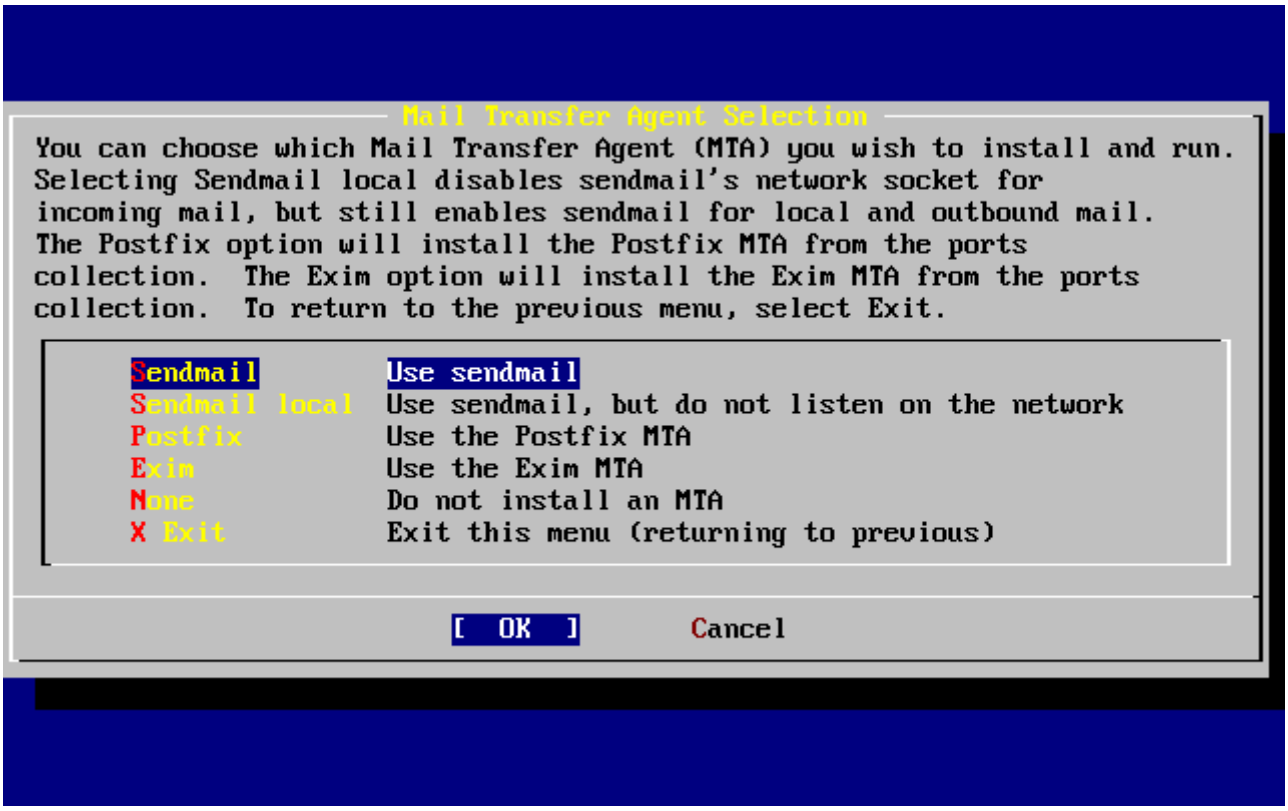
Anon FTP сонголт нь anonymous буюу нэр үл мэдэгдэгч FTP холболтыг зөвшөөрдөг. Үүнийг сонгосноор, уг машиныг нэр нь үл мэдэгдэгч холболт хүлээж авдаг FTP сервер болгож байна. Энэ сонголтыг дагаад үүсэх нууцлал болон аюулгүй байдлыг бодох хэрэгтэй. Өөр цэс үүсэж аюулгүй байдалтай холбоотой нөхцлүүд болон тохируулгын талаар сануулга өгдөг.

Gateway тохируулга нь өмнө тайлбарласны дагуу машиныг хэрхэн сүлжээний хаалга

болгож тохируулах тохиргоонуудыг санал болгодог. Энэ хэсэгт та өмнө нь суулгацын үед санамсаргүй хүсэлгүйгээр тохируулснаа Gateway сонголт ашиглаж буцааж болно.

Inetd сонголтоор өмнө тайлбарласан inetd(8) далд чөтгөрийг тэр чигээр нь хорьж болно.

Mail сонголтоор системийн андагч MTA буюу Mail Transfer Agent буюу захиа дамжуулагч агентийг тохируулдаг. Энэ сонголтыг сонгох үед дараах цэс харуулагддаг:




*Зураг 55. Анхдагч MTA-г сонгох*

Та энэ хэсэгт, суулгах анхдагч MTA-г сонгох гэж байна. MTA бол системийн хэрэглэгчид уруу юм уу эсвэл интернэтэд байгаа хэрэглэгчид уруу захиа илгээгч энгийн сервер билээ.

Sendmail-г сонгосноор, та FreeBSD-н анхдагч бөгөөд алдаршиж тархсан sendmail серверийг суулгах болно. Sendmail local сонголт нь sendmail-г анхдагч MTA болгож суулгах боловч интернэтээс ирж байгаа захианууды хүлээж авахгүйгээр тохируулдаг. Бусад сонголтууд болох Postfix болон Exim нар нь Sendmail-тэй ижил үүрэг гүйцэтгэдэг. Тэд хоёулаа захиа илгээдэг бөгөөд зарим хэрэглэгчид sendmail-н оронд MTA болгож хэрэглэж болдог.

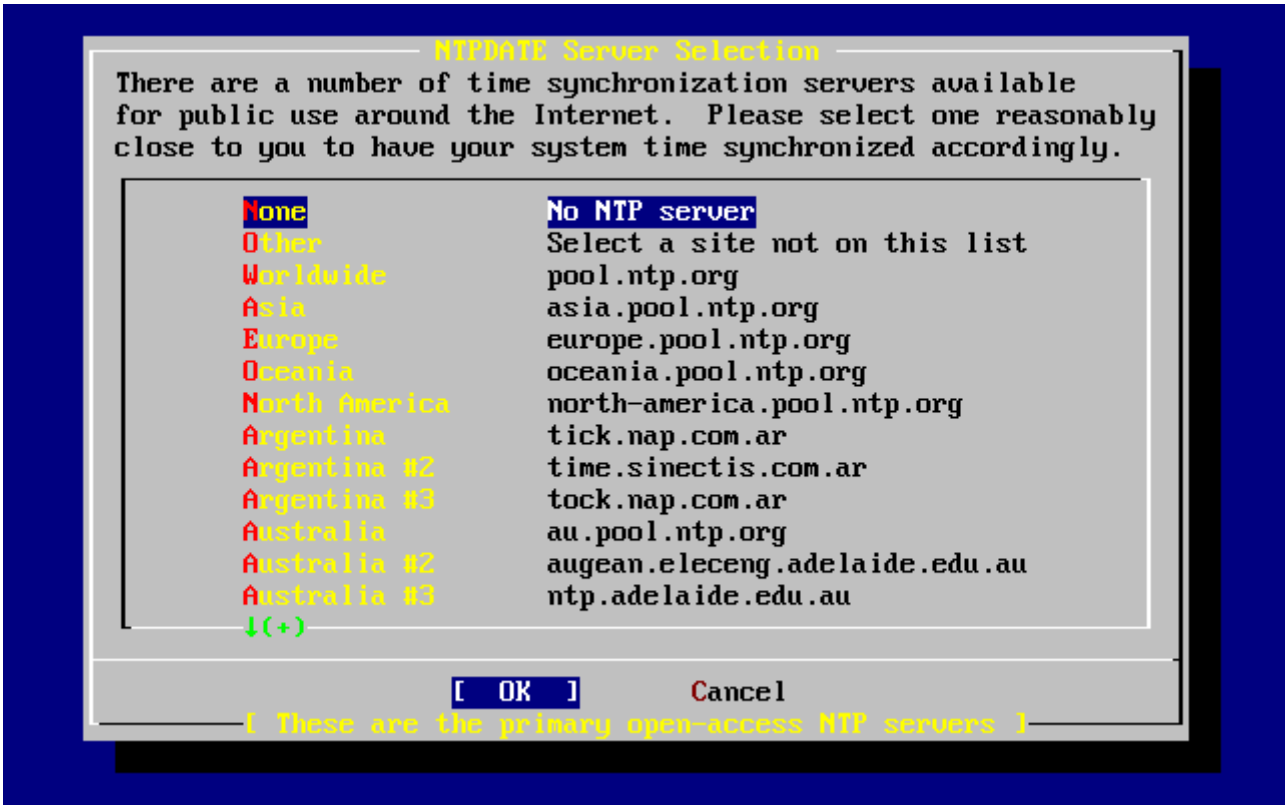
MTA-г сонгосны дараа юм уу эсвэл MTA-г сонголгүйгээр цаашлах үед, NFS client тохируулгыг хийх сүлжээний тохируулгын цэс гарч ирдэг.

NFS client сонголт нь системийг NFS сүлжээгээр сервертэй холбоход хэрэглэгддэг. NFS сервер нь NFS протокол ашиглан, файл системийг сүлжээгээр бусад машинд хандах боломж өгдөг. Хэрэв энэ нь дангаараа ажилладаг машин бол, энэ сонголтыг сонгохгүйгээр үлдээж болно. Систем танаас нэмэлт тохируулга хийхийг шаардаж магадгүй; сервер хэрэглэгч хоёрын тохиргооны тухай дэлгэрэнгүй мэдээллийг Сүлжээний Файлын Систем (NFS) хэсгээс харна уу.

Саяны сонголтын доор нь NFS server-н тохируулга байгаа бөгөөд системийг NFS сервер

болгоход хэрэглэгдэнэ. Энэ хэсэгт RPC буюу алсад буй компьютерт програм ажиллуулагч үйлчилгээг эхлүүлэх тохиргоо хийдэг. RPC нь компьютер болон програм хоёрын хоорондох холболтыг зохицуулахад хэрэглэгддэг.

Түүний дараа Ntpdate гэсэн сонголт байгаа ба энэ нь цаг тохируулах үүрэгтэй. Хэрэв үүнийг сонговол, дараах цэс танд харуулагдана:

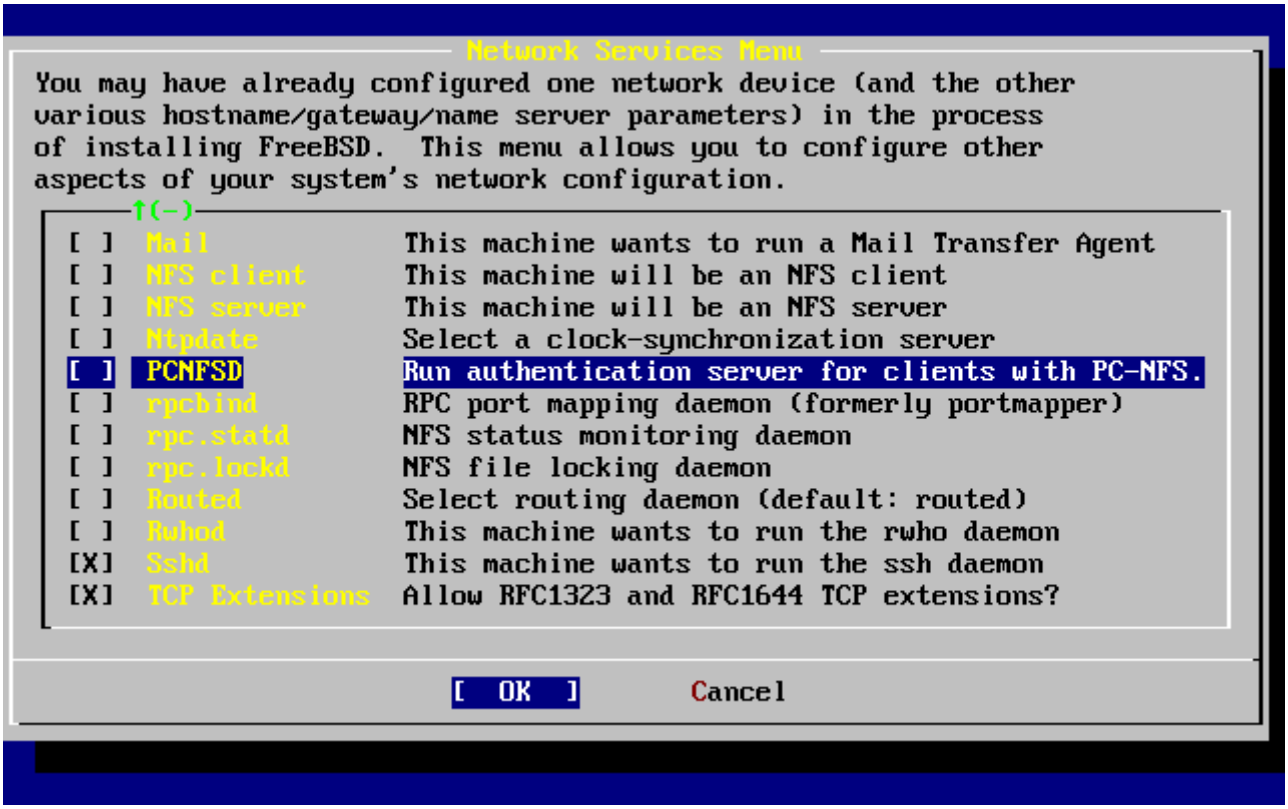



*Зураг 56. Ntpdate тохиргоо*

Энэ хэсэгт, өөртөө хамгийн ойрхон буй серверийг сонгох хэрэгтэй. Ойрхон сервер сонгосноор, таны цагийн тохиргоог хол байгаа сервер уруу мэдээлэл авах гэж цаг алдалгүйгээр илүү нарийвчлалтай хийх боломж ихсэнэ.

Дараагийн сонголт бол PCNFSD юм. Энэ сонголт net/pcnfsd багцыг Портын цуглуулгаас суулгадаг. Энэ нь Microsoft-н MS-DOS® зэрэг таньж бүртгэн нэвтрэх үйлчилгээ хийж чаддаггүй үйлдлийн системүүдэд NFS таньж нэвтрүүлэх үйлчилгээг санал болгодог тун хэрэгтэй систем юм.

Та одоо жаахан доош нь гулгуулж дараагийн сонголтуудыг харах хэрэгтэй:




*Зураг 57. Доод түвшний сүлжээний тохируулга*

rpcbind(8), rpc.statd(8), болон rpc.lockd(8) хэрэгслүүд бүгдээрээ Remote Procedure Calls (RPC) буюу алс компьютерт процедур ажиллуулагчид хэрэглэгдэнэ. `rpcbind` хэрэгсэл NFS сервер болон хэрэглэгчийн хоорондох холболтыг зохицуулдаг бөгөөд NFS серверийг зөв ажиллаж байх нөхцөл шаарддаг. rpc.statd далд чөтгөр нь өөр компьютер дээр буй rpc.statd далд чөтгөртэй харилцаж төлөв байдлыг хянаж байдаг. Мэдээлэгдэх ёстой төлөв байдлууд нь /var/db/statd.status файлд хадгалагаж байгаа. Дараагийн сонголт нь rpc.lockd бөгөөд энэ нь сонгогдвол файлуудыг түгжих үйлдлийг хангаж өгдөг. Энэ голдуу rpc.statd-тэй хэрэглэгддэг бөгөөд ямар компьютер ямар файлыг хорихыг хүсэж байна, хэр давтамжтай хүсэж байгаа зэргийн төлвийг хянаж байдаг. Сүүлийн хоёр сонголт нь дибаг хийхэд маш гайхамшигтай байдаг боловч эдгээр нь NFS сервер болон хэрэглэгч зөв ажиллахад шаардлагагүй.

Цааш нь харвал Routed сонголт буй. Энэ нь дамжуулга хийгч далд чөтгөр юм. routed(8) хэрэгсэл нь сүлжээний дамжуулах хүснэгтийг удирдаж multicast дамжуулагчдыг олж хүсэлтийн дагуу сүлжээн дэх физик холболттой машинд өөрийнхөө дамжуулах хүснэгтийг нийлүүлж байдаг. Энэ хэрэгсэл нь голдуу дотоод сүлжээндээ хаалганы үүрэг гүйцэтгэж байгаа машинд хэрэглэгддэг. Үүнийг сонговол цэс гарч ирэн уг хэрэгслийн байрлах анхдагч байршлыг асуудаг. Уг цэсэн дээр анхдагч байршил нь аль хэдийнээ сонгогдсон байдаг болохоор шууд `Enter` дарж болно. Танд дахин өөр цэс харуулагдах бөгөөд, энэ үед танаас routed далд чөтгөрийг ажиллууллах нэмэлт сонголт оруулахыг асуудаг. Анхдагч сонголт нь `-q` байдаг бөгөөд үргэлж энэ сонголттойгоо харуулагдах болно.

Дараагийн тохиргооны сонголт бол Rwhod бөгөөд хэрэв энэ сонгогдвол, системийн эхлэн ачаалах үед rwhod(8) далд чөтгөрийг ажиллуулдаг. `rwhod` хэрэгсэл нь сүлжээгээр байнга системийн мэдээнүүдийг цацах, эсвэл "consumer" горимдоо байгаа бол уг мэдээнүүдийг цуглуулж байдаг. Дэлгэрэнгүй мэдээллийг ruptime(1) юм уу rwho(1)-н гарын авлагаас харж болно.

Жагсаалтын сүүлийн өмнөх сонголт бол sshd(8)далд чөтгөр юм. Энэ нь OpenSSH-д зориулсан, нууцлалын хувиргууртай бүрхүүлийн сервер бөгөөд энгийн хэрэглээнд байдаг telnet болон FTP серверүүдийн оронд хэрэглэхийг санал болгодог. sshd сервер нь нэг компьютераас нөгөө руу нууцлалын хувиргалттай холболт тогтоож байдаг.

Эцэст нь TCP Extensions сонголт байгаа. Энэ хэрэв сонгогдвол, RFC 1323 болон RFC 1644-д заасан байдаг TCP өргөтгөлийг идэвхжүүлдэг. Ихэнх компьютер дээр энэ нь сүлжээний хурдыг ихэсгэдэг ч гэсэн зарим холболтыг унагаж болзошгүй. Энэ нь зөвхөн дангаараа ажилладаг серверт санал болгогдож байдаг.

Ингээд та сүлжээнийхээ үйлчилгээнүүдийг тохируулж дууслаа. Та хамгийн дээд талын X Exit уруу гүйлгэж дараагийн тохиргоо уруу орон үргэлжлүүлж болох бөгөөд эсвэл X Exit-ийг хоёр удаа сонгон дараа нь **[ X Exit Install ]**-г сонгон sysinstall-с гарч болно.

## 2.10.15. FreeBSD эхлэн ачаалалт

### 2.10.15.1. FreeBSD/i386 машин дээрх FreeBSD-н эхлэн ачаалалт

Хэрэв бүх зүйл амжилттай болсон бол дэлгэцэн дээр тань мэдэгдлүүд урсаж өнгөрөөд нэвтрэх хэсэгт тулж ирэх болно. Урсаж өнгөрсөн мэдэгдлүүдийг эргэж харахын тулд `Scroll-Lock` товчийг дараад `PgUp` болон `PgDn` товчоор дээш доош нь гулгуулдаг. `Scroll-Lock` товчийг дахин дарснаар нэвтрэх хэсэгт буцаж ирдэг.

Бүх мэдэгдэл харуулагдахгүй байх нөхцөл үүсвэл (түр хадгалагчийн хэмжээнээс болж) сүүлд нь тушаал өгдөг горим дээр `dmesg` тушаалаар харж болдог.

Суулгацын үед нэмж оруулсан хэрэглэгчийн нэр болон нууц үгээр нэвтэрч орох хэрэгтэй (энэ тохиолдолд `rpratt` гэж буй). Шаардлага гараагүй бол `root` эрхээр орохоос сэргийлэх хэрэгтэй.

Ердийн эхлэн ачаалах үеийн мэдэгдэл (хувилбарын тухай хэсгийг оруулалгүйгээр):

```
Copyright (c) 1992-2002 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.

Timecounter "i8254"  frequency 1193182 Hz
CPU: AMD-K6(tm) 3D processor (300.68-MHz 586-class CPU)
  Origin = "AuthenticAMD"  Id = 0x580  Stepping = 0
  Features=0x8001bf<FPU,VME,DE,PSE,TSC,MSR,MCE,CX8,MMX>
  AMD Features=0x80000800<SYSCALL,3DNow!>
real memory  = 268435456 (262144K bytes)
config> di sn0
config> di lnc0
config> di le0
config> di ie0
config> di fe0
config> di cs0
config> di bt0
config> di aic0
config> di aha0
```

```
config> di adv0
config> q
avail memory = 256311296 (250304K bytes)
Preloaded elf kernel "kernel" at 0xc0491000.
Preloaded userconfig_script "/boot/kernel.conf" at 0xc049109c.
md0: Malloc disk
Using $PIR table, 4 entries at 0xc00fde60
npx0: <math processor> on motherboard
npx0: INT 16 interface
pcib0: <Host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcib1: <VIA 82C598MVP (Apollo MVP3) PCI-PCI (AGP) bridge> at device 1.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <Matrox MGA G200 AGP graphics accelerator> at 0.0 irq 11
isab0: <VIA 82C586 PCI-ISA bridge> at device 7.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <VIA 82C586 ATA33 controller> port 0xe000-0xe00f at device 7.1 on pci0
ata0: at 0x1f0 irq 14 on atapci0
ata1: at 0x170 irq 15 on atapci0
uhci0: <VIA 83C572 USB controller> port 0xe400-0xe41f irq 10 at device 7.2 on pci0
usb0: <VIA 83C572 USB controller> on uhci0
usb0: USB revision 1.0
uhub0: VIA UHCI root hub, class 9/0, rev 1.00/1.00, addr 1
uhub0: 2 ports with 2 removable, self powered
chip1: <VIA 82C586B ACPI interface> at device 7.3 on pci0
ed0: <NE2000 PCI Ethernet (RealTek 8029)> port 0xe800-0xe81f irq 9 at
device 10.0 on pci0
ed0: address 52:54:05:de:73:1b, type NE2000 (16 bit)
isa0: too many dependant configs (8)
isa0: unexpected small tag 14
fdc0: <NEC 72065B or clone> at port 0x3f0-0x3f5,0x3f7 irq 6 drq 2 on isa0
fdc0: FIFO enabled, 8 bytes threshold
fd0: <1440-KB 3.5" drive> on fdc0 drive 0
atkbdc0: <keyboard controller (i8042)> at port 0x60-0x64 on isa0
atkbd0: <AT Keyboard> flags 0x1 irq 1 on atkbdc0
kbd0 at atkbd0
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: model Generic PS/2 mouse, device ID 0
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
sc0: <System console> at flags 0x1 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
sio0 at port 0x3f8-0x3ff irq 4 flags 0x10 on isa0
sio0: type 16550A
sio1 at port 0x2f8-0x2ff irq 3 on isa0
sio1: type 16550A
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
ppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/15 bytes threshold
ppbus0: IEEE1284 device found /NIBBLE
Probing for PnP devices on ppbus0:
plip0: <PLIP network interface> on ppbus0
```

```
lpt0: <Printer> on ppbus0
lpt0: Interrupt-driven port
ppi0: <Parallel I/O> on ppbus0
ad0: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata0-master using UDMA33
ad2: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata1-master using UDMA33
acd0: CDROM <DELTA OTC-H101/ST3 F/W by OIPD> at ata0-slave using PIO4
Mounting root from ufs:/dev/ad0s1a
swapon: adding /dev/ad0s1b as swap device
Automatic boot in progress...
/dev/ad0s1a: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1a: clean, 48752 free (552 frags, 6025 blocks, 0.9% fragmentation)
/dev/ad0s1f: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1f: clean, 128997 free (21 frags, 16122 blocks, 0.0% fragmentation)
/dev/ad0s1g: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1g: clean, 3036299 free (43175 frags, 374073 blocks, 1.3% fragmentation)
/dev/ad0s1e: filesystem CLEAN; SKIPPING CHECKS
/dev/ad0s1e: clean, 128193 free (17 frags, 16022 blocks, 0.0% fragmentation)
Doing initial network setup: hostname.
ed0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
        inet6 fe80::5054::5ff::fede:731b%ed0 prefixlen 64 tentative scopeid 0x1
        ether 52:54:05:de:73:1b
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x8
        inet6 ::1 prefixlen 128
        inet 127.0.0.1 netmask 0xff000000
Additional routing options: IP gateway=YES TCP keepalive=YES
routing daemons:.
additional daemons: syslogd.
Doing additional network setup:.
Starting final network daemons: creating ssh RSA host key
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
cd:76:89:16:69:0e:d0:6e:f8:66:d0:07:26:3c:7e:2d root@k6-2.example.com
 creating ssh DSA host key
Generating public/private dsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
f9:a1:a9:47:c4:ad:f9:8d:52:b8:b8:ff:8c:ad:2d:e6 root@k6-2.example.com.
setting ELF ldconfig path: /usr/lib /usr/lib/compat /usr/X11R6/lib
/usr/local/lib
a.out ldconfig path: /usr/lib/aout /usr/lib/compat/aout /usr/X11R6/lib/aout
starting standard daemons: inetd cron sshd usbd sendmail.
Initial rc.i386 initialization:.
rc.i386 configuring syscons: blank_time screensaver moused.
Additional ABI support: linux.
Local package initialization:.
Additional TCP options:.
```

```
FreeBSD/i386 (k6-2.example.com) (ttyv0)

login: rpratt
Password:
```

RSA болон DSA түлхүүрүүдийг үүсгэх явц нь удаан машин дээр хэсэг хугацаа авч магадгүй. Энэ явц нь зөвхөн суулгацын дараах анхны эхлэн ачаалах үед л хийгддэг. Дараагийн ачаалах үед хийгддэггүй болохоор хурдан байх болно.

Хэрэв X сервер тохируулагдаад анхдагч график орчин сонгогдсон байгаа бол, `startx` гэсэн тушаалаар тэдгээрийг ажиллуулж болно.

### 2.10.16. FreeBSD-г унтраах

Үйлдлийн системийг зөв унтраах нь тун чухал. Шууд хүч хэрэглэн унтрааж болохгүй. Эхлээд `su` гэсэн тушаал оруулаад `root` нууц үгээ бичиж өгөн супер хэрэглэгч болох хэрэгтэй. Энэ нь зөвхөн тухайн хэрэглэгч `wheel` бүлэгт хамаарагдсан үед биелнэ. Эсвэл `root` хэрэглэгч болж нэвтрэх хэрэгтэй. Ингээд `shutdown -h now` тушаалыг оруулна.

```
The operating system has halted.
Please press any key to reboot.
```

Унтраах тушаал өгсний дараа "Please press any key to reboot буюу дурын товч дарж шинээр ачаална уу" гэсэн мэдэгдэл гарсан үед унтраах нь зөв юм. Хэрэв шууд унтраалгүйгээр ямар нэгэн товч дарвал систем шинээр ачаалагдах болно.

Та мөн `Ctrl` + `Alt` + `Del` товчлуурын хослол хэрэглэн шинээр ачаалж болох боловч, ердийн үед энэ нь тийм ч их хэрэглэгддэг арга биш.

## 2.11. Хүндрэл тулгарвал

Дараах хэсэгт хүмүүсийн мэдээлсний дагуу тэдэнд тулгарч байсан энгийн хүндрэлийн тухай авч хэлэлцэх болно. Мөн энэ хэсэгт MS-DOS® эсвэл Windows® үйлдлийн системийг FreeBSD-тэй хамт хэрхэн нэг компьютерт суулгаж ачаалах тухай хүмүүсийн асуусан асуулт болон хариултыг агуулсан буй.

### 2.11.1. Хэрэв буруу тийшээ эргээд эхэлбэл юу хийх хэрэгтэй вэ

Компьютерийн төрлөөс хамааран бүх юмыг 100% цэвэр гүйцэтгэж болохгүй ч гэсэн хэрэв ямар нэгэн зүйл буруу болбол, та хэд хэдэн арга хэмжээг авч чадна.

Таны FreeBSD-ийн хувилбарт зориулсан Тоног төхөөрөмжийн тэмдэглэл баримтаас таны төхөөрөмжийг дэмжсэн эсэхийг нь шалгах хэрэгтэй.

Хэрэв таны төхөөрөмж дэмжигдсэн мөртлөө хүндрэл гарвал, та өөрсчөн тохируулсан цөм бүтээх шаардлагатай болно. Энэ нь GENERIC цөмд байхгүй төхөөрөмжүүдийн дэмжлэгийг нэмэх боломжийг танд олгох болно. Ачаалагч дискэн дээр буй цөм нь таны

төхөөрөмжүүдийг IRQ-үүд, IO хаяг, мөн DMA суваг зэрэг тохиргоог үйлдвэрээс зааж өгсөн анхныхаа утгатайгаа байгаа гэж авч үздэг. Хэрэв таны төхөөрөмжийн тохиргоо өөрчлөгдсөн бол тэдгээр зүйлсийг FreeBSD хаанаас олж болохыг хэлэхийн тулд та цөмийн тохиргоогоо засварлаж дахин бүтээх хэрэгтэй болно.

Мөн залгаж туршигдаагүй төхөөрөмжүүд сүүлд залгагдаад туршигдах үед алдаа үүсгэх тохиолдол гардаг. Ийм үед хүндрэлтэй байгаа төхөөрөмжийн драйвер буюу таниулагч файлуудыг нь хорих хэрэгтэй.

Ихэнх суулгацын хүндрэлүүд нь төхөөрөмжийн үйлдвэрээс гаргасан сүүлийн үеийн таниулагч файлуудыг нь татаж авч суулгаснаар илаарших нөхцөл бүрддэг бөгөөд ялангуяа motherboard буюу эх хавтан дээр иймэрхүү үзэгдэл их гардаг. Эх хавтангийн үйлдвэрээс гаргасан програмыг ер нь BIOS гэж нэрлэдэг ба бараг бүх үйлдвэрүүд өөрсдийн гэсэн вэб хуудастай бөгөөд уг програмыг хэрхэн сүүлийн үеийн хэлбэрт шинэчлэх тухай мэдээлэл болон файлууд агуулж байдаг.

Ихэнх үйлдвэрлэгчид нь, ямар нэгэн ноцтой хүндрэл гарахаас зайлсхийж хүндтэй нөхцөл үүсээгүй бол BIOS-г шинэчилж хэрэггүй гэж зөвлөдөг. Шинэчлэх явц амжилттүй болох *нөхцөл нь* BIOS цахилгаан схемд ноцтой эвдрэл гаргаж болзошгүйг анхаарна уу.

### 2.11.2. MS-DOS® болон Windows®-н файл системийг хэрэглэх нь

FreeBSD нь одоогийн байдлаар Double Space™ програмаар шахсан файл системийг дэмждэггүй. Тийм болохоор, хэрэв ийм төрлийн файл системд хандахаар бол эхлээд уг файл системийн шахалтыг задлаад, дараа нь файл уруу хандах хэрэгтэй. Задлах энэ үйлдэл нь Compression Agent програмыг Start> Programs > доторх System Tools цэснээс гүйцэтгэгдэж болно.

FreeBSD нь MS-DOS® файлын системүүдийг (заримдаа FAT файлын системүүд гэгддэг) дэмжиж чаддаг. mount_msdosfs(8) тушаал нь тийм файлын системүүдийг байгаа сангийн шатлалд системийн агуулгад хандах боломжтой болгон холбож өгдөг. mount_msdosfs(8)-ийг ихэвчлэн шууд ажиллуулдаггүй; харин /etc/fstab файл дахь мөрийн тусламжтай системээр дуудагдах юм уу эсвэл mount(8) хэрэгслийг тохирох нэмэлт өгөгдлүүдтэй нь дууддаг.

/etc/fstab файлд байж болох мөр ийм байна:

```
/dev/ad0sN  /dos  msdosfs rw  0 0
```

/dos сан энэ тохиолдолд өмнө нь үүссэн байх ёстой. /etc/fstab-ийн хэлбэршилтийн талаар илүү дэлгэрэнгүйг fstab(5)-с үзнэ үү.

MS-DOS® файлын системд зориулсан mount(8) дуудлага иймэрхүү байна:

```
# mount -t msdosfs /dev/ad0s1 /mnt
```

Энэ жишээн дээр, MS-DOS® файл систем нь эхний дискний эхний хуваалт дээр байрласныг харуулж байна. Мэдээж таны систем дээр янз бүр байх бөгөөд `dmesg` болон `mount` тушаалуудын гаралтын мэдээллийг харах хэрэгтэй. Эдгээр тушаалууд нь дискний хуваалтын тухай хангалттай мэдээллийг өгч чаддаг.

FreeBSD нь дискний зүсмэлүүдийг (MS-DOS® зүсмэлүүд) бусад үйлдлийн системүүдээс өөрөөр дугаарлаж болно. Ялангуяа өргөтгөсөн MS-DOS® файлын системүүдэд үндсэн MS-DOS® хуваалтуудынхаас дээш дугааруудыг ихэвчлэн өгдөг. fdisk(8) хэрэгсэл нь аль зүсмэлүүд FreeBSD-д хамаарч байгаа болон аль нь бусад үйлдлийн системүүдэд хамаатай болохыг тодорхойлоход тусалж чадна.

NTFS хуваалтыг бас таниулан холбохдоо өмнөхтэй ижил аргаар mount_ntfs(8) тушаал хэрэглэж болно.

## 2.11.3. Алдааг олж засварлахтай холбоотой асуултууд болон хариултууд

### 2.11.3.1. Миний систем ачаалж тоног төхөөрөмж шалгаж байх үедээ гацах юм уу эсвэл суулгах үед хачин ажиллаад байна, эсвэл уян дискний хөтчийг шалгахгүй байна.

FreeBSD нь системийн ACPI үйлчилгээг ачаалах явцад илрүүлсэн бол түүнийг i386, amd64 болон ia64 тавцангууд дээр өргөнөөр ашигладаг. Харамсалтай нь ACPI драйвер болон системийн эх хавтан ба BIOS-д алдаанууд байсаар байгаа билээ. Гуравдагч шатны ачаалагч дуудагч дээр `hint.acpi.0.disabled` тохиргоог тохируулснаар ACPI-ийн хэрэглээг хааж болдог:

```
set hint.acpi.0.disabled="1"
```

Энэ нь систем ачаалах тоолонд өөрчлөгдөх учир /boot/loader.conf файлд `hint.acpi.0.disabled="1"` мөрийг нэмж өгөх шаардлагатай. Ачаалагч дуудагчийн талаар дэлгэрэнгүй мэдээллийг Ерөнхий агуулга хэсгээс олж болно.

### 2.11.3.2. FreeBSD-г суулгасны дараа хатуу дискнээс анх удаагаа ачаалах гэхэд цөм дуудагдаж миний тоног төхөөрөмжийг шалгасан боловч иймэрхүү мэдээлэл гаргаад зогсоод байна:

Ачаалах диск системийн эхний диск биш тохиолдолд гардаг асуудал аль эртнийх байдаг. BIOS нь FreeBSD-д өөр дугаарлах аргыг хэрэглэдэг бөгөөд аль дугаар нь алинтай таардгийг зөвөөр тохируулах төвөгтэй байдаг.

Ачаалах диск нь систем дээр эхний диск биш бол FreeBSD-д түүнийг олоход тусламж хэрэгтэй болдог. Энд хоёр нийтлэг тохиолдол байдаг бөгөөд аль ч тохиолдолд та FreeBSD-д root файлын систем хаана байгаа хэлж өгөх шаардлагатай. Та BIOS-ийн дискний дугаар, дискний төрөл болон тэр төрөлд зориулсан FreeBSD-ийн дискний дугаарыг зааж өгөн үүнийг хийж болно.

Эхнийх нь та хоёр IDE дисктэй бөгөөд диск бүр өөр өөрийн IDE шугамнууд дээр мастер болон тохируулагдсан байх ба FreeBSD-г хоёр дахь дискнээс ачаалахыг хүсэж байгаа тохиолдол юм.

BIOS нь эдгээрийг диск 0 болон диск 1 гэж хардаг бол FreeBSD нь тэдгээрийг ad0 болон ad2 гэж хардаг.

FreeBSD нь `ad` төрлийн BIOS диск 1 дээр байгаа бөгөөд FreeBSD-ийн дискний дугаар нь 2 юм. Тэгэхээр та ингэж хэлж өгөх хэрэгтэй-:

```
1:ad(2,a)kernel
```

Хэрэв та анхдагч шугаман дээр боол дисктэй бол дээр дурдсан нь шаардлагагүй юм (тэгээд бас маш буруу юм).

Хоёр дахь нь та систем дээрээ нэг буюу түүнээс олон IDE дискнүүдтэй бөгөөд SCSI дискнээс ачаалах тохиолдол юм. Энэ тохиолдолд FreeBSD-ийн дискний дугаар нь BIOS-ийн дискний дугаараас бага байдаг. Хэрэв хоёр IDE дисктэй бөгөөд бас SCSI дисктэй бол SCSI диск нь `da` төрлийн BIOS диск 2 ба FreeBSD-ийн дискний дугаар 0 байна. Тэгэхээр та:

```
2:da(0,a)kernel
```

гэж FreeBSD-д систем дэх эхний SCSI диск болох BIOS диск 2-оос ачаалахыг хүсэж байгаагаа хэлж өгнө. Хэрэв танд зөвхөн нэг IDE диск байсан бол дээрхийн оронд та `1:`-ийг харах байсан билээ.

Та ашиглах зөв утгуудыг тодорхойлсныхоо дараа стандарт текст засварлагч ашиглан /boot.config файлд бичиж өгсөн шигээ тушаалыг хийж өгч болно. Өөрөөр зааварлаагүй л бол FreeBSD нь энэ файлын агуулгыг `boot:` хүлээх мөрний анхдагч хариулт болгон ашиглах болно.

#### 2.11.3.3. FreeBSD-г суулгасны дараа хатуу дискнээс анх удаагаа ачаалах гэхэд ачаалах цэсэн дээр Ачаалагч Менежерийн хүлээх мөр зөвхөн F?-г хэвлээд ачаалалт цааш үргэлжлэхгүй байна.

Таныг FreeBSD-г суулгаж байхад хатуу дискний геометр Хуваалтын засварлагч дээр буруу тохируулагдсан байна. Хуваалтын засварлагч уруу буцаж ороод өөрийн хатуу дискний жинхэнэ геометрийг зааж өгөх хэрэгтэй. Та зөв геометртэйгээр FreeBSD-г эхнээс нь дахин суулгах шаардлагатай.

Хэрэв та өөрийн машины хувьд зөв геометрийг ерөөсөө олж чадахгүй байгаа бол ийм зөвлөгөө өгье: дискний эхэнд жижиг MS-DOS® хуваалт суулгаад дараа нь FreeBSD-г суулгах хэрэгтэй. Суулгалтын програм нь MS-DOS® хуваалтыг харж түүнээс зөв геометрийг олж авахыг оролддог. Ингэснээр энэ нь ихэвчлэн ажилладаг билээ.

Дараах зааврыг зөвлөхөө больсон бөгөөд энд лавлагааны зориулалтаар оруулав:

Хэрэв та (ирээдүйд) MS-DOS®, Линукс эсвэл бусад үйлдлийн системтэй нийцтэй байх эсэх нь хамаагүй цэвэр FreeBSD сервер эсвэл ажлын станц суулгаж байгаа бол FreeBSD нь хамгийн эхний сектороос эхлээд сүүлийн сектор хүртэл бүхэл дискийг ашигладаг стандарт бус тохируулгыг сонгож бүхэл дискийг (хуваалтын засварлагч дээр A) ашиглах сонголт бас танд байдаг. Энэ нь геометрийн бүх л тооцооллыг орхих боловч дискэн дээр FreeBSD-ээс өөр бусад

үйлдлийн системийг хэзээ ч ажиллуулахгүй гэж тооцоогүй л бол зарим талаараа хязгаарлагдмал байдаг.

#### 2.11.3.4. Систем миний ed(4) сүлжээний картыг олсон боловч device timeout гэсэн алдаа гарсаар байх юм.

Таны карт /boot/device.hints файл дээр зааснаас өөр IRQ-г ашиглаж байж магадгүй юм. ed(4) драйвер нь анхдагчаар "soft" буюу зөөлөн тохиргоог (MS-DOS® дээр EZSETUP-ийг ашиглан оруулсан утгуудыг) ашигладаггүй боловч хэрэв таныг интерфэйсийн хувьд -1-г зааж өгөх юм бол програм хангамжийн тохиргоог ашиглах болно.

Карт дээр байгаа жижиг шилжүүлэгчийг тогтсон буюу хатуу тохиргоо руу (шаардлагатай бол цөмийн тохиргоонуудыг өөрчлөн) тохируулах юм уу эсвэл зөвлөгөөг `hint.ed.0.irq="-1"` гэж IRQ-г -1 болгож тохируулан зааж өгөх хэрэгтэй. Энэ нь цөмд зөөлөн тохиргоог ашиглахыг хэлж өгөх болно.

Өөр нэг шалтгаан нь таны карт IRQ 2-той хуваалцсан IRQ 9 дээр байж (ялангуяа IRQ 2-г ашигладаг VGA карттай бол) байнга асуудал болж байдаг байж болох юм. Та аль болох IRQ 2 эсвэл 9-ийг ашиглах ёсгүй юм.

#### 2.11.3.5. sysinstall-ийг X11 терминал дээр ашигласан тохиолдолд цайвар саарал дээр шар фонтыг уншихад хэцүү байдагcolor contrast. Энэ програмын хувьд илүү өндөр нягтралтай үзүүлэх боломж бий юу?

Хэрэв та X11-ийг суулгасан бөгөөд sysinstall-ийн сонгосон анхдагч өнгөнүүд нь xterm(1) эсвэл rxvt(1) ашиглах үед текстийг унших боломжгүй болгоод байвал илүү бараан саарал болгохын тулд өөрийн ~/.Xdefaults файлдаа дараах `XTerm*color7: #c0c0c0` мөрийг нэмэх хэрэгтэй.

## 2.12. Суулгацын нэмэлт гарын авлага

Энэ хэсэгт FreeBSD-г хэрхэн онцгой тохиолдолд суулгах талаар өгүүлэх болно.

### 2.12.1. FreeBSD-г дэлгэц юм уу гар үгүй систем дээр суулгах нь

Энэ төрлийн суулгацыг FreeBSD-г суулгаж байгаа машин нь дэлгэцгүй юм уу тэр байтугай дэлгэцийн залгуургүй байдаг учраас "headless install буюу толгойгүй суулгац" гэж нэрлэдэг. Яаж ингэж болдог юм бол? гэж та гайхаж байвал, serial console буюу цуваа консол хэрэглэснээр ийм суулгац хийж болдог. Цуваа консолоор өөр машиныг дэлгэц болон гар болгож ашиглаж болдог. Үүнийг гүйцэтгэхийн тулд Эхлэн ачаалах төхөөрөмжийг бэлдэх дээр заасан зааврын дагуу суулгац USB санах ойн зөөгчийг бэлдэх юм уу эсвэл зөв ISO дүрсийг татаж авах хэрэгтэй. Суулгацын CDROM үүсгэх хэсгээс үзнэ үү.

Эдгээр зөөгчийг өөрчилж цуваа консолоос эхлэн ачаалахын тулд дараах алхмуудыг биелүүлэх хэрэгтэй (хэрэв та CDROM ашиглах хүсэлтэй байгаа бол эхний алхмыг алгасаж болно):

1. Эхлэн ачаалагч USB зөөгчийг цуваа консол уруу холбогдохыг зөвшөөрөх

Хэрэв та урьд нь USB зөөгчөөс эхлэн ачаалж байсан бол, FreeBSD нь ердийн энгийн суулгацын горимоор эхлэн ачаалах болно. Гэхдээ бид нар суулгацаа цуваа холболт уруу орж эхлүүлэх ёстой билээ. Ингэхийн тулд FreeBSD уруу USB дискийг mount(8) тушаал ашиглаж таниулж холбох хэрэгтэй.

```
# mount /dev/da0a /mnt
```

Өөрийн нөхцөл байдалд зориулж төхөөрөмжийн цэг болон холбох цэгийг тааруулаарай.

Одоо та зөөгчийг холбосон болохоор USB зөөгчийг цуваа консолоос эхлэн ачаалахыг зааж өгөх ёстой. Та USB зөөгчийн файлын системийн loader.conf файлыг системийн консолийг цуваа консол гэж зааж байгаа мөрөн дээр нэмж өгөх шаардлагатай:

```
# echo 'console="comconsole"' >> /mnt/boot/loader.conf
```

Ингээд та USB зөөгчөө зөв тохируулчихсан учраас таниулсан дискээ umount(8) тушаалаар буцааж салгах хэрэгтэй:

```
# umount /mnt
```

Одоо USB зөөгчөө салгаад энэ процедурын гурав дахь шат руу шууд очиж болно.

2. Цуваа консол руу ачаалахаар суулгалтын CD-г идэвхжүүлэх

   Хэрэв та суулгалтын ISO дүрснээс (Суулгацын CDROM үүсгэх-с үзнэ үү) өөрийн хийсэн CD-ээс ачаалж байгаа бол FreeBSD ердийн суулгалтын горим руу ачаалах болно. Бид FreeBSD-г цуваа консол руу ачаалахыг хүсэж байгаа. Ингэхийн тулд та CD-R зөөгч рүү шарахаасаа өмнө ISO дүрсийг задлан засаж дахин үүсгэх ёстой.

   Суулгалт ISO дүрсийг жишээ нь FreeBSD-12.0-RELEASE-i386-disc1.iso хадгалсан FreeBSD системээс tar(1) хэрэгслийг ашиглан бүх файлыг задалж авна:

   ```
   # mkdir /path/to/headless-iso
   # tar -C /path/to/headless-iso -pxvf FreeBSD-12.0-RELEASE-i386-disc1.iso
   ```

   Та одоо цуваа консол руу ачаалах суулгалтын зөөгчийг тохируулж өгөх ёстой. Та задлагдсан ISO дүрснээс авсан loader.conf файлдаа системийн консолийг цуваа консол болгон тохируулах мөр нэмж өгөх ёстой:

   ```
   # echo 'console="comconsole"' >> /path/to/headless-iso/boot/loader.conf
   ```

   Дараа нь бид өөрчлөгдсөн модноос шинэ ISO дүрсээ үүсгэж болно. sysutils/cdrtools портын mkisofs(8) хэрэгслийг ашиглана:

```
# mkisofs -v -b boot/cdboot -no-emul-boot -r -J -V "Headless_install" \
        -o Headless-FreeBSD-12.0-RELEASE-i386-disc1.iso /path/to/headless-iso
```

Одоо та өөрийн ISO дүрсийг зөв тохируулсан болохоор өөрийн дуртай шарагч програмаар CD-R руу шарж болно.

3. Null-modem кабелиар холбох

   Та одоо хоёр машинаа null-modem кабелиар холбох хэрэгтэй. Уг хоёр машины цуваа холболтын залгууранд нь тус тусад нь залгах хэрэгтэй. *Ердийн цуваа залгуур энэ тохиолдолд ажилладаггүй* бөгөөд дундаа сэлгэж залгагдсан null-modem залгуур танд хэрэгтэй.

4. Суулгац эхлүүлэх гэж шинээр ачаалах

   Одоо суулгацаа эхлүүлэх цаг боллоо. USB санах ойн зөөгчөө толгойгүй суулгалт хийж байгаа машиндаа хийгээд машинаа асаана. Хэрэв та бэлтгэсэн CDROM ашиглаж байгаа бол машинаа асаагаад дискээ хийн ачаалах хэрэгтэй.

5. Толгойгүй машин уруугаа холбогдох

   Та одоо тэр машин уруугаа cu(1) тушаал хэрэглэж холбогдох хэрэгтэй:

   ```
   # cu -l /dev/cuau0
   ```

Ингээд л боллоо! Та одоо нөгөө толгойгүй машинаа `cu` тушаалын хэсгээр удирдах боломжтой боллоо. Энэ нь цөмийг дуудах бөгөөд дараа нь ямар төрлийн терминал хэрэглэхийг асуух болно. Тэр үед нь FreeBSD color console буюу өнгөт консол гэдгийг сонгоод цааш нь суулгацаа үргэлжлүүлээрэй!

# 2.13. Өөртөө зориулж тусгай суулгацын төхөөрөмж бэлдэх нь

Дахин нуршихаас сэргийлж таны худалдаж авсан юм уу бэлдэж авсан CDROM эсвэл DVD нарыг "FreeBSD диск" гэж нэрлэв.

Заримдаа танд өөрийн гэсэн өөрчилж тохируулсан FreeBSD суулгацын төхөөрөмжөөс суулгах хэрэг гарч болзошгүй. Энэ нь физик төхөөрөмж болох бичлэгт хальс юм уу эсвэл sysinstall дээр зааж өгсөн FTP хуудас эсвэл MS-DOS® хуваалтад буй файлаас суулгаж болзошгүй.

Жишээ нь:

- Танд маш олон компьютер холбогдсон дотоод сүлжээ байгаа бөгөөд зөвхөн нэг л FreeBSD диск байж болох юм. Та дотоод FTP хуудас үүсгээд тэрэндээ FreeBSD дискээ байрлуулж, уг хуудаснаас бусад компьютерууд холбогдож суулгац хийж болохоор тохируулж болно.

- Танд FreeBSD диск байгаа боловч таны CD/DVD төхөөрөмж уг дискийг танихгүй харин MS-DOS® / Windows® дээр таньдаг байг. Та уг компьютерийнхаа DOS хуваалт дээрээ FreeBSD суулгацыг хуулаад дараа нь уг файл уруу FreeBSD-г суулгах үед хандаж болно.
- Таны суулгахыг хүссэн компьютерт CD/DVD төхөөрөмж болон сүлжээний картын аль нь ч байхгүй бөгөөд та зөвхөн "Laplink-style" цуваа юм уу зэрэгцээ холболтын кабелиар холбогдож гүйцэтгэж болно.
- Та FreeBSD суулгаж болдог бичлэгт хальс бэлдэж болно.

### 2.13.1. Суулгацын CDROM үүсгэх

FreeBSD төсөл нь хувилбар гаргах үедээ хамгийн багадаа хоёр ширхэг CDROM дүрсийг ("ISO images буюу дискний дүрс") дэмжигдсэн архитектур бүрийн хувьд бэлтгэдэг. Хэрэв танд CD бичигч байвал эдгээр буулгац дүрсийг CD дээр буулган бичиж ("шарж") болдог. Хэрэв танд CD бичигч болон хямд үнэтэй интернэт сайн холболт байвал энэ нь FreeBSD-г суулгах хамгийн амархан арга болно.

1. Зөв ISO Images буюу буулгац дүрсийг татаж авах

   Хувилбар бүрд зориулсан ISO буулгац дүрсийг ftp://ftp.FreeBSD.org/pub/FreeBSD/ISO-IMAGES-arch/version хаягаас юм уу эсвэл хуулбар толин тусгалуудаас татаж авч болно. Машины төрөл болон хувилбарыг *arch* болон *version* гэсэн хэсгүүдэд тус тусад нь орлуулж бичих хэрэгтэй.

   Уг сан нь дараах буулгац дүрснүүдийг агуулж байдаг:

   *Хүснэгт 4. FreeBSD 8.X ISO буулгац дүрсний нэр болон агуулга*

| **Файлын нэр** | **Агуулга** |
|---|---|
| FreeBSD-version-RELEASE-arch-bootonly.iso | Энэ CD дүрс нь CD-ROM хөтчөөс ачаалж суулгалтыг эхлүүлэх боломжийг олгодог боловч FreeBSD-г уг CD-с суулгах боломжгүй. Та энэ CD-с ачаалсны дараа сүлжээгээр дамжуулж (жишээ нь FTP серверээс) суулгах шаардлагатай. |
| FreeBSD-version-RELEASE-arch-dvd1.iso.gz | Энэ DVD дүрс нь үндсэн FreeBSD үйлдлийн систем, урьдчилан бүтээсэн багцын цуглуулга болон баримтжуулалтыг суулгахад шаардлагатай бүх зүйлсийг агуулсан байдаг. Энэ нь "livefs" дээр үндэслэсэн аврах горим руу ачаалахыг бас дэмждэг. |

| Файлын нэр | Агуулга |
| --- | --- |
| FreeBSD-version-RELEASE-arch-memstick.img | Энэ дүрсийг USB санах ойн зөөгч рүү бичээд USB хөтчөөс ачаалж чаддаг машинуудад суулгаж хэрэглэж болно. Энэ нь "livefs" дээр үндэслэсэн аврах горим руу ачаалахыг бас дэмждэг. Баримтжуулалтын багцуудыг агуулдаг боловч бусад багцуудыг агуулдаггүй. |
| FreeBSD-version-RELEASE-arch-disc1.iso | Энэ CD дүрс нь үндсэн FreeBSD үйлдлийн систем болон баримтжуулалтын багцуудыг агуулдаг. Бусад багцуудыг агуулдаггүй. |
| FreeBSD-version-RELEASE-arch-disc2.iso | Гуравдагчдын програм багцуудыг ихээр багтаасан CD дүрс. Энэ дүрс нь FreeBSD 8.*X* хувилбаруудын хувьд байдаггүй. |
| FreeBSD-version-RELEASE-arch-disc3.iso | Гуравдагчдын програм багцуудыг ихээр багтаасан өөр нэг CD дүрс. Энэ дүрс нь FreeBSD 8.0 болон түүнээс дараагийн хувилбаруудын хувьд байдаггүй. |
| FreeBSD-version-RELEASE-arch-docs.iso | FreeBSD-ийн баримтжуулалт. Энэ дүрс нь FreeBSD 8.*X*-н хувьд байдаггүй. |
| FreeBSD-version-RELEASE-arch-livefs.iso | Энэ CD дүрс нь "livefs" дээр тулгуурласан аврах горим руу ачаалахад зориулагдсан боловч уг CD-ээс суулгахад зориулагдаагүй. |

Та `bootonly` ISO юм уу эсвэл `disc1` дүрс хоёрын аль нэгийг татаж авах *ёстой*. `disc1` дүрс нь `bootonly` ISO дискний бүтцийг агуулж байдаг болохоор энэ хоёрыг хоёуланг татаж авч хэрэггүй.

Хэрэв Интернэт холболт хямд бол `bootonly` ISO-г ашиглаарай. Үүгээр та FreeBSD-г суулгаад, дараа нь гуравдагч програмуудыг интернэтээр татаж авч суулгаж болно (Програм суулгах. Багцууд болон портууд-с харна уу).

Та FreeBSD-г суулгах юм уу эсвэл дискэн дээр байгаа гуравдагчдын багц програмыг суулгахыг хүсвэл `dvd1` дүрсийг ашиглаарай.

Бусад дискнүүд нь хэрэв танд хурдтай интернэт холболт байхгүй л бол тийм чухал шаардлагатай биш.

2. CD-нүүдийг бичих

   Та хуулбар дүрс нарыг дискэн дээр бичих хэрэгтэй. Хэрэв та энэ алхмыг өөр FreeBSD систем дээр гүйцэтгэж байгаа бол Оптик зөөвөрлөгчийг (CD-үүд) үүсгэж ашиглах нь хэсгээс нэмэлт мэдээллийг үзэх хэрэгтэй (burncd болон cdrecord хэсгүүдэд зарим нь буй).

Хэрэв та энэ үйлдлийг өөр үйлдлийн систем дээр гүйцэтгэж байгаа бол, уг систем дээр байгаа CD бичигчээ удирддаг дурын програмыг нь ашиглаж бичих нь зүйтэй. Эдгээр дүрс нь стандарт ISO хэлбэрээр байгаа тул маш олон бичигч нар энэ төрлийг дэмждэг билээ.

Хэрэв өөрийн гэсэн FreeBSD хувилбар бүтээнэ гэх юм бол Хувилбарыг инженерчлэх мэдээллээс хараарай.

### 2.13.2. FreeBSD дисктэй дотоод FTP хуудас бэлтгэх нь

FreeBSD дискний зохион байгуулалт нь FTP хуудастайгаа ижил байрлуулагдсан буй. Ингэснээр танд дотоод FTP хуудас байгуулж сүлжээгээр FreeBSD-г суулгахад тун дөхөм болж өгдөг.

1. FTP хуудас эрхэлж байх ёстой FreeBSD компьютерт CDROM төхөөрөмж байх шаардлагатай бөгөөд /cdrom санд таниулж холбосон байх ёстой.

```
# mount /cdrom
```

2. anonymous FTP эрх /etc/passwd дотор үүсгэх хэрэгтэй. Ингэхийн тулд vipw(8) хэрэгслийг ашиглан /etc/passwd файлыг засварлахдаа дараах мөрийг нэмэх хэрэгтэй:

```
ftp:*:99:99::0:0:FTP:/cdrom:/nonexistent
```

3. /etc/inetd.conf дотор FTP service буюу үйлчилгээ зөвшөөрөгдсөн эсэхийг лавлах хэрэгтэй.

Ингээд одоо таны компьютертай сүлжээгээр холбогдож чадах машин болгон суулгац эхлүүлэхийн тулд, суулгах төхөөрөмжөө сонгох цэснээс "Other буюу бусад" гэдгийг сонгоод цааш нь FTP гэж сонгон `ftp://машины нэр` гэж оруулах хэрэгтэй.

Хэрэв таны FTP үйлчилгээг ашиглаж байгаа хэрэглэгчийн эхлэн ачаалах төхөөрөмж нь (голдуу уян диск байдаг) FTP дээр буй хувилбартай яг ижил хувилбар биш бол, sysinstall нь таны суулгацыг бүрэн гүйцэтгэж чаддаггүй. Хэрэв хувилбарууд ижил биш байсан ч гэсэн дарж бичихийг та хүсэж байгаа бол **Options** цэснээс distribution name буюу тархацын нэрийг any гэж бичих хэрэгтэй.

Энэ арга нь галт ханаар хамгаалагдсан дотоод сүлжээнд бол найдвартай ажилладаг. Энэ FTP үйлчилгээгээ интернэтээр дамжуулан өөр машинд (таны дотоод сүлжээнээс гадуурх сүлжээнд буй компьютерт) санал болговол зарим компьютерийн сүлжээ эвдэгч хүмүүсийн анхаарлыг татаж эвгүй үр дүнд хүргэж болзошгүй. Хэрэв та нууцлалын маш сайн дадлагатай биш л бол ингэж гадуурх сүлжээнд ил гаргахаас болгоомжлохыг бид танд зөвлөж

байна.

### 2.13.3. Суулгацын уян дискийг бэлдэх нь

Хэрэв таны компьютер өөр дэмжлэггүйгээс болоод юм уу эсвэл та юмыг хүндрүүлж хийхийг хүссэнээс ч болоод юм уу, суулгацыг уян дискнээс суулгах ёстой бол (энэ аргыг *хэрэглэхгүй* байхыг санал болгож байна), та эхлээд уян дискнүүдийг суулгацад бэлдэх ёстой.

Хамгийн багадаа л гэхэд base (үндсэн түгээлт) санд буй хоёртын файлуудыг багтаахад шаардлагатай тооны 1.44 MB-н дискнүүдийг бэлдэх ёстой. Хэрэв уян дискээ MS-DOS®-с бэлдэж байгаа бол MS-DOS®-н `FORMAT` командыг ашиглан дискнүүдийг шинэчилж бэлдэх *ёстой*. Хэрэв та Windows®-с бэлдэхээр бол, Explorer дээр дискийг форматлаж болдог ( A: төхөөрөмж дээр хулганы баруун товчийг дараад "Format" цэсийг сонгоорой).

Үйлдвэрээс урьдчилан бэлтгэж цэвэрлэсэн дискэнд итгэж *болохгүй шүү*. Найдвартай байхын тулд тэдгээрийг дахин цэвэрлэх хэрэгтэй. Сүүлийн үед маш олон хэрэглэгчид зөв цэвэрлэж янзлаагүй дискнээсээ болж маш их алдаа гарлаа гэж мэдээлсэн учраас бид ингэж танд анхааруулж байгаа билээ.

Хэрэв та уян дискээ MS-DOS® файл систем дээр биш харин өөр FreeBSD машин дээр бэлтгэж байсан ч гэсэн цэвэрлэх нь зөв санаа гэдгийг дахин сануулмаар байна. Та `bsdlabel` болон `newfs` тушаалуудыг ашиглаж UFS файлын систем суулгахыг хүсвэл ( 3.5" 1.44 MB диск дээр) дараах тушаалуудыг өгөх хэрэгтэй:

```
# fdformat -f 1440 fd0.1440
# bsdlabel -w -r fd0.1440 floppy3
# newfs -t 2 -u 18 -l 1 -i 65536 /dev/fd0
```

Дараа нь дискээ таниулаад жирийн файл систем шиг бичиж болно.

Уян дискнүүдээ цэвэрлэж бэлтгэсний дараа файлуудаа хуулах хэрэгтэй. Тархцын файлууд нь зохицох зорилгоор тааруулж хуваагддаг бөгөөд таван ширхэг 1.44 MB диск дээр багтахаар тохируулагдсан байдаг. Бүх уян диск болгондоо багтах хэмжээгээр нь тулгаж хуулах зарчмаар түгээлтийн бүх файлуудыг багтааж хуулах хэрэгтэй. Тархац бүр уян диск дээр : a:\base\base.aa, a:\base\base.ab гэх мэтчилэн хуулагдах ёстой.

Түгээлтийг татаж аван нийлүүлэх үедээ хичнээн нэмэлт хэсгүүдийг хайхаа мэдэхийн тулд суулгалтын програм уншдаг учраас base.inf файл нь base цуглуулгын эхний уян дискэнд орох ёстой.

Суулгацын явцад суулгах төхөөрөмжөө сонгох Media цэс гарах үед Floppy цэсийг сонгоод цаашаагаа суулгацаа үргэлжлүүлээрэй.

### 2.13.4. MS-DOS® хуваалтаас суулгах нь

MS-DOS® хуваалтаас суулгацыг бэлтгэхийн тулд, уг хуваалтын эх хэсэг дээр freebsd нэртэй сан үүсгэж дотор нь түгээлтийн файлуудыг хуулах хэрэгтэй. Жишээ нь, c:\freebsd. CDROM юм уу FTP хуудсанд байсан файлуудын сангийн бүтэц нь уг сан доторхтой ижил байх ёстой

бөгөөд хуулахын тулд MS-DOS®-н `xcopy` тушаалыг ашиглан CD-с хуулаарай. Жишээ нь, FreeBSD-н хамгийн бага суулгацыг гүйцэтгэхийн тулд:

```
C:\> md c:\freebsd
C:\> xcopy e:\bin c:\freebsd\bin\ /s
C:\> xcopy e:\manpages c:\freebsd\manpages\ /s
```

Энд C: дискийг сул зайтай гэж үзсэн бөгөөд E: диск дээр CDROM-г таниулсан гэж авч үзэв.

Хэрэв танд CDROM төхөөрөмж байхгүй бол тархац файлыг ftp.FreeBSD.org -с татаж авч болно. Тархац бүр өөрсдийн сан дотроо байгаа бөгөөд, жишээ нь, *base* тархац 12.0/base/ сан дотор буй.

MS-DOS®-с суулгахыг хүссэн тархац болгоныг (хэрэв танд хангалттай сул зай байгаа л бол) c:\freebsd сан дотор хуулах хэрэгтэй бөгөөд - зөвхөн хамгийн багаар суулгах зориулалттай файлууд л `BIN` сан дотор байх шаардлагатай.

### 2.13.5. Суулгацын бичлэг тууз бэлтгэх нь

Бичсэн туузнаас суулгах нь FTP юм уу CDROM-с суулгаснаас хамаагүй амархан арга байж болох юм. Суулгац програм нь бичлэг дээр tar хэлбэрээр нэгтгэж бичигдсэн байх ёстой. Суулгахыг хүссэн тархцаа авсны дараа туузан дээр шахаж бичих хэрэгтэй:

```
# cd /freebsd/distdir
# tar cvf /dev/rwt0 dist1 ... dist2
```

Суулгацыг гүйцэтгэх явцдаа туузан дээр түр зуур хэрэглэхээр (сонголтоос хамаарч) туузан дээрх файлуудыг *бүхлээр* нь хуулахад багтаахаар сул зайг үлдээх ёстой. Шуурхай санамсаргүй хандалт хийгдэж болдоггүй учраас туузан дээр их хэмжээний сул зай бэлэн байлгах шаардлага гардаг юм.

Суулгацыг эхлүүлэх үед уян дискнээс эхлүүлэхээсээ *өмнө* туузыг төхөөрөмж рүү нь хийсэн байх ёстой. Ингэхгүй бол суулгац олохгүй байх магадлалтай.

### 2.13.6. Сүлжээгээр суулгахаасаа өмнө

Сүлжээгээр суулгах гурван төрлийн арга бий. Эдгээр нь Ethernet сүлжээ (стандарт Ethernet төхөөрөмж), Цуваа холболт (PPP) эсвэл Зэрэгцээ холболт (PLIP (laplink кабель)) юм.

Сүлжээний картаа ашиглаж интернэт суулгах арга бол мэдээж хамгийн сайн сонголт байх болно! FreeBSD нь ихэнх сүлжээний картуудыг дэмжиж таньдаг бөгөөд Hardware Notes буюу төхөөрөмжийн тухай мэдээлэл хэсэгт дэмждэг картуудыг (мөн тэдгээрт шаардагдах тохируулгатай нь) жагсаасан буй. Хэрэв та аль нэгэн дэмжигдсэн PCMCIA сүлжээний карт хэрэглэж байвал зөөврийн компьютераа асаахаасаа *өмнө нь* залгаарай. Харамсалтай нь одоогоор PCMCIA төрлийн картуудыг явцын дунд шууд залгаж хараахан чадахгүй байгаа билээ.

Та мөн сүлжээнийхээ IP хаягаа салгаж ангилсан netmask буюу сүлжээний шүүлт хаягтай нь хамт мэдэж байх ёстой. Хэрэв та PPP холболт ашиглаж холбогдож байгаа бол, ISP газраас тань автоматаар хаяглалт хийдэг болохоор тогтмол зааж өгсөн IP хаяг бүү зоож тохируулаарай. Таны сүлжээ тохируулагч тань ямар ямар тохируулга шаардлагатайг танд мэдэгдэх байх. Хэрэв та IP хаяг биш харин ямар нэгэн серверийн нэр оруулж тохируулах ёстой бол, танд бас name server буюу серверүүдийн нэрүүдийг зохицуулагч серверийн нэрийг мөн магадгүй gateway серверийн хаягийг бас оруулах хэрэгтэй болох байх (хэрэв та PPP хэрэглэж байгаа бол энэ нь таны интернэтээр хангагч байгууллагын тань IP хаяг байх ёстой). Хэрэв та HTTP проксигоор дамжин FTP-нээс суулгац хийх ёстой бол, та мөн проксигийн хаягийг оруулж өгөх хэрэгтэй. Хэрэв та эдгээр олон асуултанд хариулж мэдэхгүй байгаа тохиолдолд, сүлжээ тохируулагчаасаа юм уу эсвэл ISP-аасаа суулгацаа эхлүүлэхээсээ *өмнө* лавлаж асуух нь чухал.

Хэрэв та модем хэрэглэж байгаа бол PPP таны бараг цорын ганц сонголт байх болов уу. Суулгацаа бүр эхлүүлэхээсээ өмнө интернэт хөтлөгч байгууллага уруугаа холбогдох мэдээллээ сайн бэлтгэж авсан байх хэрэгтэй.

Хэрэв та ISP руугаа PAP эсвэл CHAP хэрэглэж холбогддог бол (өөрөөр хэлбэл, ISP уруугаа ямар нэгэн скрипт буюу гүйцэтгэдэг бичлэг хэрэглэлгүйгээр Windows®-с холбогдож байвал), ppp-н тушаал оруулах мөрөнд `dial` гэж оруулахад болно. Өөр тохиолдолд, PPP хэрэглэж утасдах горим нь тун энгийн терминал үйлчилгээгээр хангагдсан байдаг болохоор, та ISP руугаа модемондоо тохирсон "AT тушаал" хэрэглэж холбогдох хэрэгтэй. Хэрэглэгчийн ppp бүртгэлийн гарын авлага эсвэл FAQ буюу байнга асуудаг асуулт хариулт хэсгээс нэмэлт мэдээллийг харж болно. Хэрэв танд хүндрэл тулгарвал, `set log local …` тушаал ашиглаж ерөнхий явцын бүртгэл бичлэгийг дэлгэцэн дээрээ зэрэг хянаж болно.

Хэрэв та өөр FreeBSD машин уруу шууд залгаж холбогдох боломжтой бол "laplink" зэрэгцээ кабелиар гүйцэтгэж болох юм. Зэрэгцээ холболтоор өгөгдөл нь цуваа холболтыг бодвол арай хурдан дамжигдах (50 кбайт/сек хүртэл) бөгөөд, ингэснээр илүү түргэн суулгацаа гүйцэтгэж болох юм.

#### 2.13.6.1. NFS-р суулгахаасаа өмнө

NFS суулгац бол харьцангуй хурдан шулуухан байдаг. NFS сервер дээрээ FreeBSD тархалтынхаа хуулбарыг хуулаад дараа нь суулгац гүйцэтгэх төхөөрөмжийг сонгох цэсэн дээр NFS гэж зааж өгөхөд болно.

Хэрэв сервер зөвхөн "заагдсан порт" ашиглах ёстой бол (Sun төрлийн компьютерт энэ нь анхнаасаа заагдсан байдаг шиг), та **Options** цэсэн дээр `NFS Secure` сонголтыг сонгож тохируулаад суулгацаа цааш нь үргэлжлүүлээрэй.

Хэрэв таны сүлжээний карт тийм сайн биш бөгөөд хурдан дамжуулалт хийдэггүйг та мэддэг бол `NFS Slow` гэдгийг бас сонгож тохируулаарай.

NFS суулгацыг ажиллуулахын тулд сервер нь дэд санг таньдаг байх ёстой. Жишээ нь, хэрэв таны FreeBSD 12.0 тархац: ziggy:/usr/archive/stuff/FreeBSD санд байгаа бол, `ziggy` нь /usr эсвэл /usr/archive/stuff гэх мэтчилэн дамжиж таниулах биш харин шууд /usr/archive/stuff/FreeBSD санг таньж байхаар тохируулагдах ёстой.

FreeBSD-н /etc/exports файл дотор үүнийг тохируулахын тулд `-alldirs` гэсэн сонголт оруулж

өгдөг. Бусад NFS серверүүд өөр зохион байгуулалттай байж болох юм. Хэрэв та `permission denied буюу хандах эрх зөвшөөрөгдөхгүй` гэсэн мэдээлэл серверээс хүлээж авбал, та саяны сонголтыг оруулж өгөөгүй байна гэсэн үг юм.

# Chapter 3. FreeBSD 9.X болон түүнээс хойшхи хувилбаруудыг суулгах нь

## 3.1. Ерөнхий агуулга

FreeBSD нь текст дээр суурилсан, суулгахад хялбар програмтай ирдэг. FreeBSD 9.0-RELEASE болон түүнээс хойшхи хувилбарууд bsdinstall гэсэн суурилуулалтын програм ашигладаг бөгөөд FreeBSD 9.0-RELEASE-с өмнөх хувилбарууд sysinstall-г суулгахдаа ашигладаг. Энэ бүлэг нь bsdinstall-г ашиглахыг тайлбарлах болно. sysinstall-г FreeBSD 8.X болон түүнээс өмнөх хувилбар суулгах нь-н хэрэглээ бүлэгт бичигдсэн байгаа.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD суулгац бүхий зөөвөрлөгч хэрхэн бэлдэх талаар.
- FreeBSD нь хатуу дискийг хэрхэн хуваах ханддаг талаар.
- bsdinstall-г хэрхэн эхлүүлэх талаар.
- bsdinstall-н асуусан асуултууд, тэдгээрийн утга болон хэрхэн хариулах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- Таны суулгах гэж байгаа FreeBSD-н хувилбар дээрх тоног төхөөрөмжийн дэмжлэгийг уншаад таны төхөөрөмж дэмжигдсэн эсэхийг шалгаарай.

Ерөнхийдөө энэ суулгалтын заавар i386™ ("PC нийцэт") төрлийн архитектурт зориулагдсан. Шаардлагатай тохиолдолд бусад тавцангуудад зориулсан зааврыг үзүүлэх болно. Энд үзүүлсэн зүйл болон суулгагчийн хооронд ялгаа бага байж болох учир яг үгчилсэн заавар гэж ойлголгүйгээр энэ бүлгийг ерөнхий заавар хэлбэрээр ашиглаарай.

## 3.2. Тоног төхөөрөмжийн шаардлагууд

### 3.2.1. Минимум тохиргоо

FreeBSD-г суулгах хамгийн бага тохиргоо нь FreeBSD-н хувилбар болон тоног төхөөрөмжийн архитектураас хамаардаг.

Энэ мэдээллийн дүгнэлт дараагийн хэсгүүдэд гарна. FreeBSD суулгах аргаас хамаарч танд дэмжигдсэн CDROM хөтөч болон зарим тохиолдолд сүлжээний адаптер хэрэг болж магадгүй. Эдгээрийг Суулгалтын зөөвөрлөгчийг бэлдэх хэсэгт үзүүлнэ.

#### 3.2.1.1. FreeBSD/i386

FreeBSD/i386 нь 486 буюу түүнээс илүү процессор болон хамгийн багаар бодоход 64 MB RAM шаарддаг. Хамгийн багаар бодоход 1.1 GB хэмжээтэй дискний сул зай минимум суулгацад хэрэгтэй байдаг.

Хуучин компьютерууд дээр илүү хурдан процессор суулгаснаас илүүтэй RAM болон хатуу дискийн зайг нэмэгдүүлэх нь үр дүнтэй байдаг.

#### 3.2.1.2. FreeBSD/amd64

FreeBSD/amd64 ажиллах хоёр төрлийн процессор байдаг. Эхнийх нь AMD Athlon™64, AMD Athlon™64-FX, AMD Opteron™ эсвэл эдгээрээс илүү сайн процессоруудыг оруулсан AMD64 юм.

FreeBSD/amd64-г ашигладаг дараагийн нэг төрлийн процессор бол Intel® EM64T архитектурыг ашигладаг процессорууд юм. Эдгээр процессоруудад Intel® Core™ 2 Duo, Quad, Extreme процессорын гэр бүл, Intel® Xeon™ 3000, 5000, болон 7000 серийн процессорууд ба Intel® Core™ i3, i5 ба i7 процессорууд багтдаг.

Хэрэв танд nVidia nForce3 Pro-150 дээр тулгуурласан машин байгаа бол BIOS-ийн тохиргоог ашиглан IO APIC-г хаах хэрэгтэй. Хэрэв үүнийг хийх тохиргооны боломж байхгүй бол ACPI-г хаагаарай. Pro-150 чип нь алдаатай бөгөөд үүнийг давах аргыг бид одоогоор олоогүй байгаа юм.

#### 3.2.1.3. FreeBSD/powerpc Apple® Macintosh®

USB-н дэмжлэгтэй Apple® Macintosh®-ийн сүүлийн үеийн системүүд бас дэмжигдсэн. Олон CPU-тай машинууд дээр SMP дэмжигдсэн.

32 битийн цөм нь зөвхөн эхний 2 GB RAM-г ашигладаг. FireWire® нь Цэнхэр болон Цагаан PowerMac G3 дээр дэмжигдээгүй.

#### 3.2.1.4. FreeBSD/sparc64

FreeBSD/sparc64 дэмжигддэг системүүдийн жагсаалт FreeBSD/sparc64 төслийн хуудаснаа бий.

FreeBSD/sparc64-ийн хувьд тусдаа диск хэрэгтэй. Одоогоор өөр үйлдлийн системтэй дискийг хуваалцан хэрэглэх боломжгүй байгаа.

### 3.2.2. Дэмжигдсэн тоног төхөөрөмж

FreeBSD-н хувилбар дэмжигддэг тоног төхөөрөмжүүдийн талаар Hardware Notes файлд бий. Ихэвчлэн HARDWARE.TXT гэсэн нэртэй байдаг бөгөөд хувилбар байгаа дискийн root санд байрладаг. Дэмжигдсэн тоног төхөөрөмжийн жагсаалтын хуулбарууд FreeBSD веб сайтын Хувилбарын мэдээлэл хуудсанд бас байгаа.

## 3.3. Суулгахын өмнөх ажлууд

### 3.3.1. Өгөгдлөө нөөцөл

FreeBSD суулгах компьютер дээрх чухал өгөгдлөө нөөцөлж авах хэрэгтэй. Цааш үргэлжлүүлэхээсээ өмнө нөөцөө зөв ажиллаж байгаа эсэх дээр тест хийгээрэй. FreeBSD-н суулгалтын програм дискэд өөрчлөлт хийхээс өмнө асууна, гэхдээ нэгэнт процесс эхэлсэн бол буцаах боломжгүй.

### 3.3.2. FreeBSD-г хаана суулгахаа шийд

Хэрэв FreeBSD нь зөвхөн суулгах ганц систем бөгөөд бүх хатуу дискийг бүхэлд нь ашиглахаар зөвшөөрөгдөх бол үлдсэн хэсгийг алгасаж болно. Гэхдээ хэрэв FreeBSD нь өөр үйлдлийн системтэй хамт ашиглагдах бол дискийн мэдээллийг ойлгох нь суулгалтын явцад хэрэгтэй байдаг.

#### 3.3.2.1. FreeBSD/i386 болон FreeBSD/amd64-д зориулсан дискний мэдээлэл

Хатуу диск нь олон хэсгүүдэд хуваагдаж болно. Эдгээр хэсгүүдийг *partitions* буюу хуваалтууд гэдэг.

Дискийг хоёр янзын аргаар хуваах болдог. Уламжлалт *Master Boot Record* (MBR) нь дөрөв хүртэлх *primary partitions* буюу үндсэн хуваалт бүхий хуваалтын хүснэгтийг агуулдаг. (Түүхэн шалтгаанаас болоод FreeBSD үндсэн хуваалтуудыг *slices* буюу зүсмэлүүд гэдэг.) Том дискүүдийн хувьд зөвхөн дөрвөн хуваалт нь хязгаарлагдмал байдаг бөгөөд эдгээрийн нэг үндсэн хуваалтыг *extended partition* буюу өргөтгөсөн хуваалт болгож болдог. Дараа нь энэ өргөтгөсөн хуваалт дотроо *logical partitions* буюу логик хуваалтуудыг үүсгэж болдог. Энэ нь сонин сонсогдож болох юм, гэхдээ ийм л байдаг.

*GUID Partition Table* (GPT) нь шинэ бөгөөд дискийг хуваах илүү хялбар арга юм. GPT нь MBR хуваалтын хүснэгтийг бодох юм бол илүү уян хатан юм. Ердийн GPT шийдэл нь дискийн хувьд логик хуваалт гэх мэт хялбар биш аргуудыг ашиглалгүйгээр 128 хүртэлх хуваалтыг ашиглахыг зөвшөөрдөг.

Windows® XP гэх мэт хуучин үйлдлийн системүүд нь GPT хуваалтын схемтэй нийцтэй биш юм. Хэрэв FreeBSD нь тийм үйлдлийн системтэй цуг хэрэглэгдэх бол MBR хуваалтыг ашиглах хэрэгтэй.

FreeBSD-н стандарт ачаалал дуудагч нь үндсэн юм уу эсвэл GPT хуваалтыг шаарддаг. (FreeBSD-н эхлүүлэх процессын талаар дэлгэрэнгүйг FreeBSD-ийн Ачаалах процесс хэсгээс үзнэ үү). Хэрэв бүх үндсэн эсвэл GPT хуваалтууд ашиглагдаж байгаа бол FreeBSD-д зориулж нэгийг чөлөөлөх хэрэгтэй.

FreeBSD-н минимум суулгалт 1 GB дискний хэмжээ эзэлдэг. Гэхдээ энэ нь *хамгийн* минимум суулгалт бөгөөд бараг сул зайгүй байдаг. Арай илүү боломжит минимум хэмжээ нь график орчингүй бол 3 GB, график хэрэглэгчийн интерфэйс ашиглагдах бол 5 GB байдаг. Гуравдагч талуудын програмууд илүү хэмжээ шаарддаг.

Төрөл бүрийн хуваалт хийдэг чөлөөт болон арилжааны хэрэгслүүд байдаг. GParted Live нь GParted хуваалт засварлагч бүхий чөлөөт амьд CD юм. GParted бас өөр олон Линуксын Амьд CD түгээлтүүдэд байдаг.

Диск хуваах програмууд нь өгөгдлийг эвдэх аюултай. Дискийн хуваалтыг өөрчлөхөөсөө өмнө бүрэн нөөцийг авч зөв эсэхийг нь шалгах хэрэгтэй.

Microsoft® Vista хуваалтыг өөрчлөх төвөгтэй байдаг. Тийм үйлдэл хийх бол Vista-ийн суулгалтын CD хэрэгтэй байдаг.

*Жишээ 3. Байгаа хуваалтыг ашиглах*

Windows® компьютер нь 20 GB хуваалтаар хуваагдсан 40 GB дисктэй гэж үзье. Windows® нь тэдгээрийг C: ба D: гэдэг. C: хуваалт нь 10 GB өгөгдлөөс тогтох бөгөөд D: хуваалт нь 5 GB өгөгдлөөс бүтнэ.

D:-с C: рүү өгөгдөл шилжүүлснээр хоёр дахь хуваалтыг FreeBSD-д зориулан ашиглах боломжтой болгоно.

*Жишээ 4. Байгаа хуваалтыг багасгах*

Windows® компьютер нь ганц 40 GB дисктэй бөгөөд бүх дискийг нэг том хуваалт эзэлж байя. Windows® нь энэ 40 GB хуваалтыг нэг C: гэж харуулна. 15 GB ашиглагдаж байгаа. Зорилго бол Windows®-г 20 GB хуваалт дээр үлдээгээд FreeBSD-д зориулж 20 GB хуваалт бий болгох явдал юм.

Үүнийг хийх хоёр арга бий:

1. Windows® дээрх өгөгдлөө нөөцөл. Дараа нь Windows®-г суулгах явцдаа 20 GB хуваалттай болгоно.
2. Windows® хуваалтыг багасгаж чөлөөтэй болсон зай дээр FreeBSD-д зориулж шинэ хуваалт үүсгэхийн тулд GParted гэх мэт хуваалт өөрчилдөг хэрэгслүүд ашигла.

Өөр төрлийн үйлдлийн системүүд бүхий дискийн хуваалтууд нь тэдгээр үйлдлийн системүүдийн аль нэгийг тухайн үед ашиглах боломжийг олгодог. Нэгэн зэрэг олон үйлдлийн системийг ажиглах боломжийг олгодог өөр нэг аргын талаар Виртуалчлал хэсэгт бичигдсэн байгаа.

### 3.3.3. Сүлжээний талаар мэдээлэл цуглуулах

FreeBSD-н зарим суулгалтын аргууд нь файлуудыг татаж авахын тулд сүлжээний холболт ашиглах хэрэгтэй байдаг. Ethernet сүлжээнд (эсвэл кабел аль эсвэл Ethernet интерфэйстэй DSL модем) холбогдохын тулд суулгагч нь сүлжээний талаар зарим мэдээллийг асуух болно.

*DHCP* нь сүлжээг автоматаар тохируулах боломжийг ихэвчлэн олгодог. Хэрэв DHCP боломжгүй бол энэ сүлжээний мэдээллийг локал сүлжээний администратор юм уу эсвэл үйлчилгээ үзүүлэгчээс авах ёстой:

1. IP хаяг
2. Subnet маск
3. Анхдагч чиглүүлэгчийн IP хаяг
4. Локал сүлжээний домен нэр
5. DNS серверийн IP хаягууд

### 3.3.4. FreeBSD-н алдааны мэдээллийг шалгах

FreeBSD төсөл нь FreeBSD хувилбар бүрийнхээ хувьд аль болох алдаа мадаггүй, тогтворгүй байхыг хичээж ажилладаг боловч процессын явцад алдаанууд гардаг. Маш ховор тохиолдолд тэдгээр алдаанууд нь суулгалтын процессд нөлөөлдөг. Эдгээр асуудлуудыг илрүүлж засварладаг бөгөөд энэ талаар FreeBSD-н веб сайтын FreeBSD Errata хаяг дээр тэмдэглэгдсэн байдаг. Суулгалтад нөлөөлж болзошгүй асуудлууд байж болзошгүй учир суулгахаасаа өмнө алдааны хуудсыг шалгах хэрэгтэй.

Бүх хувилбарын талаарх мэдээлэл болон алдааны талаар FreeBSD веб сайтын хувилбарын мэдээлэл хэсгээс олж болно.

### 3.3.5. Суулгалтын зөөвөрлөгчийг бэлдэх

FreeBSD-н суулгалтыг FreeBSD суулгац бүхий CD, DVD, эсвэл USB санах ойн диск ашиглан компьютераа ачаалах замаар эхлүүлнэ. Суулгалтын програм нь өөр үйлдлийн системээс ажиллуулж болдог програм биш юм.

FreeBSD-н бүх суулгалтын файлуудыг агуулдаг стандарт суулгалтын зөөвөрлөгчөөс гадна *bootonly* буюу зөвхөн ачаалах хувилбар байдаг. Зөвхөн ачаалахад зориулсан суулгалтын зөөвөрлөгч нь суулгалтын файлуудгүй байдаг боловч суулгах явцдаа тэдгээрийг сүлжээгээр татаж авдаг. Зөвхөн ачаалахад зориулсан суулгалтын CD нь бага хэмжээтэй байдаг бөгөөд суулгах явцдаа зөвхөн хэрэгцээтэй файлуудыг татаж аван сүлжээний зурвасын ашиглалтыг багасгадаг.

FreeBSD вэб сайтад FreeBSD-н суулгалтын зөөвөрлөгчийн хуулбарууд байдаг.

Хэрэв танд CD, DVD, эсвэл USB санах ойн диск дээр FreeBSD-н хуулбар байгаа бол энэ хэсгийг алгасч болно.

FreeBSD-н CD ба DVD дүрс нь ачаалагдах боломжтой ISO файлууд юм. Суулгахад зөвхөн нэг CD эсвэл DVD хэрэгтэй байдаг. Одоо ажиллаж байгаа үйлдлийн систем дээрээ байгаа CD бичих програм ашиглан ISO дүрсийг ачаалагдах CD эсвэл DVD дээр шарна.

Ачаалагдах боломжтой санах ойн диск үүсгэхийн тулд дараах алхмуудыг хийнэ:

1. Санах ойн дискний дүрсийг олж авах

   FreeBSD 9.0-RELEASE ба түүнээс хойшхи хувилбаруудын хувьд Санах ойн дискний дүрсийг ISO-IMAGES/ сан дахь `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/arch/arch/ISO-IMAGES/version/FreeBSD-version-RELEASE-arch-memstick.img` хаягаас татаж авч болно. *arch* ба *version* гэдгүүдийг та өөрийн суулгахыг хүссэн архитектур болон хувилбарын дугаараар солиорой. Жишээ нь FreeBSD/i386 9.0-RELEASE-д зориулсан санах ойн дискний дүрс ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/i386/ISO-IMAGES/9.0/FreeBSD-9.0-RELEASE-i386-memstick.img хаяг дээр бий.

   FreeBSD 8.*X* болон өмнөх хувилбаруудын хувьд өөр сан ашиглагддаг. FreeBSD 8.*X* болон өмнөх хувилбаруудыг татаж авч суулгах талаар FreeBSD 8.X болон түүнээс өмнөх хувилбар суулгах нь хэсгээс харна

уу.

Санах ойн дискний дүрс .img өргөтгөлтэй байдаг. ISO-IMAGES/ сан олон төрлийн дүрснүүдээс тогтох бөгөөд танд хэрэгтэй байгаа нь суулгах FreeBSD-н хувилбар болон зарим тохиолдолд тоног төхөөрөмжөөс хамаардаг.

Үргэлжлүүлэхээсээ өмнө USB диск дээрх өгөгдлөө *нөөцөлж* аваарай, эс тэгвээс энэ алхам *устгах* болно.

2. Санах ойн диск рүү дүрс файлыг бичих

**Procedure: FreeBSD ашиглан дүрсийг бичих**

Доорх жишээ дүрс бичигдэх төхөөрөмжийг /dev/da0 гэж харуулж байна. Зөв төхөөрөмж сонгосон эсэх дээрээ болгоомжтой байгаарай, эс тэгвээс та өгөгдлөө устгаж болох юм.

a. dd(1) ашиглан дүрсийг бичих

.img файл нь ердийн файл *биш* юм. Энэ нь санах ойн дискний бүрэн гүйцэд тогтцын *дүрс* юм. Үүнийг ердийн файл бичдэг шигээр бичиж *болохгүй*, харин dd(1) ашиглан бичих ёстой юм:

```
# dd if=FreeBSD-9.0-RELEASE-i386-memstick.img of=/dev/da0 bs=64k
```

**Procedure: Windows® ашиглан дүрсийг бичих**

Гаралт дээрээ зөв хөтчийн үсгийг өгсөн эсэхээ нягтлаарай, эс тэгвээс та байгаа өгөгдлөө дарж устгаж болзошгүй.

a. Image Writer for Windows®-г олж авах

Image Writer for Windows® нь санах ойн диск рүү дүрс зөв бичдэг чөлөөтэй програм хангамж юм. Үүнийг https://launchpad.net/win32-image-writer/ хаягаас татаж аваад сан руу задална.

b. Image Writer ашиглан дүрсийг бичих

Програмыг эхлүүлэхийн тулд Win32DiskImager дээр хоёр дарна. `Device` доор харагдах хөтчийн үсэг санах ойн диск байгаа хөтөч эсэхийг шалгаарай. Сангийн дүрсэн дээр дараад санах ойн диск рүү бичигдэх дүрсийг сонгоно. Дүрс файлын нэрийг **[ Save ]** дарж сонгоно. Бүх юм зөв эсэхийг болон санах ойн дискний сангуудаас өөр цонхон дээр онгойгоогүй эсэхийг шалгаарай. Бүх юм бэлэн болсны дараа санах ойн диск рүү дүрсийг бичихийн тулд **[ Write ]**-г дарна.

Уян дискнээс суулгалт хийх нь дэмжигдэхээ больсон.

Та одоо FreeBSD суулгаж эхлэхэд бэлэн боллоо.

# 3.4. Суулгалтыг эхлүүлэх нь

Анхдагчаар дараах зурвас гарч иртэл суулгалт таны диск дээр ямар ч өөрчлөлт хийхгүй:

```
Your changes will now be written to disk.  If you
have chosen to overwrite existing data, it will
be PERMANENTLY ERASED. Are you sure you want to
commit your changes?
```

Энэ анхааруулга гарахаас өмнө хатуу дискнийхээ өгөгдлийг өөрчлөлгүйгээр суулгалтыг ямар ч үед зогсоож болно. Хэрэв та ямар нэг зүйлийг буруу тохируулсан байх гэж эмээж байгаа бол энэ үед хүрэхээсээ өмнө компьютераа зүгээр л унтраачихаж болох бөгөөд ямар ч хохирол гарахгүй юм.

## 3.4.1. Ачаалах нь

### 3.4.1.1. i386™ ба amd64 дээр ачаалах нь

1. Хэрэв та Суулгалтын зөөвөрлөгчийг бэлдэх-д тайлбарласан шиг "bootable" буюу ачаалагдах USB диск бэлдсэн бол компьютераа асаахаасаа өмнө USB дискээ залгаарай.

   Хэрэв та CDROM-с ачаалж байгаа бол компьютераа асаагаад CDROM-оо эхний боломж гарсан даруй хийгээрэй.

2. Суулгалтад ашиглах зөөвөрлөгчөөсөө хамаараад машинаа CDROM эсвэл USB-ээс ачаалахаар тохируулаарай. BIOS тохиргоо нь ачаалах төхөөрөмжийг сонгох боломжийг олгодог. Ихэвчлэн `F10`, `F11`, `F12`, юм уу эсвэл `Escape` дарж эхлэх үедээ ачаалах төхөөрөмжийг сонгох боломжийг ихэнх системүүд олгодог.
3. Хэрэв таны компьютер асуудалгүй эхэлж одоо байгаа үйлдлийн систем ачаалбал:
   a. Ачаалах процессын үед дискийг эрт хийгээгүй байж болох юм. Тэнд нь үлдээгээд компьютераа дахин ачаалаад үзээрэй.
   b. Өмнө хийсэн BIOS-н өөрчлөлт зөв ажиллаагүй байж болно. Та зөв сонголттой болтлоо тэр алхмыг дахин хийх хэрэгтэй.
   c. Таны BIOS хүссэн зөөвөрлөгчөөс ачаалахыг дэмждэггүй байж болно. Plop Boot Manager ашиглаж хуучин компьютерийг CD эсвэл USB зөөвөрлөгчөөс ачаалж болно.
4. FreeBSD ачаалж эхэлнэ. Хэрэв та CDROM-с ачаалж байгаа бол та доорхтой адилыг

харах болно (хувилбарын мэдээллийг оруулаагүй болно):

```
Booting from CD-ROM...
645MB medium detected
CD Loader 1.2

Building the boot loader arguments
Looking up /BOOT/LOADER... Found
Relocating the loader and the BTX
Starting the BTX loader

BTX loader 1.00 BTX version is 1.02
Consoles: internal video/keyboard
BIOS CD is cd0
BIOS drive C: is disk0
BIOS drive D: is disk1
BIOS 636kB/261056kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1

Loading /boot/defaults/loader.conf
/boot/kernel/kernel text=0x64daa0 data=0xa4e80+0xa9e40 syms
=[0x4+0x6cac0+0x4+0x88e9d]
\
```

5. FreeBSD ачаалагч дуудагч гарч ирнэ:

*Зураг 58. FreeBSD ачаалагч дуудагч цэс*

Either wait ten seconds, or press `Enter`.

#### 3.4.1.2. Macintosh® PowerPC® дээр ачаалах

Ихэнх машин дээр ачаалах үед `C` дарж CD-с ачаална. Үгүй бол `Command` + `Option` + `O` + `F`, юм уу эсвэл `Windows` + `Alt` + `O` + `F` товчлууруудыг Apple® биш гар дээр дарж хийнэ. `0 >` гарч ирэхэд доорхийг бичнэ.

```
boot cd:,\ppc\loader cd:0
```

Гар байхгүй Xserves-ийн хувьд Open Firmware рүү ачаалах талаар Apple®-н сайтаас хараарай.

#### 3.4.1.3. sparc64-н хувьд ачаалах нь

Ихэнх sparc64 системүүд дискнээс автоматаар ачаалахаар тохируулагдсан байдаг. FreeBSD суулгахын тулд сүлжээнээс юм уу эсвэл CDROM-с ачаалах хэрэгтэй бөгөөд та PROM (OpenFirmware) руу орох шаардлагатай.

Ингэхийн тулд системээ дахин эхлүүлээд ачаалах зурвас гарч иртэл хүлээнэ. Загвараас хамаарах бөгөөд доорхтой төстэй байна:

```
Sun Blade 100 (UltraSPARC-IIe), Keyboard Present
Copyright 1998-2001 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.2, 128 MB memory installed, Serial #51090132.
Ethernet address 0:3:ba:b:92:d4, Host ID: 830b92d4.
```

Энэ үед таны систем дискнээс ачаалахаар завдах бол PROM руу орохын тулд та `L1` + `A` эсвэл `Stop` + `A` дарах юм уу эсвэл сериал консолоор `BREAK` илгээх хэрэгтэй. Иймэрхүү харагдах болно:

```
ok     ①
ok {0} ②
```

① This is the prompt used on systems with just one CPU.

② This is the prompt used on SMP systems, the digit indicates the number of the active CPU.

Энэ үед хөтөч дотроо CDROM-оо хийгээд PROM дээрээс `boot cdrom` гэж бичнэ.

### 3.4.2. Reviewing the Device Probe Results

Дэлгэц дээр гасан мөрүүдийн сүүлийн хэдэн зуун мөр хадгалагддаг бөгөөд дахин харах боломжтой байдаг.

Буфферийг дахин үзэхийн тулд `Scroll Lock` дарна. Энэ нь дэлгэц дээр буцаж гүйлгэх боломжтой болгодог. Та дараа нь сум дарах юм уу эсвэл `PageUp` ба `PageDown` дарж үр дүнг харж болно. `Scroll Lock` дахин дарж гүйлгэх боломжгүй болгоно.

Цөм төхөөрөмжийг шалгаж байх үеийн текст хүртэл буцаан гүйлгэж одоо үзээрэй. Та Ердийн төхөөрөмж шалгасан үр дүн дээрх шиг адил төстэй текстийг харах бөгөөд таны

компьютер дээр байгаа төхөөрөмжүүдээс хамаарч текст өөр байж болно.

*Ердийн төхөөрөмж шалгасан үр дүн*

```
Copyright (c) 1992-2011 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.
FreeBSD 9.0-RELEASE #0 r225473M: Sun Sep 11 16:07:30 BST 2011
    root@psi:/usr/obj/usr/src/sys/GENERIC amd64
CPU: Intel(R) Core(TM)2 Duo CPU     T9400  @ 2.53GHz (2527.05-MHz K8-class CPU)
  Origin = "GenuineIntel"  Id = 0x10676  Family = 6  Model = 17  Stepping = 6
  Features
=0xbfebfbff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,APIC,SEP,MTRR,PGE,MCA,CMOV,PAT,PSE36,CL
FLUSH,DTS,ACPI,MMX,FXSR,SSE,SSE2,SS,HTT,TM,PBE>
  Features2
=0x8e3fd<SSE3,DTES64,MON,DS_CPL,VMX,SMX,EST,TM2,SSSE3,CX16,xTPR,PDCM,SSE4.1>
  AMD Features=0x20100800<SYSCALL,NX,LM>
  AMD Features2=0x1<LAHF>
  TSC: P-state invariant, performance statistics
real memory  = 3221225472 (3072 MB)
avail memory = 2926649344 (2791 MB)
Event timer "LAPIC" quality 400
ACPI APIC Table: <TOSHIB A0064   >
FreeBSD/SMP: Multiprocessor System Detected: 2 CPUs
FreeBSD/SMP: 1 package(s) x 2 core(s)
 cpu0 (BSP): APIC ID:  0
 cpu1 (AP): APIC ID:  1
ioapic0: Changing APIC ID to 1
ioapic0 <Version 2.0> irqs 0-23 on motherboard
kbd1 at kbdmux0
acpi0: <TOSHIB A0064> on motherboard
acpi0: Power Button (fixed)
acpi0: reservation of 0, a0000 (3) failed
acpi0: reservation of 100000, b6690000 (3) failed
Timecounter "ACPI-safe" frequency 3579545 Hz quality 850
acpi_timer0: <24-bit timer at 3.579545MHz> port 0xd808-0xd80b on acpi0
cpu0: <ACPI CPU> on acpi0
ACPI Warning: Incorrect checksum in table [ASF!] - 0xFE, should be 0x9A
(20110527/tbutils-282)
cpu1: <ACPI CPU> on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci0: <ACPI PCI bus> on pcib0
vgapci0: <VGA-compatible display> port 0xcff8-0xcfff mem 0xff400000-
0xff7fffff,0xe0000000-0xefffffff irq 16 at device 2.0 on pci0
agp0: <Intel GM45 SVGA controller> on vgapci0
agp0: aperture size is 256M, detected 131068k stolen memory
vgapci1: <VGA-compatible display> mem 0xffc00000-0xffcfffff at device 2.1 on pci0
pci0: <simple comms> at device 3.0 (no driver attached)
em0: <Intel(R) PRO/1000 Network Connection 7.2.3> port 0xcf80-0xcf9f mem 0xff9c0000-
0xff9dffff,0xff9fe000-0xff9fefff irq 20 at device 25.0 on pci0
```

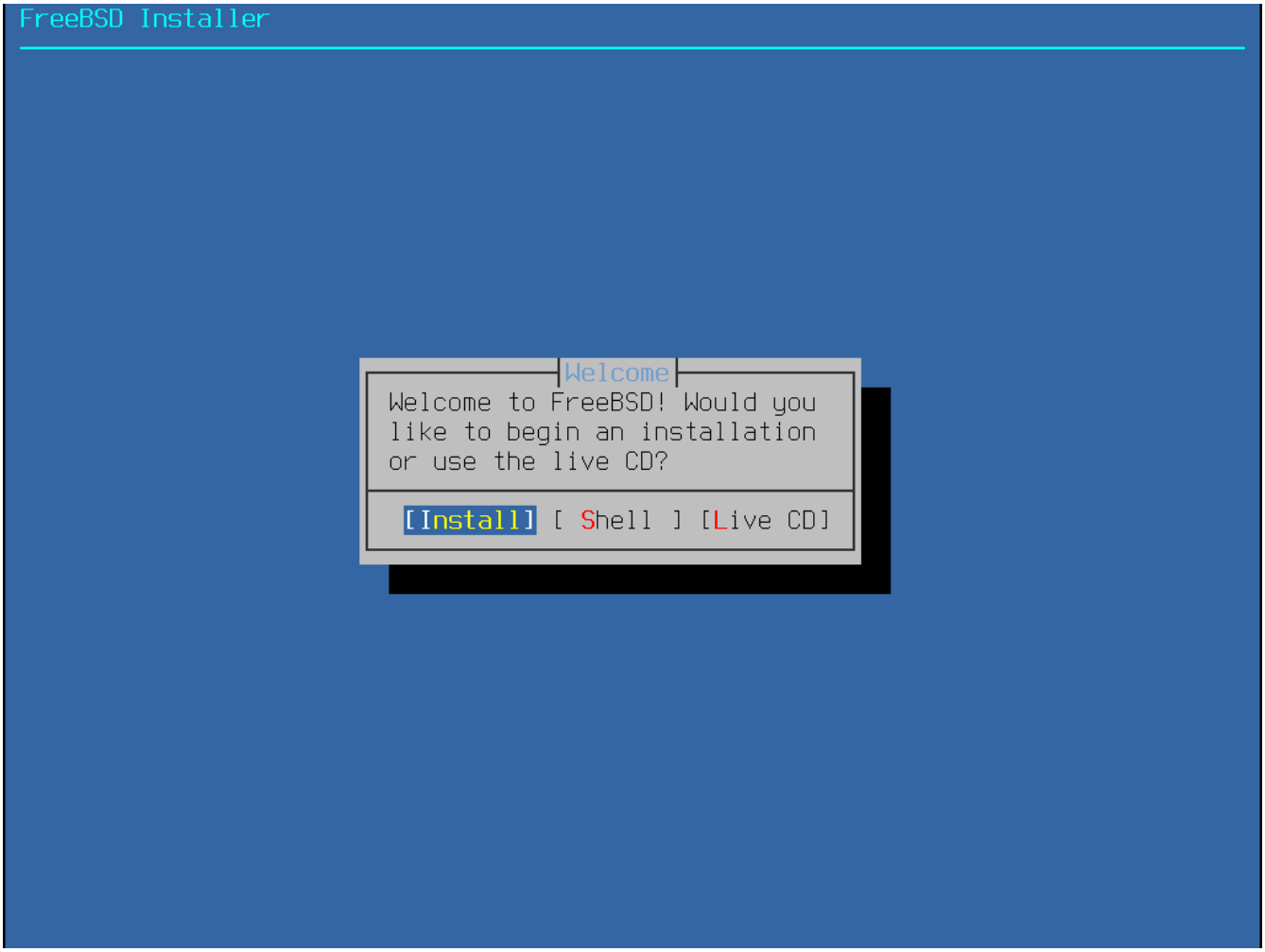
```
em0: Using an MSI interrupt
em0: Ethernet address: 00:1c:7e:6a:ca:b0
uhci0: <Intel 82801I (ICH9) USB controller> port 0xcf60-0xcf7f irq 16 at device 26.0
on pci0
usbus0: <Intel 82801I (ICH9) USB controller> on uhci0
uhci1: <Intel 82801I (ICH9) USB controller> port 0xcf40-0xcf5f irq 21 at device 26.1
on pci0
usbus1: <Intel 82801I (ICH9) USB controller> on uhci1
uhci2: <Intel 82801I (ICH9) USB controller> port 0xcf20-0xcf3f irq 19 at device 26.2
on pci0
usbus2: <Intel 82801I (ICH9) USB controller> on uhci2
ehci0: <Intel 82801I (ICH9) USB 2.0 controller> mem 0xff9ff800-0xff9ffbff irq 19 at
device 26.7 on pci0
usbus3: EHCI version 1.0
usbus3: <Intel 82801I (ICH9) USB 2.0 controller> on ehci0
hdac0: <Intel 82801I High Definition Audio Controller> mem 0xff9f8000-0xff9fbfff irq
22 at device 27.0 on pci0
pcib1: <ACPI PCI-PCI bridge> irq 17 at device 28.0 on pci0
pci1: <ACPI PCI bus> on pcib1
iwn0: <Intel(R) WiFi Link 5100> mem 0xff8fe000-0xff8fffff irq 16 at device 0.0 on pci1
pcib2: <ACPI PCI-PCI bridge> irq 16 at device 28.1 on pci0
pci2: <ACPI PCI bus> on pcib2
pcib3: <ACPI PCI-PCI bridge> irq 18 at device 28.2 on pci0
pci4: <ACPI PCI bus> on pcib3
pcib4: <ACPI PCI-PCI bridge> at device 30.0 on pci0
pci5: <ACPI PCI bus> on pcib4
cbb0: <RF5C476 PCI-CardBus Bridge> at device 11.0 on pci5
cardbus0: <CardBus bus> on cbb0
pccard0: <16-bit PCCard bus> on cbb0
isab0: <PCI-ISA bridge> at device 31.0 on pci0
isa0: <ISA bus> on isab0
ahci0: <Intel ICH9M AHCI SATA controller> port 0x8f58-0x8f5f,0x8f54-0x8f57,0x8f48-
0x8f4f,0x8f44-0x8f47,0x8f20-0x8f3f mem 0xff9fd800-0xff9fdfff irq 19 at device 31.2 on
pci0
ahci0: AHCI v1.20 with 4 3Gbps ports, Port Multiplier not supported
ahcich0: <AHCI channel> at channel 0 on ahci0
ahcich1: <AHCI channel> at channel 1 on ahci0
ahcich2: <AHCI channel> at channel 4 on ahci0
acpi_lid0: <Control Method Lid Switch> on acpi0
battery0: <ACPI Control Method Battery> on acpi0
acpi_button0: <Power Button> on acpi0
acpi_acad0: <AC Adapter> on acpi0
acpi_toshiba0: <Toshiba HCI Extras> on acpi0
acpi_tz0: <Thermal Zone> on acpi0
attimer0: <AT timer> port 0x40-0x43 irq 0 on acpi0
Timecounter "i8254" frequency 1193182 Hz quality 0
Event timer "i8254" frequency 1193182 Hz quality 100
atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0
atkbd0: <AT Keyboard> irq 1 on atkbdc0
kbd0 at atkbd0
atkbd0: [GIANT-LOCKED]
```

```
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: model GlidePoint, device ID 0
atrtc0: <AT realtime clock> port 0x70-0x71 irq 8 on acpi0
Event timer "RTC" frequency 32768 Hz quality 0
hpet0: <High Precision Event Timer> iomem 0xfed00000-0xfed003ff on acpi0
Timecounter "HPET" frequency 14318180 Hz quality 950
Event timer "HPET" frequency 14318180 Hz quality 450
Event timer "HPET1" frequency 14318180 Hz quality 440
Event timer "HPET2" frequency 14318180 Hz quality 440
Event timer "HPET3" frequency 14318180 Hz quality 440
uart0: <16550 or compatible> port 0x3f8-0x3ff irq 4 flags 0x10 on acpi0
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
ppc0: cannot reserve I/O port range
est0: <Enhanced SpeedStep Frequency Control> on cpu0
p4tcc0: <CPU Frequency Thermal Control> on cpu0
est1: <Enhanced SpeedStep Frequency Control> on cpu1
p4tcc1: <CPU Frequency Thermal Control> on cpu1
Timecounters tick every 1.000 msec
hdac0: HDA Codec #0: Realtek ALC268
hdac0: HDA Codec #1: Lucent/Agere Systems (Unknown)
pcm0: <HDA Realtek ALC268 PCM #0 Analog> at cad 0 nid 1 on hdac0
pcm1: <HDA Realtek ALC268 PCM #1 Analog> at cad 0 nid 1 on hdac0
usbus0: 12Mbps Full Speed USB v1.0
usbus1: 12Mbps Full Speed USB v1.0
usbus2: 12Mbps Full Speed USB v1.0
usbus3: 480Mbps High Speed USB v2.0
ugen0.1: <Intel> at usbus0
uhub0: <Intel UHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0
ugen1.1: <Intel> at usbus1
uhub1: <Intel UHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus1
ugen2.1: <Intel> at usbus2
uhub2: <Intel UHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus2
ugen3.1: <Intel> at usbus3
uhub3: <Intel EHCI root HUB, class 9/0, rev 2.00/1.00, addr 1> on usbus3
uhub0: 2 ports with 2 removable, self powered
uhub1: 2 ports with 2 removable, self powered
uhub2: 2 ports with 2 removable, self powered
uhub3: 6 ports with 6 removable, self powered
ugen2.2: <vendor 0x0b97> at usbus2
uhub8: <vendor 0x0b97 product 0x7761, class 9/0, rev 1.10/1.10, addr 2> on usbus2
ugen1.2: <Microsoft> at usbus1
ada0 at ahcich0 bus 0 scbus1 target 0 lun 0
ada0: <Hitachi HTS543225L9SA00 FBEOC43C> ATA-8 SATA 1.x device
ada0: 150.000MB/s transfers (SATA 1.x, UDMA6, PIO 8192bytes)
ada0: Command Queueing enabled
ada0: 238475MB (488397168 512 byte sectors: 16H 63S/T 16383C)
ada0: Previously was known as ad4
ums0: <Microsoft Microsoft 3-Button Mouse with IntelliEyeTM, class 0/0, rev 1.10/3.00,
```

```
addr 2> on usbus1
SMP: AP CPU #1 Launched!
cd0 at ahcich1 bus 0 scbus2 target 0 lun 0
cd0: <TEAC DV-W28S-RT 7.0C> Removable CD-ROM SCSI-0 device
cd0: 150.000MB/s transfers (SATA 1.x, ums0: 3 buttons and [XYZ] coordinates ID=0
UDMA2, ATAPI 12bytes, PIO 8192bytes)
cd0: cd present [1 x 2048 byte records]
ugen0.2: <Microsoft> at usbus0
ukbd0: <Microsoft Natural Ergonomic Keyboard 4000, class 0/0, rev 2.00/1.73, addr 2>
on usbus0
kbd2 at ukbd0
uhid0: <Microsoft Natural Ergonomic Keyboard 4000, class 0/0, rev 2.00/1.73, addr 2>
on usbus0
Trying to mount root from cd9660:/dev/iso9660/FREEBSD_INSTALL [ro]...
```

FreeBSD таны хүсэн хүлээж байсан бүх төхөөрөмжийг олсон эсэхийг төхөөрөмж шалгасан үр дүнгээс сайн шалгаарай. Хэрэв төхөөрөмж олдоогүй бол харагдахгүй байх болно. Цөмийн модулиуд нь GENERIC цөмд байхгүй төхөөрөмжүүдийн дэмжлэгийг хийх боломжийг олгодог.

Төхөөрөмж шалгах алхмын дараа та Суулгалтын зөөвөрлөгч сонгох нь-г харах болно. Суулгалтын зөөвөрлөгчийг гурван янзын аргаар ашиглаж болно: FreeBSD суулгах, амьд CD маягаар, эсвэл FreeBSD бүрхүүл ашиглаж болно. Сум дарж сонгох арга дээрээ очоод `Enter` дарна.




*Зураг 59. Суулгалтын зөөвөрлөгч сонгох нь*

**[ Install ]**-г сонгосноор суулгагч руу орно.

# 3.5. bsdinstall-г танилцуулах нь

bsdinstall нь Nathan Whitehorn <nwhitehorn@FreeBSD.org>-ий бичсэн текст дээр тулгуурласан FreeBSD-н суулгалтын програм бөгөөд 2011 онд FreeBSD 9.0-д зориулж танилцуулагдсан.

Kris Moore <kmoore@FreeBSD.org>-н pc-sysinstall нь PC-BSD-д байдаг бөгөөд бас FreeBSD суулгахад ашиглаж болно. bsdinstall-той хольж сольж ойлгох тохиолдол байдаг ч хоорондоо хамааралгүй юм.

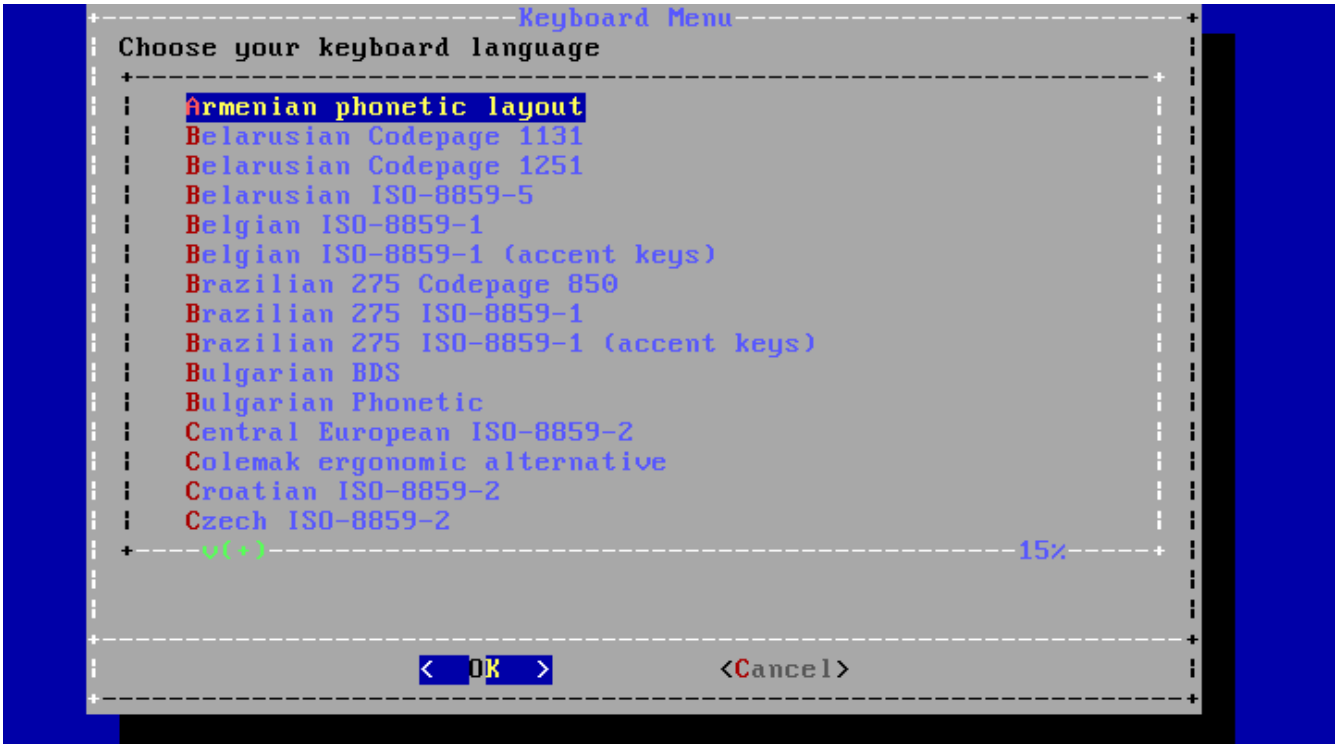
bsdinstall-н цэсний системийг сум, `Enter`, `Tab`, `Space` болон бусад товчлууруудын тусламжтай хянаж болно.

## 3.5.1. Keymap цэс сонгох нь

Системийн ямар консол ашиглаж байгаагаас хамаарч bsdinstall нь анхдагч биш гарын байрлал сонгох дэлгэц харуулж болно.

*Зураг 60. Гарын байрлалын сонголт*

Хэрэв **[ YES ]**-г сонгосон бол дараах гар сонгох дэлгэц гарч ирнэ. Үгүй бол энэ сонголтын дэлгэц гарч ирэхгүй бөгөөд анхдагч гарын байрлал хэрэглэгдэнэ.



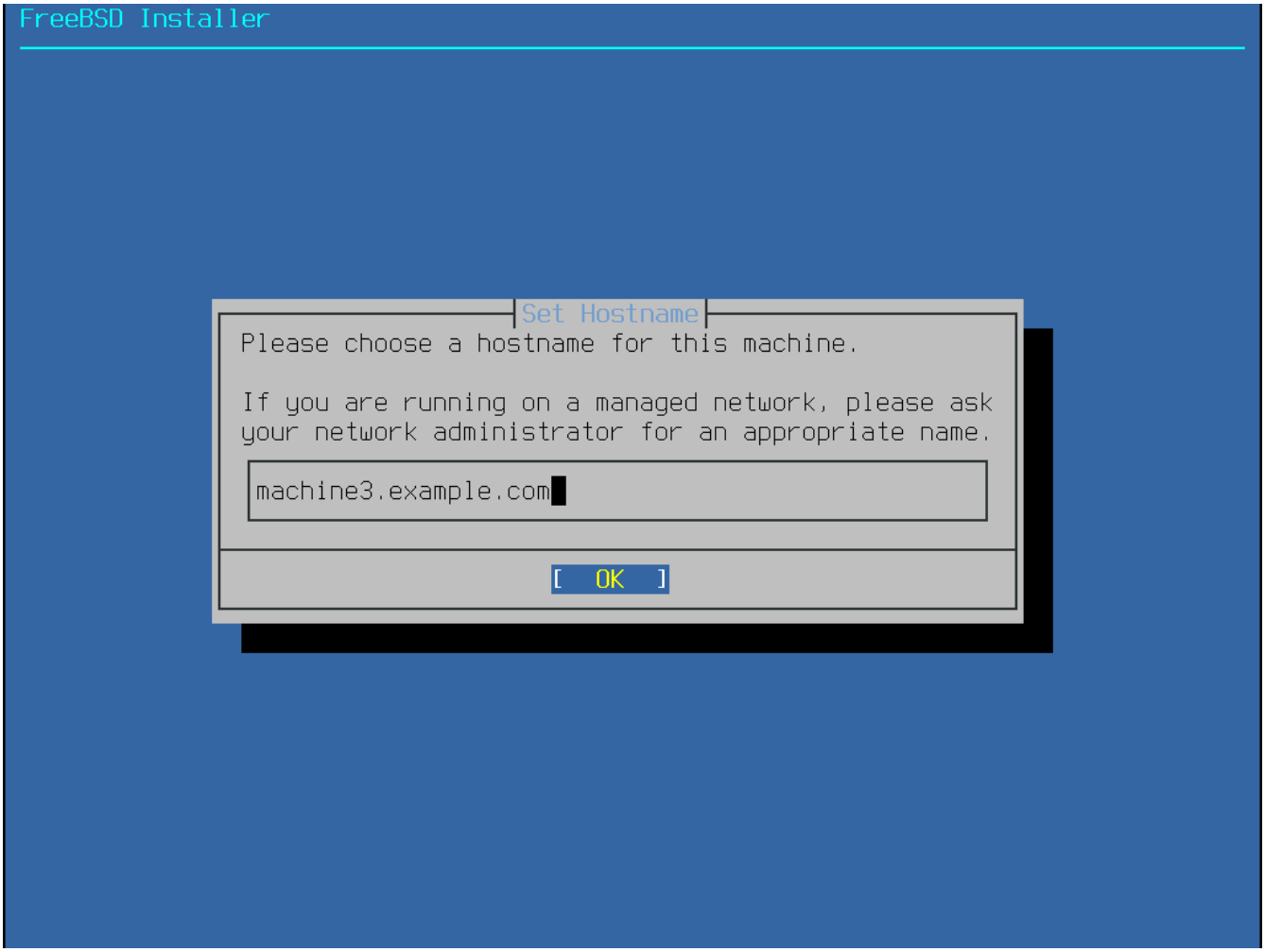

*Зураг 61. Гарын цэсийг сонгох*

Системд холбоотой байгаа гартай хамгийн төстэй гарын байрлалыг дээш доош гүйлгэх сумыг ашиглан `Enter` дарж сонгоно.

`Esc` дарснаар анхдагч гарын байрлалыг сонгоно. United States of America ISO-8859-1 нь гарын байрлал тодорхой биш бол бас аюулгүй сонголт юм.

### 3.5.2. Хостын нэрийг тохируулах нь

Дараа нь bsdinstall шинэ суух системд өгөх хостын нэрийг асууна.

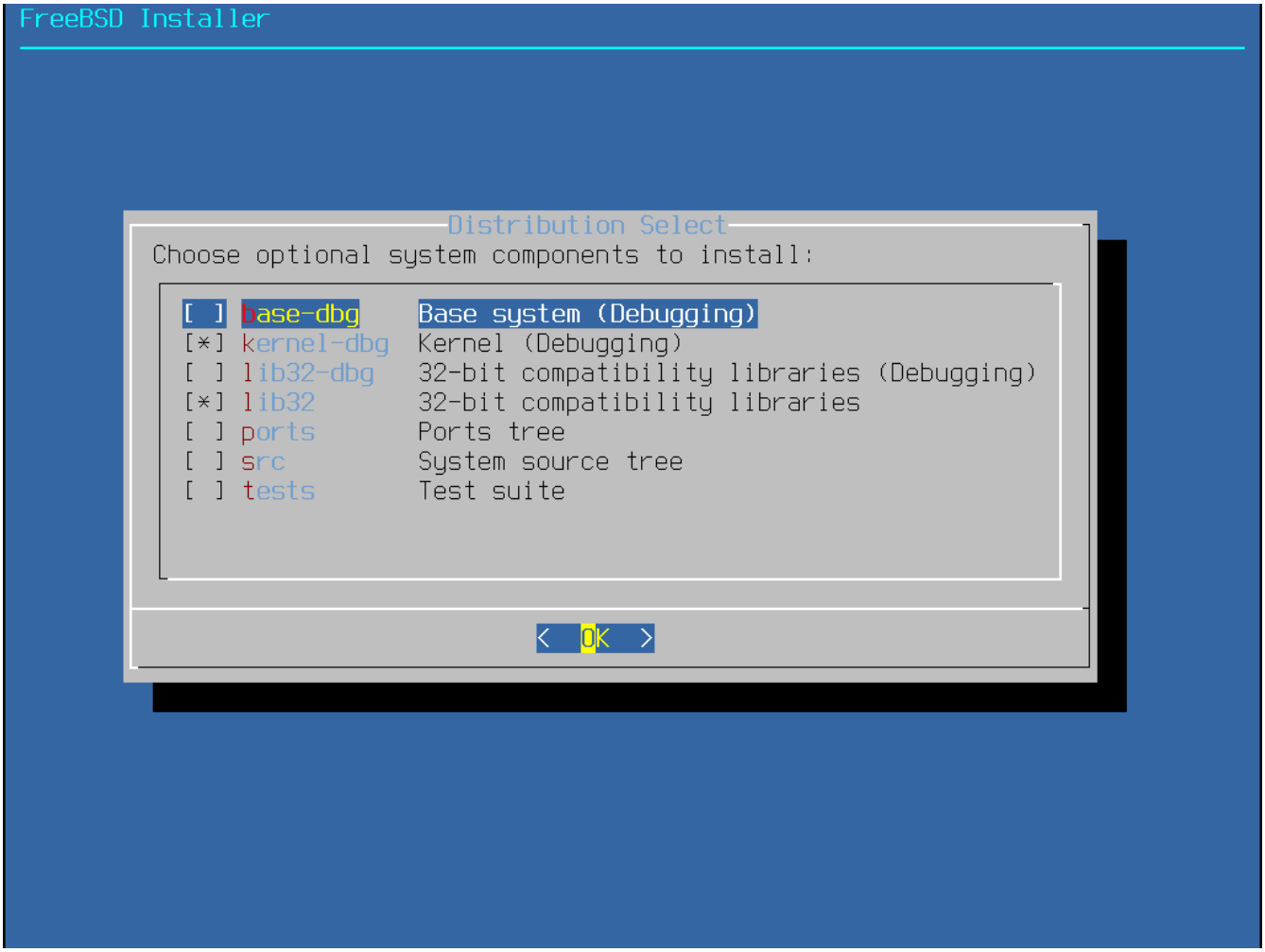



*Зураг 62. Хостын нэрийг тохируулах*

Оруулсан хостын нэр нь `machine3.example.com` гэх мэт бүрэн хостын нэр байх ёстой.

### 3.5.3. Бүрэлдэхүүн хэсгүүдийг сонгож суулгах нь

Дараа нь bsdinstall бүрэлдэхүүн хэсгүүдийг суулгах сонголтыг харуулж асууна.




*Зураг 63. Бүрэлдэхүүн хэсгүүдийг суулгахаар сонгох*

Алийг суулгахаа шийдэх нь системийг ямар зорилгоор ашиглах болон дискний хэмжээнээс ихээхэн хамаарна. FreeBSD цөм болон хэрэглэгчийн талбар (хамтдаа үндсэн систем буюу "base system" гэгддэг) үргэлж сууна.

Суулгах төрлөөс хамаарч зарим нэг нь харагдахгүй байж болно.

*Сонгох боломжтой бүрэлдэхүүнүүд*

- `doc` - Голчлон түүхийн сонирхлоос улбаатай нэмэлт баримтууд. FreeBSD-н баримтжуулах төслийн баримт бичгүүдийг дараа нь суулгаж болно.
- `games` - fortune, rot13 болон бусад BSD-ийн хэд хэдэн уламжлалт тоглоом.
- `lib32` - 64 битийн FreeBSD дээр 32 битийн програм ажиллуулахад зориулсан нийцтэй сангууд.
- `ports` - FreeBSD-н портын цуглуулга.

  Портын цуглуулга нь програм суулгах хялбар бөгөөд тохиромжтой арга юм. Портын цуглуулга нь програмыг бүтээхэд шаардлагатай эх кодыг агуулдаггүй. Харин гуравдагч талуудын програмуудыг татаж бүтээн суулгах процессыг автоматжуулахад хэрэгтэй файлуудын цуглуулга юм. Програм суулгах. Багцууд болон портууд хэсэгт портын цуглуулгыг хэрхэн ашиглах талаар хэлэлцдэг.

Суулгалтын програм шаардлагатай зай байгаа эсэхийг шалгадаггүй. Энэ

> сонголтыг зөвхөн шаардлагатай зайтай бол сонгоорой. FreeBSD 9.0 дээр FreeBSD-н портын цуглуулга ойролцоогоор 3 GB хэмжээтэй байдаг. Та илүү сүүлийн үеийн FreeBSD-н хувилбарууд дээр арай илүү хэмжээтэй байна гэж ойлгоход аюулгүй.

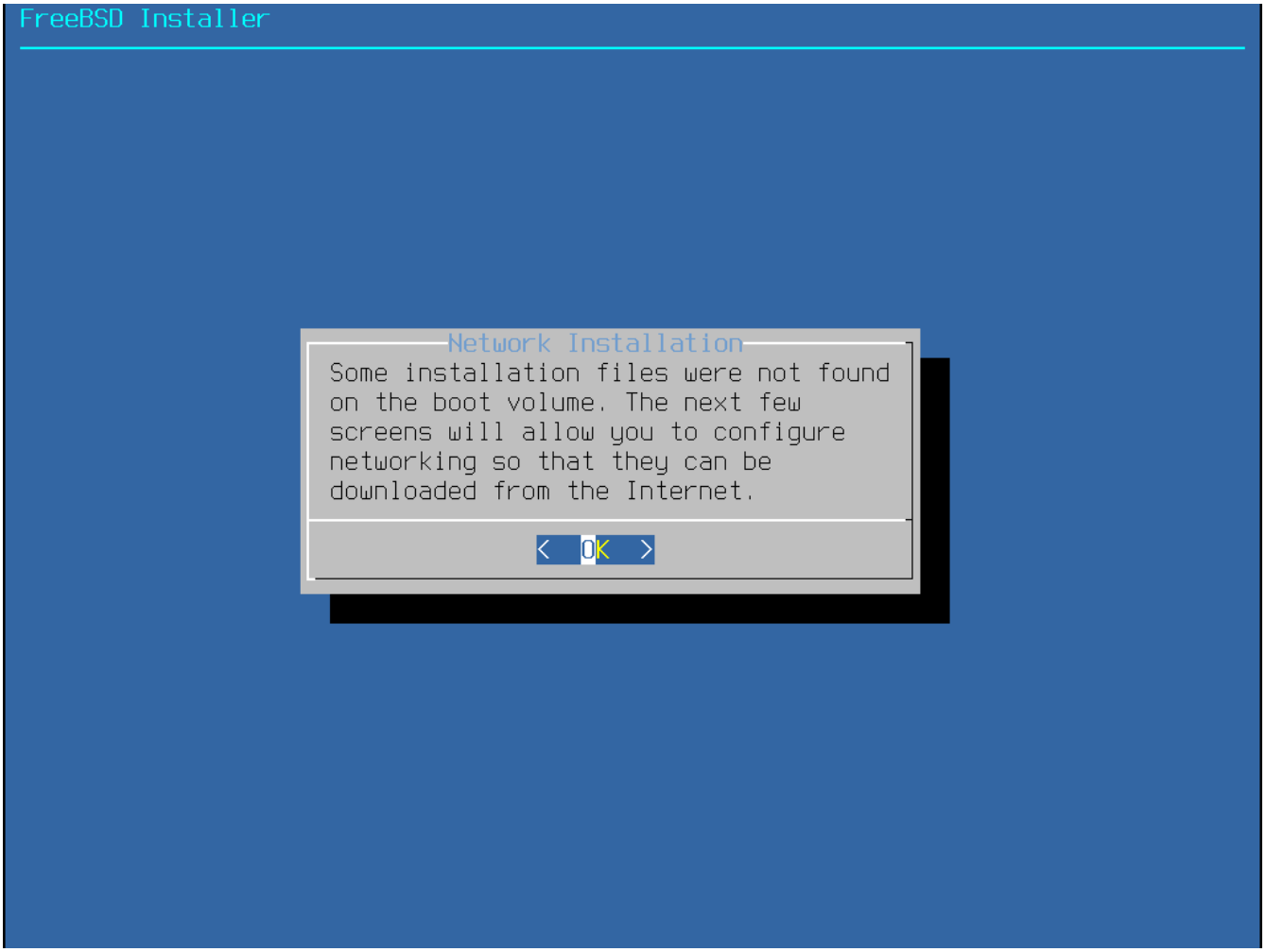
- `src` - Системийн эх код.

  FreeBSD нь цөм болон хэрэглэгчийн талбарын бүрэн хэмжээний эх кодтой ирдэг. Ихэнх програмын хувьд шаардлагагүй боловч эх хэлбэрээр ирсэн зарим нэг програм (жишээ нь төхөөрөмжийн драйверууд эсвэл цөмийн модулиуд) эсвэл FreeBSD-г хөгжүүлэхэд шаардлагатай байж болох юм.

  Бүрэн хэмжээний эх кодын мод 1 GB дискний хэмжээг шаардах бөгөөд FreeBSD системийг дахин бүтээхэд нэмэлт 5 GB зай шаарддаг.

## 3.6. Сүлжээнээс суулгах нь

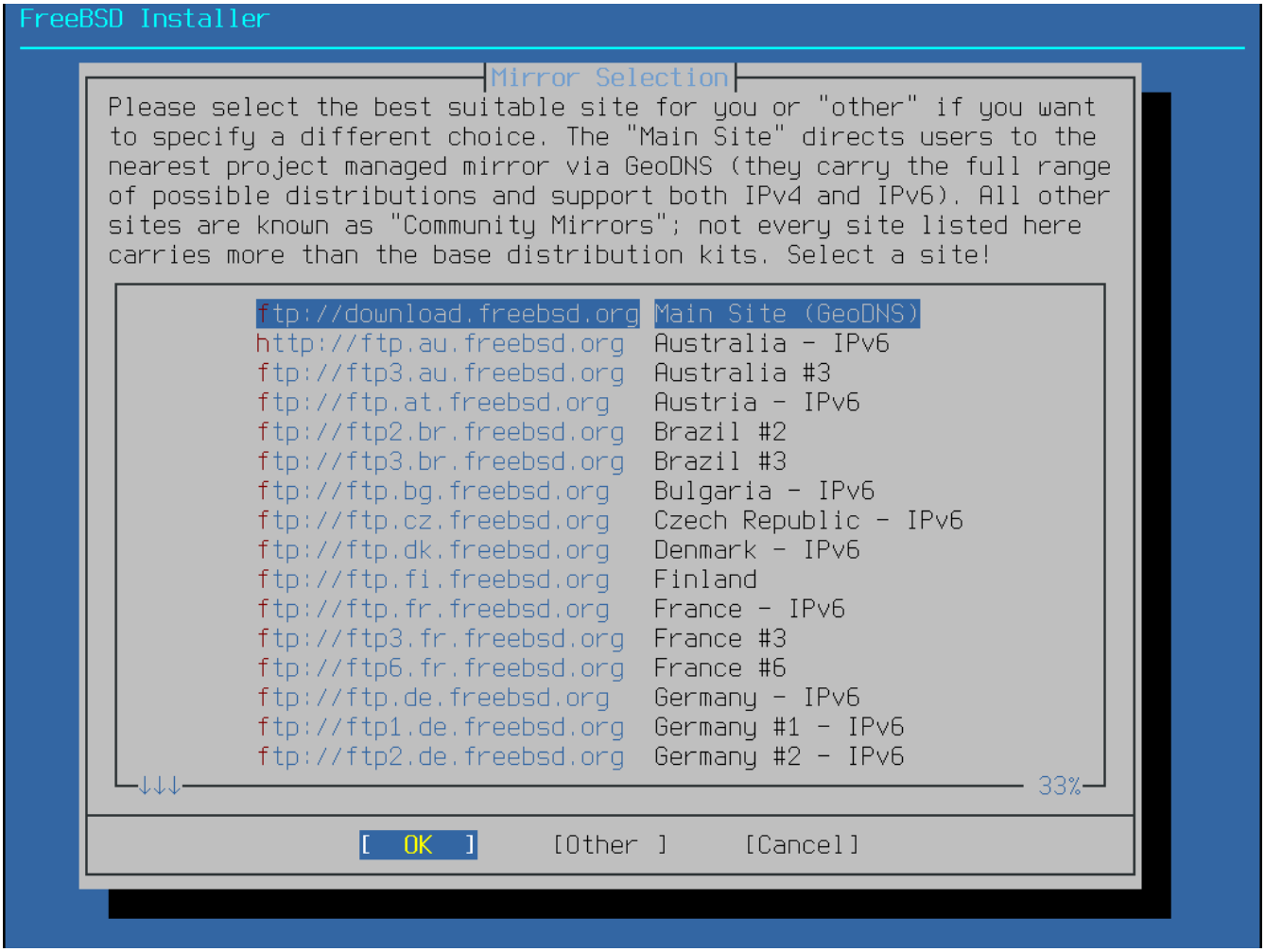
*bootonly* буюу зөвхөн ачаалах суулгалтын зөөвөрлөгч нь суулгалтын файлуудын хуулбарыг агуулдаггүй. *bootonly* суулгалтын арга ашиглах үед файлуудыг шаардлагатай үед нь сүлжээгээр татаж авах ёстой байдаг.




*Зураг 64. Сүлжээнээс суулгах нь*

Сүлжээний интерфэйсүүдийг тохируулах нь хэсэгт заагдсаны дагуу сүлжээний холболтын

тохиргоо хийгдсэний дараа толин тусгал сайт сонгогддог. Толин тусгал сайтууд нь FreeBSD-н файлуудын хуулбарыг хадгалж байдаг. FreeBSD суулгах компьютер байгаа тэр газартайгаа ойрхон бүсээс толин тусгалын сайтаа сонгох хэрэгтэй. Суулгах компьютертай ойролцоо толин тусгал сайтыг сонгох нь файлыг илүү хурдан татаж авах боломжийг олгох бөгөөд суулгалтын хугацаа богиносох болно.

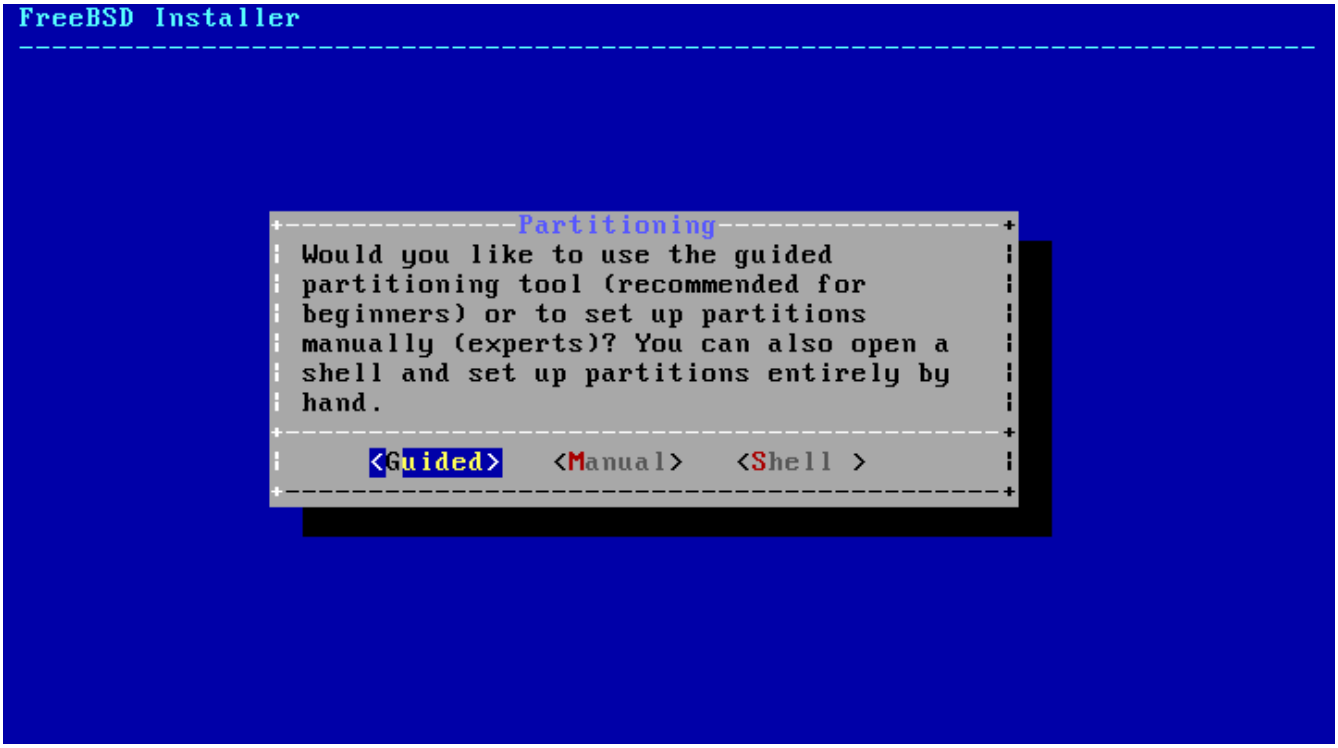



*Зураг 65. Толин тусгал сонгох нь*

Суулгалтын файлууд локал зөөвөрлөгч дээр байгаа юм шиг суулгалт үргэлжлэх болно.

## 3.7. Дискний зай авах нь

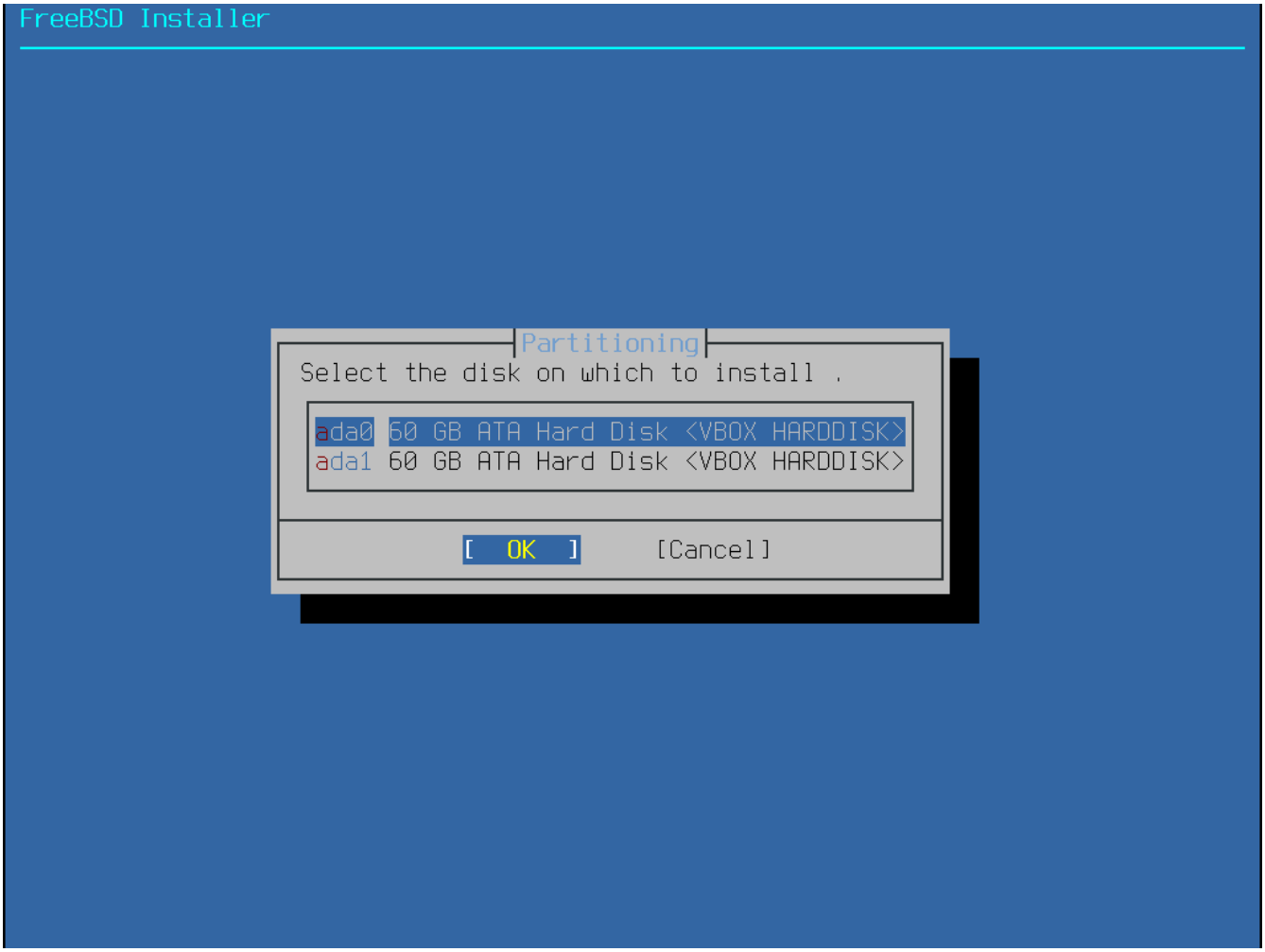
FreeBSD дээр гурван янзын аргаар дискний зай хуваарилж болно. *Guided* буюу зааварчилсан хуваалт нь дискийн хуваалтыг автоматаар хийдэг бол *Manual* хуваалт нь дэвшилтэт хэрэглэгчийн хувьд хуваалтад өөрчлөлт хийх боломжийг олгодог. Төгсгөлд нь gpart(8), fdisk(8), болон bsdlabel(8) гэх мэт тушаалын мөрийн програмуудыг шууд ашиглах боломжийг олгодог бүрхүүлийг эхлүүлэх сонголт бас бий.




*Зураг 66. Зааварчилсан эсвэл Гар аргаар хуваах горимыг сонгох*

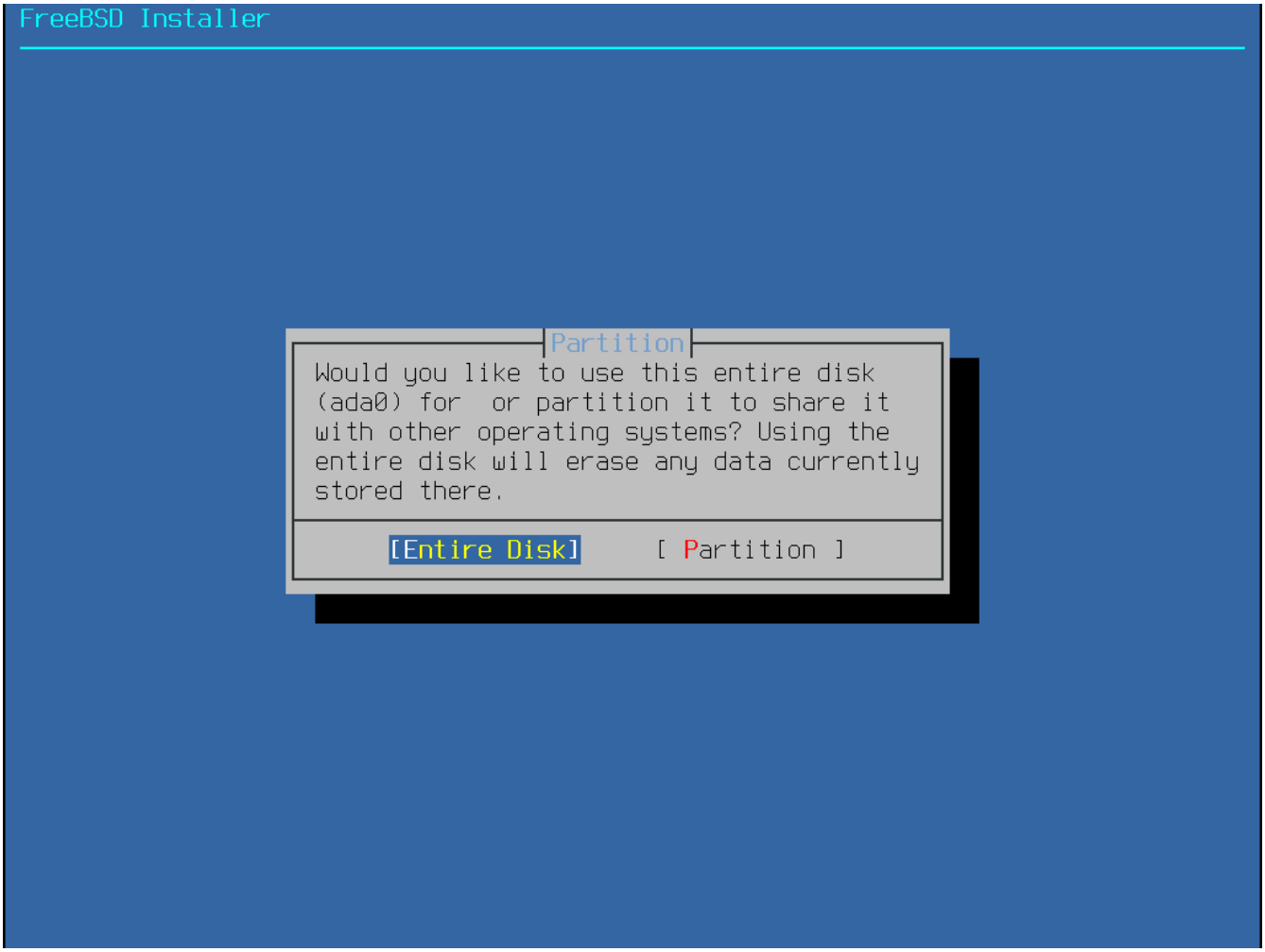
### 3.7.1. Зааварчилсан хуваалт

Олон диск холбоотой бол FreeBSD суулгах дискийг сонгох хэрэгтэй.



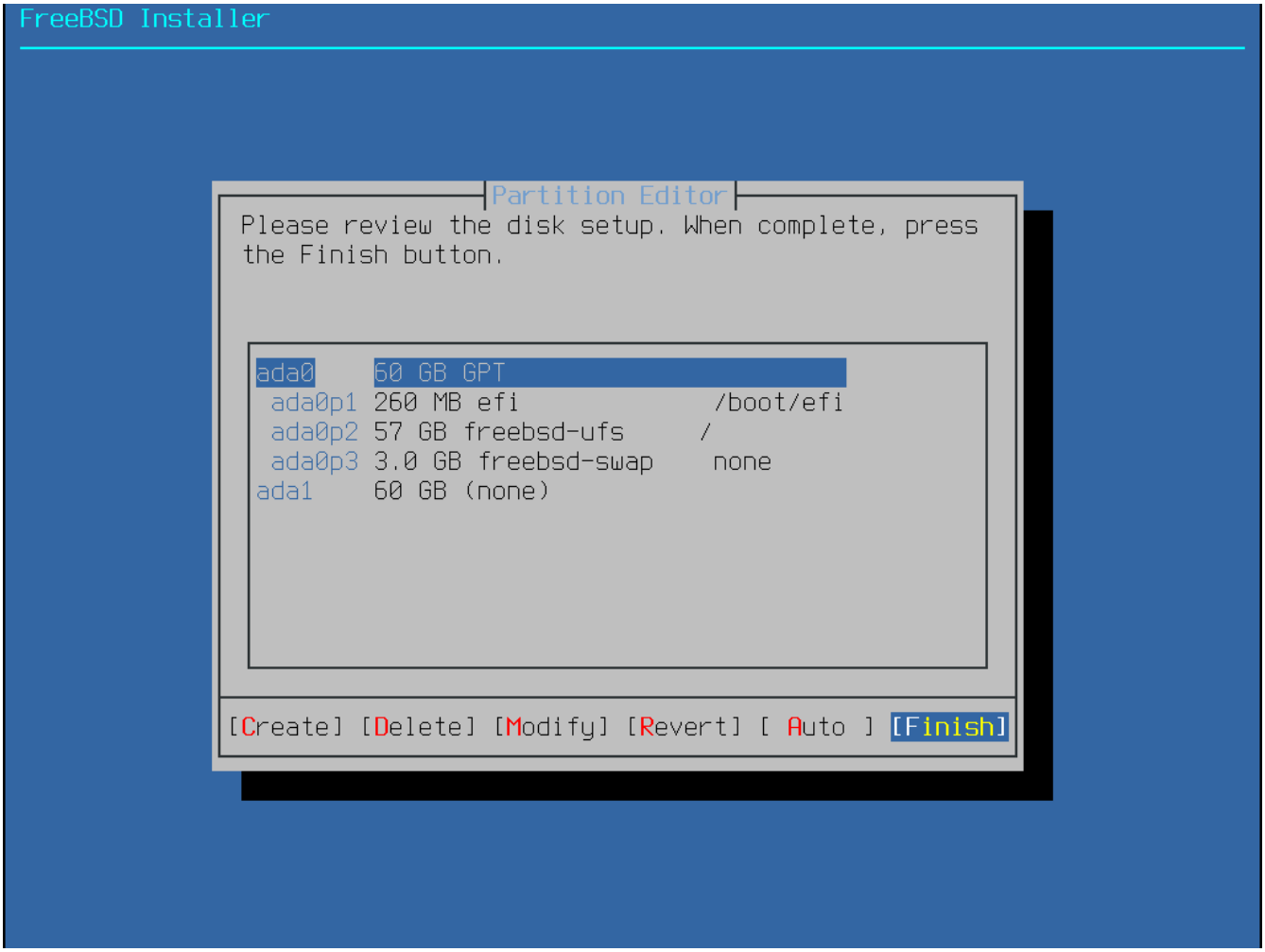

*Зураг 67. Олон дискээс сонгох*

Бүх дискийг FreeBSD-д зориулж хуваарилж болно, эсвэл зөвхөн хэсгийг хуваарилж болно. Хэрэв **[ Entire Disk ]** сонгосон бол бүх дискийг эзэлсэн ерөнхий хуваалтын тогтоц үүснэ. **[ Partition ]**-г сонгосноор дискний ашиглагдаагүй байгаа хэсэгт хуваалтын тогтцыг үүсгэнэ.




*Зураг 68. Бүх диск эсвэл хуваалтыг сонгох*

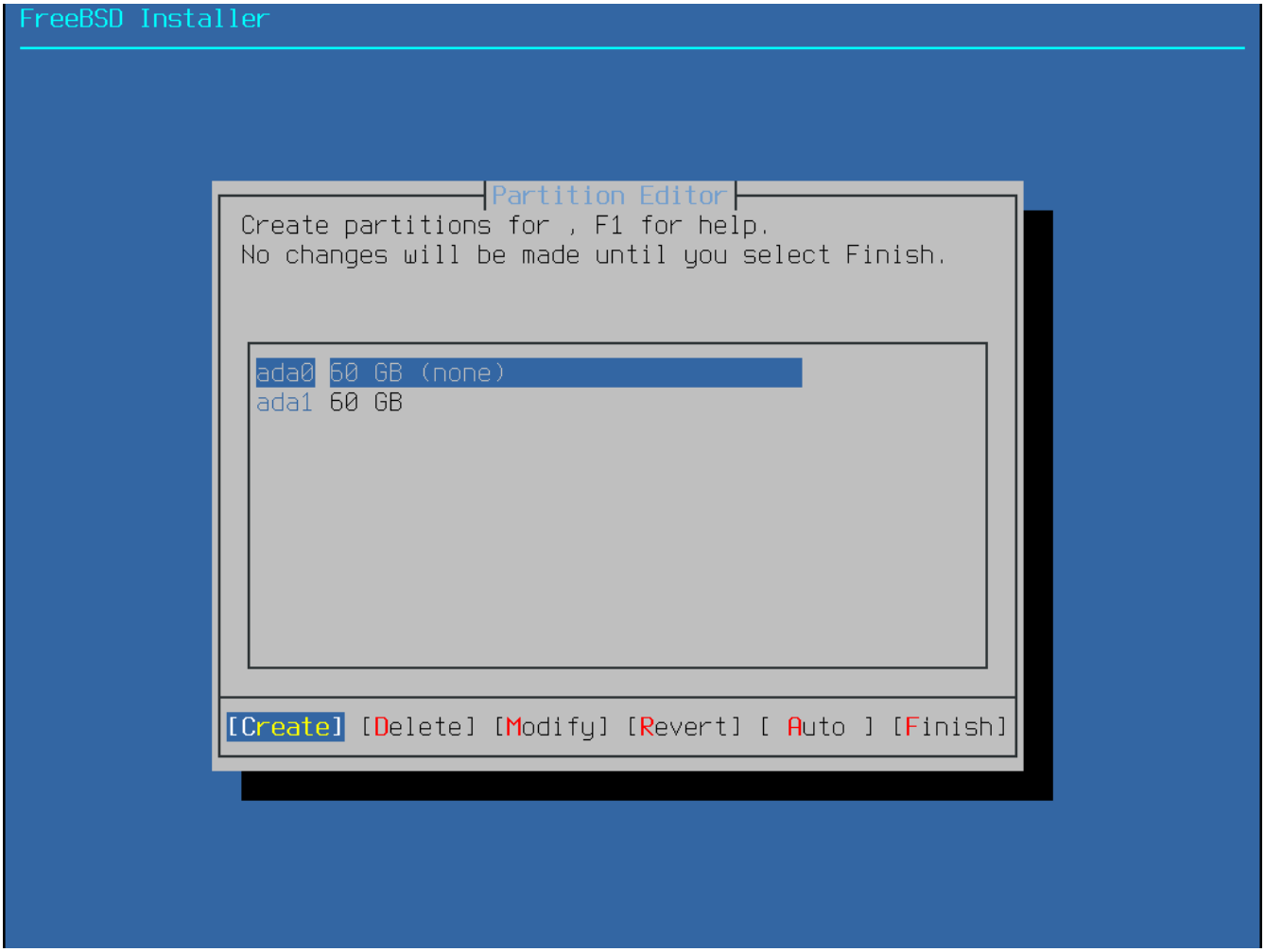
Хуваалтын тогтцыг үүсгэсний дараа зөв эсэхийг шалгаарай. Хэрэв алдаа хийсэн бол **[ Revert ]** гэдгийг сонгосноор хуваалтуудыг урьдын байрлалд аваачна, эсвэл **[ Auto ]** нь автомат FreeBSD-н хуваалтуудыг үүсгэх болно. Хуваалтуудыг гараар үүсгэх, өөрчлөх, эсвэл устгах боломжтой. Хуваалт зөв бол **[ Finish ]**-г сонгож суулгалтыг үргэлжлүүлнэ.




*Зураг 69. Үүсгэсэн хуваалтуудыг шалгах*

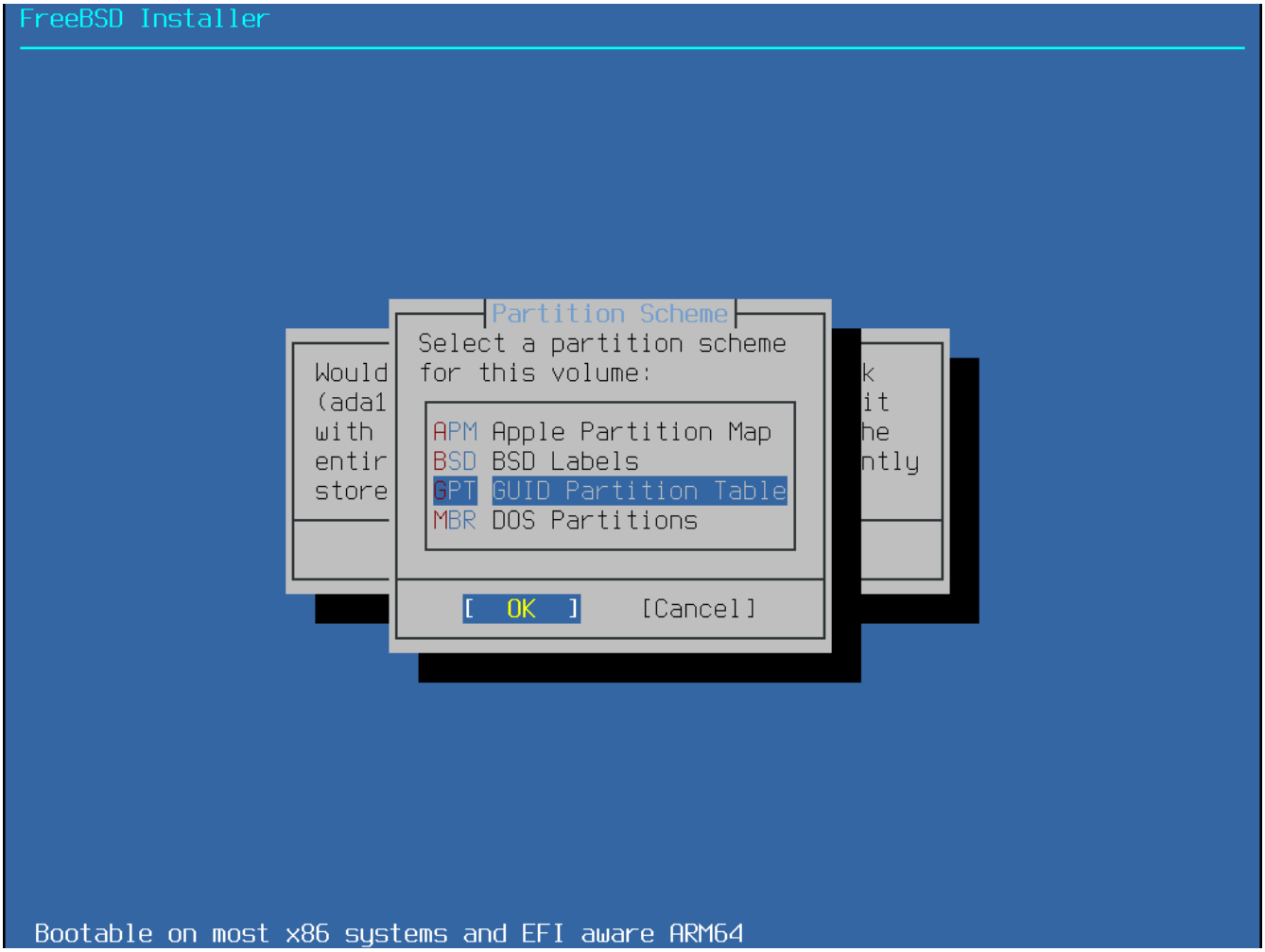
### 3.7.2. Гараар хуваах нь

Гараар хуваалт хийхэд шууд хуваалт засварлагч руу оруулдаг.




*Зураг 70. Хуваалтуудыг гараар хийх нь*

Хөтчийг (энэ жишээн дээр ada0) тодруулаад **[ Create ]**-г сонгоход *хуваалтын схемийн* төрлийг сонгож болох цэсийг харуулдаг.



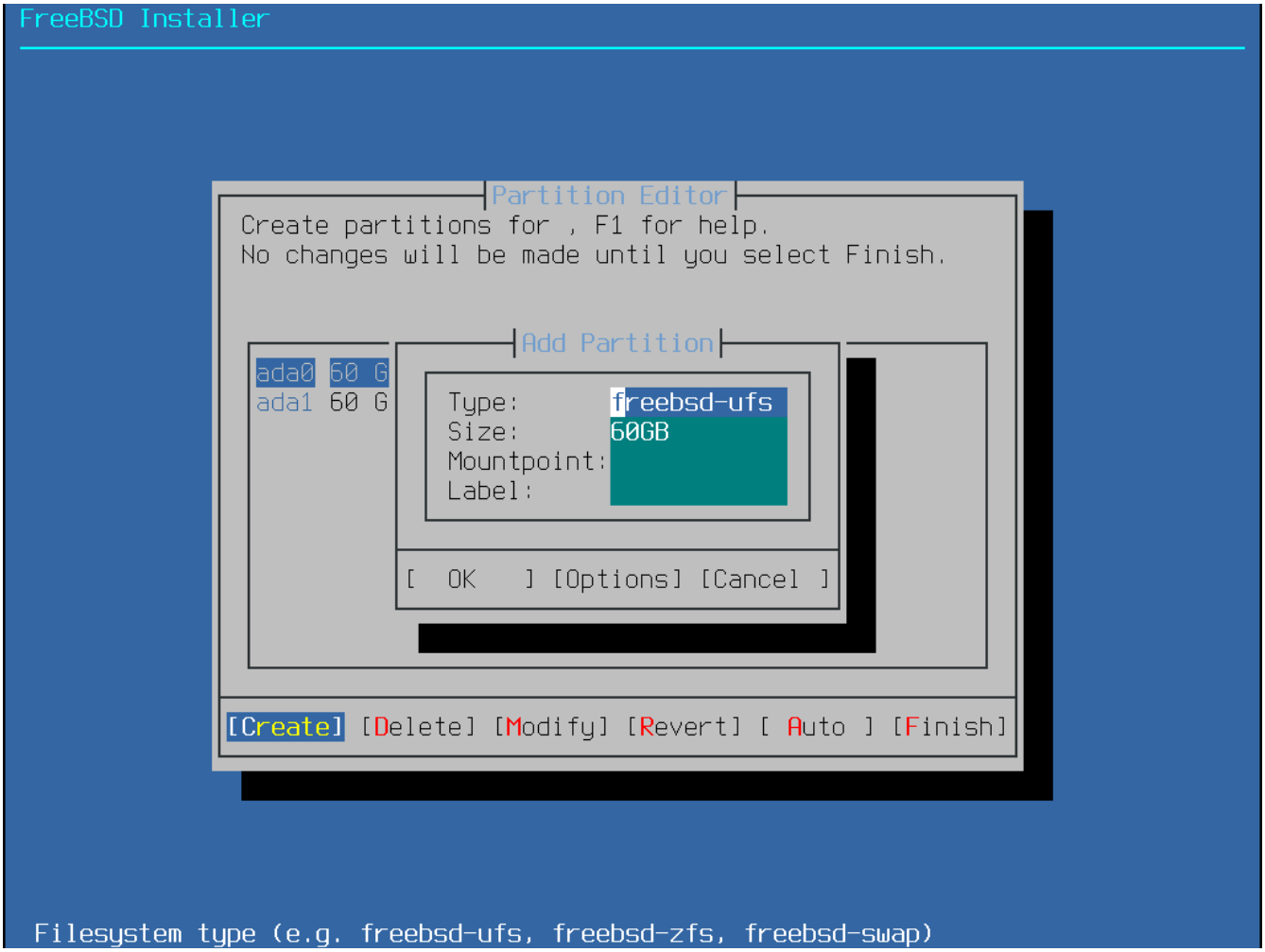

*Зураг 71. Хуваалтуудыг гараах хийх нь*

GPT хуваалт нь PC төрлийн компьютеруудын хувьд ихэвчлэн хамгийн зөв сонголт байдаг. GPT-тэй нийцтэй биш байж болох хуучин PC-н үйлдлийн системүүд MBR хуваалтыг шаардаж болох юм. Бусад хуваалтын схемүүдийг ердийн биш эсвэл хуучин компьютерийн системүүдийн хувьд ерөнхийдөө ашигладаг.

*Хүснэгт 5. Хуваалтын схемүүд*

| Товчлол | Тайлбар |
|---|---|
| APM | Apple Partition Map буюу PowerPC® Macintosh®-н хэрэглэдэг Apple-н Хуваалт. |
| BSD | MBR-гүй BSD-н хаягууд, заримдаа "dangerously dedicated mode" гэгддэг. bsdlabel(8)-с үзнэ үү. |
| GPT | GUID Хуваалтын Хүснэгт. |
| MBR | Master Boot Record буюу Мастер Ачаалах Бичлэг |
| PC98 | MBR-н хувилбар, NEC PC-98 компьютерууд ашигладаг. |
| VTOC8 | Volume Table Of Contents, Sun SPARC64 бас UltraSPARC компьютерууд ашигладаг. |

Хуваалтын схемийг сонгож үүсгэсний дараа **[ Create ]**-г сонгосноор шинэ хуваалтууд үүсгэх болно.




*Зураг 72. Гараар хуваалтуудыг үүсгэх нь*

Стандарт FreeBSD-н GPT суулгалт багаар бодоход гурван хуваалт ашигладаг:

*Стандарт FreeBSD GPT хуваалтууд*

- `freebsd-boot` - FreeBSD ачаалах код.
- `freebsd-ufs` - FreeBSD UFS файлын систем.
- `freebsd-swap` - FreeBSD swap зай.

Өөр нэг дурдахад илүүдэхгүй хуваалт бол FreeBSD ZFS файлын систем агуулах хуваалтуудад хэрэглэгдэх `freebsd-zfs` хуваалт юм. Уламжлалт тусдаа файлын системийн хуваалтуудыг үүсгэх-г харна уу. gpart(8) нь илүү олон GPT хуваалтын төрлүүдийг харуулдаг.

Файлын системийн олон хуваалтыг ашиглаж болох бөгөөд зарим хүмүүс /, /var, /tmp ба /usr файлын системүүдийн хувьд тусдаа хуваалт бүхий уламжлалт тогтцыг илүүд үзэж болох юм. Жишээг Уламжлалт тусдаа файлын системийн хуваалтуудыг үүсгэх хаягаас үзнэ үү.

Хэмжээг ердийн товчлолоор оруулж өгч болно: Килобайтын хувьд *K*, Мегабайтын хувьд *M* эсвэл гигабайтын хувьд *G* гэнэ.

Секторын зөв тэгш байдал ажиллагааг сайжруулах бөгөөд олон 4K байт

> хэмжээнд тэгш хуваагдахаар хуваалтын хэмжээг өгөх нь 512-байт эсвэл 4K байт сектор бүхий хөтчүүд дээр тэгш байдлыг хангахад тусална. Ерөнхийдөө олон 1M эсвэл 1G-д тэгш хуваагдахаар байх хуваалтын хэмжээг ашиглах нь хуваалт бүр тэгш тооны 4K-с эхлэх боломжийг бий болгодог. Гэхдээ нэг бодох юм нь *freebsd-boot* хуваалт нь ачаалах кодын хязгаарлалтаас болоод 512K-с ихгүй байх ёстой байдаг.

Хэрэв энэ хуваалт нь файлын систем агуулах бол холболтын цэг хэрэгтэй болдог. Хэрэв зөвхөн ганц UFS хуваалт үүсэх бол холболтын цэг нь / байна.

*хаяг*ийг бас асуудаг. Хаяг нь энэ хуваалтыг таних нэр болдог. Хөтчийн нэр эсвэл тоо нь хөтөч өөр хянагч юм уу порт руу холбогдсон бол өөрчлөгдөж болох боловч хуваалтын хаяг өөрчлөгдөхгүй. /etc/fstab гэх мэт файлууд дээрх хөтчийн нэр ба хуваалтын тоонуудын оронд хаягийг ашиглах нь тоног төхөөрөмжийн өөрчлөлттэй холбоотой асуудал үүсгэлгүй системийг найдвартай ажиллагааг илүү болгодог. GPT хаягууд нь диск холбогдсон үед /dev/gpt/ гэж орж ирдэг. Бусад хуваалтын схемүүд өөр хаягийн боломжуудтай бөгөөд тэдгээрийн хаягууд нь /dev/ дотор өөр сангуудад гарч ирдэг.

> Ижил хаягуудтай байхаас сэргийлж файл систем бүрт өөр хаяг ашиглах хэрэгтэй. Компьютерийн нэр, хэрэглээ, эсвэл байрлалаас хэдэн үсгийг хаягт нэмж өгч болно. Жишээ нь лабораторийн компьютерийн хувьд түүний UFS root хуваалтыг "labroot" юм уу эсвэл "rootfs-lab" гэж өгч болох юм.

*Жишээ 5. Уламжлалт тусдаа файлын системийн хуваалтуудыг үүсгэх*

/, /var, /tmp, болон /usr сангууд тусдаа файл систем хэлбэрээр өөр өөрийн хуваалтууд дээр байрлах уламжлалт хуваалтуудын хувьд GPT хуваалтын схем үүсгээд доор үзүүлсэн шиг хуваалтыг үүсгэх хэрэгтэй. Үзүүлсэн хуваалтын хэмжээнүүд нь 20G дискний хувьд ийм байх нь элбэг байдаг. Хэрэв диск дээр илүү зай байгаа бол илүү том swap юм уу эсвэл /var хуваалт байх нь хэрэгтэй байдаг. Энд үзүүлсэн хаягуудын өмнө "жишээ нь" `ex` тавигдсан байгаа бөгөөд уншигч нар дээр тайлбарласны дагуу өөр бусдаас ялгаатай хаягийн утгыг ашиглах хэрэгтэй.

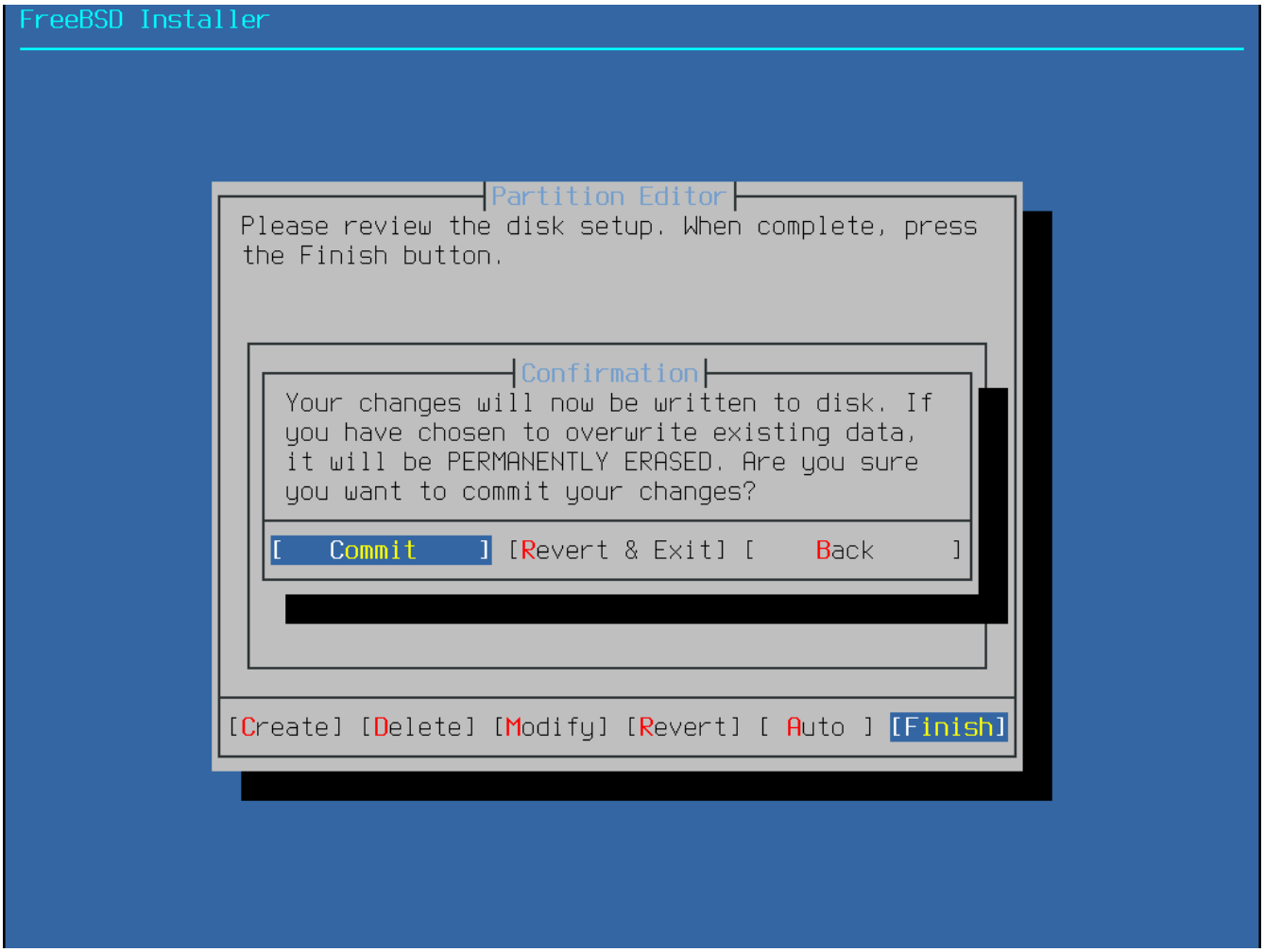
FreeBSD-н gptboot нь анхдагчаар эхний UFS хуваалтыг / хуваалт байна гэж үздэг.

| Хуваалтын төрөл | Хэмжээ | Холбох цэг | Хаяг |
|---|---|---|---|
| `freebsd-boot` | `512K` | | |
| `freebsd-ufs` | `2G` | / | `exrootfs` |
| `freebsd-swap` | `4G` | | `exswap` |
| `freebsd-ufs` | `2G` | /var | `exvarfs` |
| `freebsd-ufs` | `1G` | /tmp | `extmpfs` |
| `freebsd-ufs` | анхдагчийг хүлээн авах (дискний үлдсэн хэсэг) | /usr | `exusrfs` |

Хуваалтыг үүсгэсний дараа суулгалтыг **[ Finish ]** дарж үргэлжлүүлнэ.

## 3.8. Суулгалтыг гүйцэтгэх нь

Энэ мөч нь суулгалт хатуу диск рүү өөрчлөлт хийхээс өмнө зогсоох хамгийн сүүлийн боломж юм.



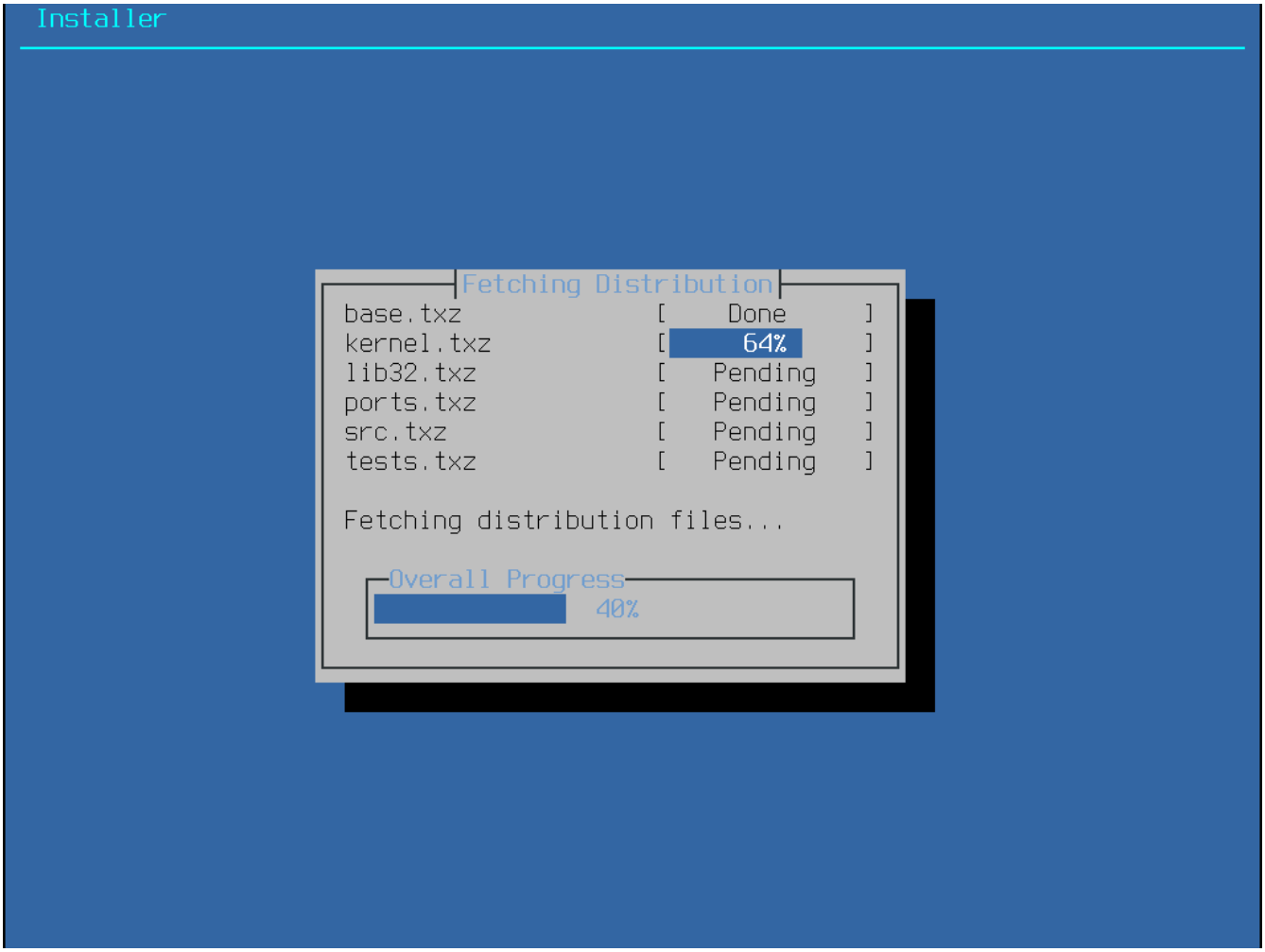

*Зураг 73. Сүүлийн лавлагаа*

**[ Commit ]**-г сонгож `Enter` дарж үргэлжлүүлнэ. Хэрэв өөрчлөлт хийх хэрэгтэй бол **[ Back ]** дарж хуваалт засварлагч руу буцна. **[ Revert & Exit ]** нь хатуу диск рүү ямар ч өөрчлөлт хийлгүйгээр суулгагчаас гаргана.

Сонгосон түгээлтүүд, суулгалт зөөвөрлөгч болон компьютерийн хурдаас хамаарч суулгах хугацаа харилцан адилгүй байдаг Процесс явж байгааг харуулсан хэд хэдэн зурвасууд гарч ирдэг.

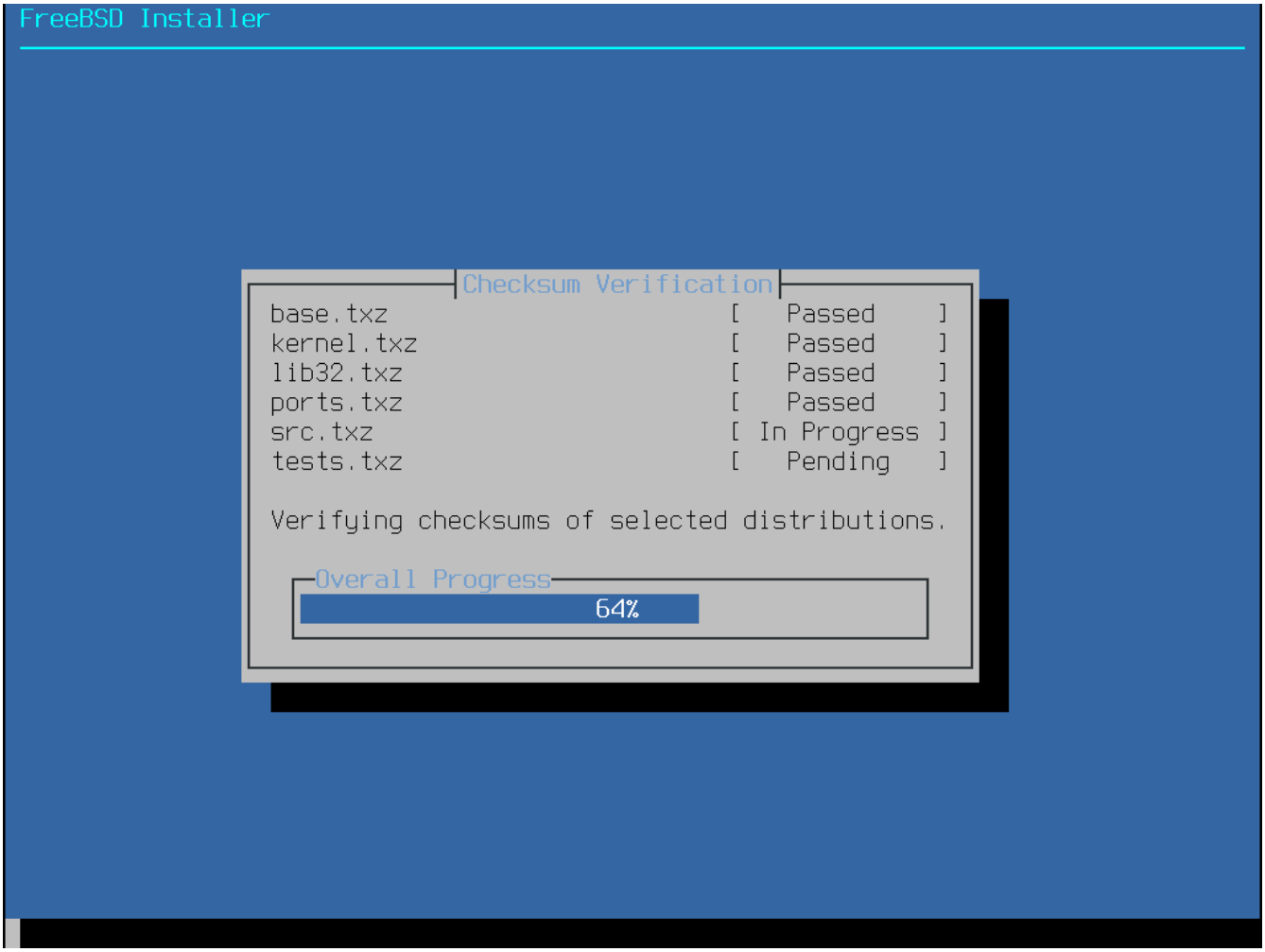
Эхлээд суулгагч хуваалтуудыг диск рүү бичиж `newfs` хийж хуваалтуудыг бэлэн болгоно.

Хэрэв сүлжээгээр суулгаж байгаа бол bsdinstall шаардлагатай түгээлтийн файлуудыг татаж авах болно.



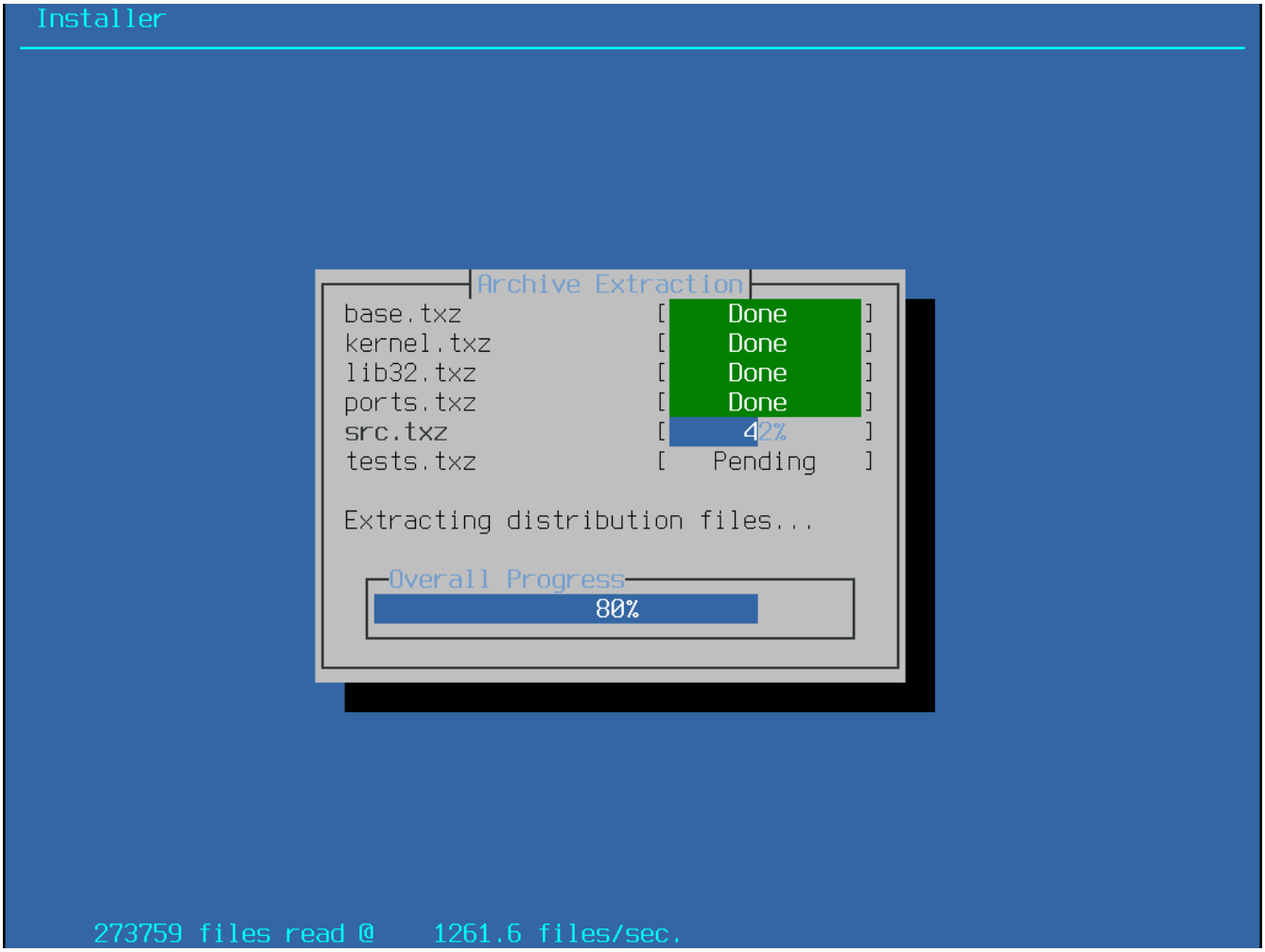

*Зураг 74. Түгээлтийн файлуудыг татах нь*

Дараа нь татаж авсан эсвэл суулгалтын зөөвөрлөгчөөс уншсан файлууд нь эвдрээгүйг магадлахын тулд түгээлтийн файлуудын бүрэн бүтэн байдлыг шалгадаг.




*Зураг 75. Түгээлтийн файлуудыг шалгах нь*

Төгсгөлд нь шалгагдсан түгээлтийн файлуудыг диск рүү задалдаг.




*Зураг 76. Түгээлтийн файлуудыг задлах нь*

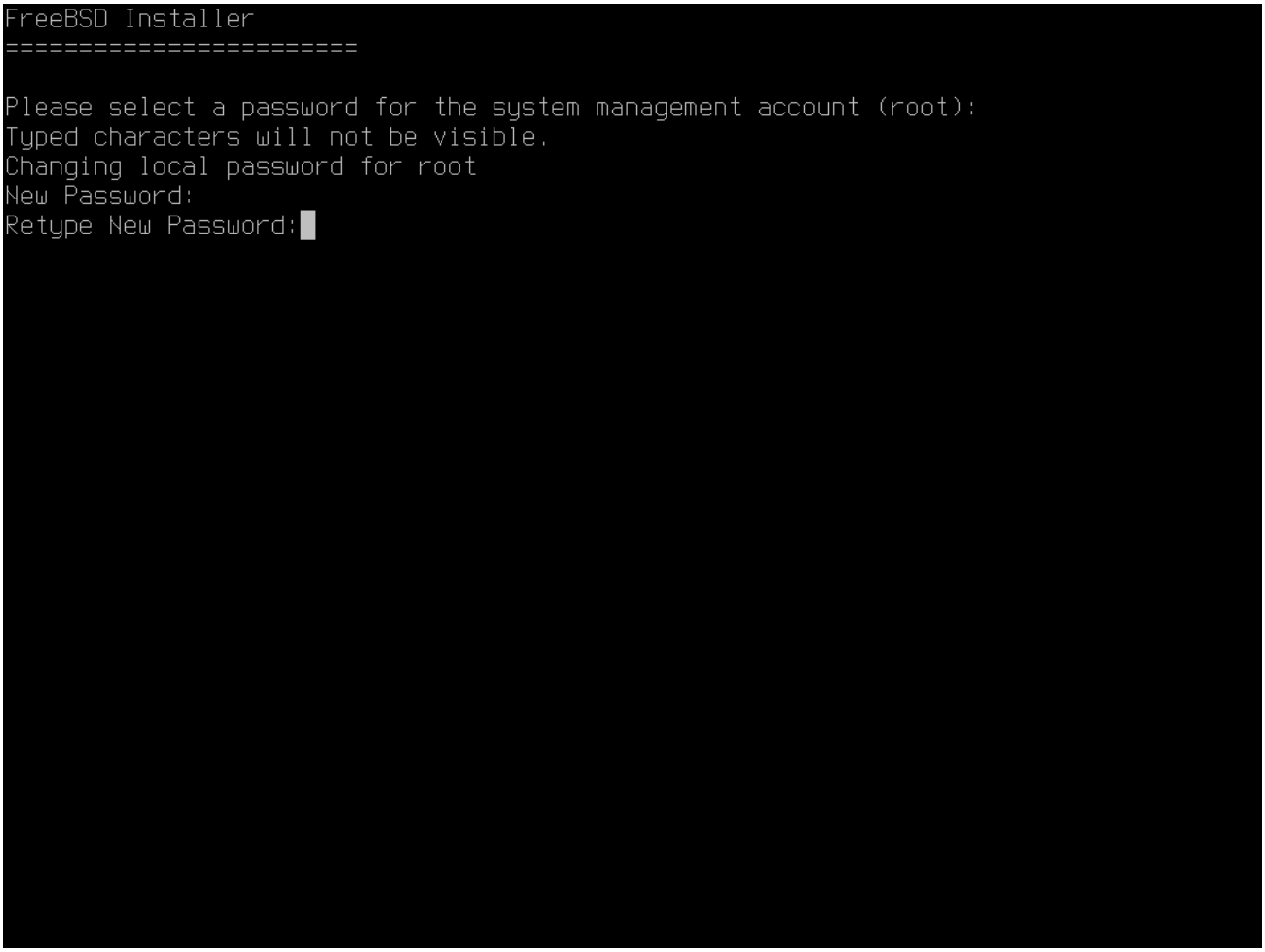
Бүх хүссэн түгээлтийн файлуудыг задалсны дараа bsdinstall нь суулгалт дууссаны дараах тохиргооны ажлууд руу ордог. (Суулгасны дараах үйлдлүүд-г үзнэ үү).

# 3.9. Суулгасны дараах үйлдлүүд

Төрөл бүрийн тохиргоо хийсний дараа FreeBSD амжилттай сууна. Шинэ суулгасан FreeBSD систем рүү ачаалахаасаа өмнө төгсгөлийн цэсээс тохиргооны сонголтууд руу орж тохиргоог өөрчилж болно.

## 3.9.1. `root` нууц үгийг тохируулах

`root` нууц үгийг өгөх хэрэгтэй. Нууц үгийг оруулж байхад дэлгэц дээр бичсэн тэмдэгтүүд харагддаггүйг санаарай. Нууц үгийг оруулсны дараа дахин оруулах ёстой. Энэ нь алдаа гарахаас сэргийлдэг.



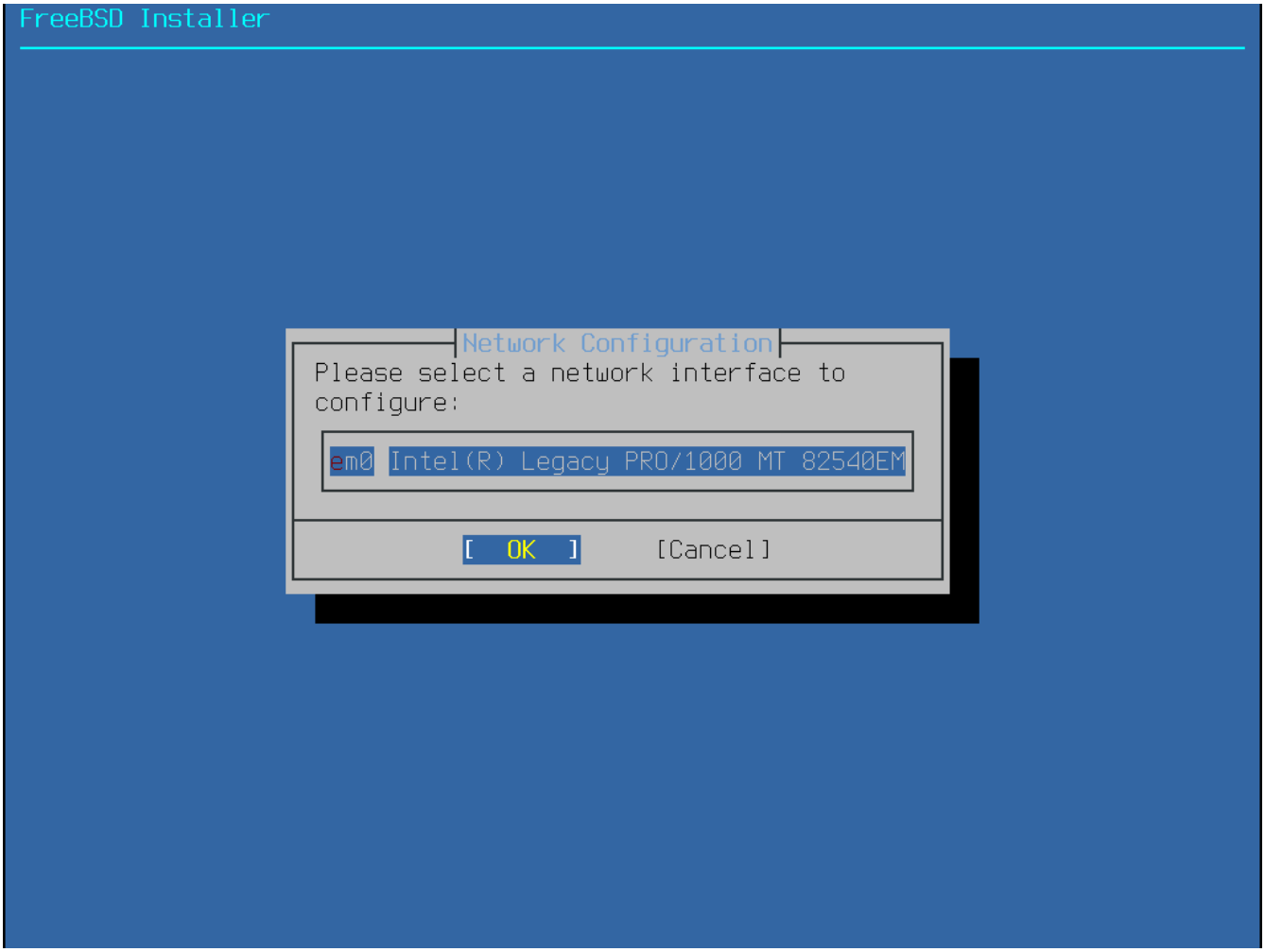

*Зураг 77. `root` нууц үгийг тохируулах*

Нууц үгийг оруулсны дараа суулгалт үргэлжлэх болно.

### 3.9.2. Сүлжээний интерфэйсүүдийг тохируулах нь

*bootonly* суулгалтын үеэр тохиргоо хийгдсэн бол сүлжээний тохиргоог алгасах болно.

Компьютер дээр олдсон бүх сүлжээний интерфэйсийн жагсаалтыг харуулна. Тохиргоо хийхээ сонгоорой.




*Зураг 78. Сүлжээний интерфэйсийг сонгоно*

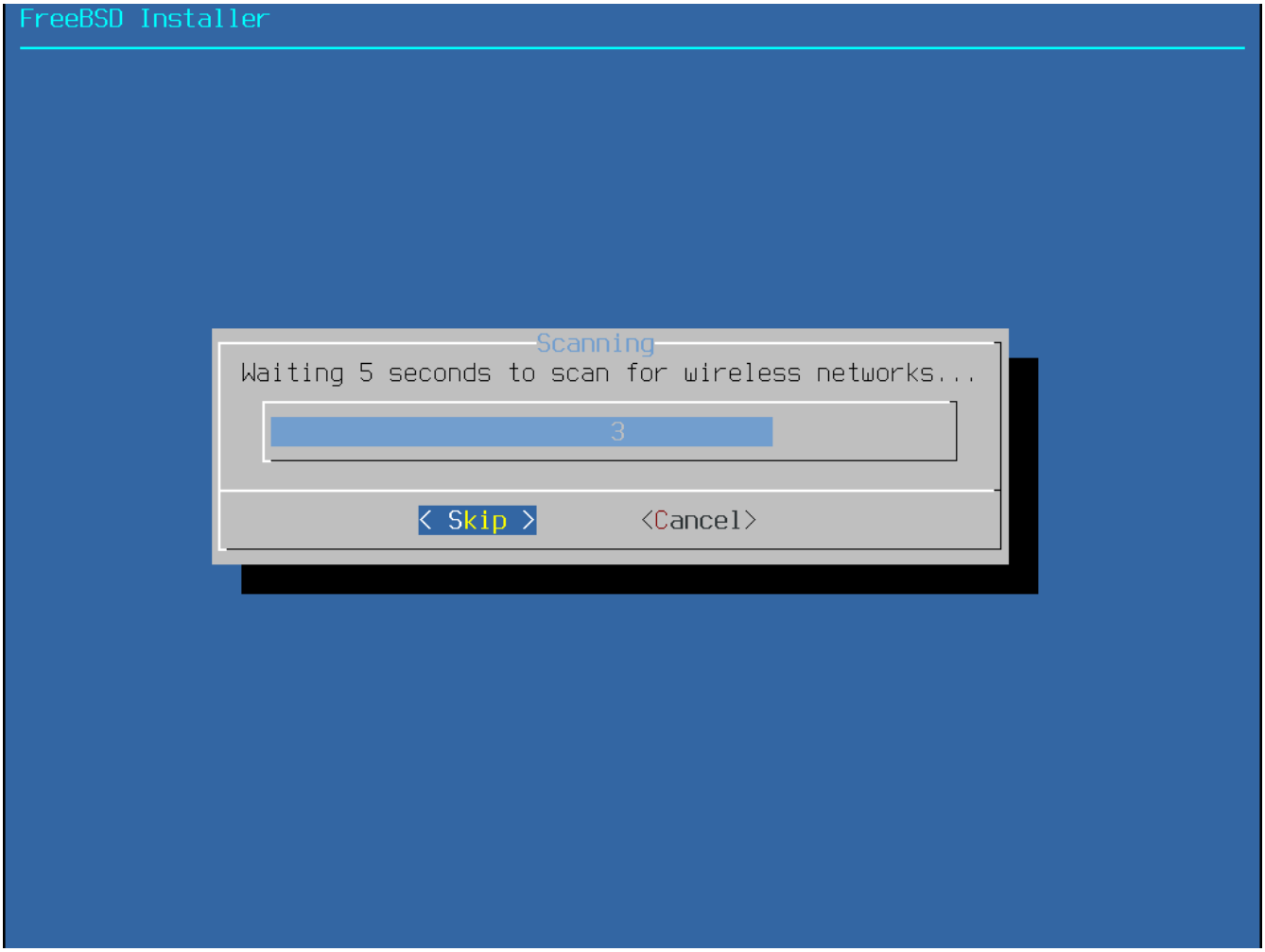
#### 3.9.2.1. Утасгүй сүлжээний интерфэйсийг тохируулах нь

Хэрэв утасгүй сүлжээний интерфэйс сонгосон бол сүлжээнид холбогдохын тулд утасгүй сүлжээний танилт болон аюулгүй байдалтай холбоотой параметрүүдийг оруулж өгөх хэрэгтэй.

Утасгүй сүлжээг Service Set Identifier буюу SSID-р танидаг. SSID нь богино бөгөөд сүлжээ бүрийн хувьд өөр нэр өгдөг.

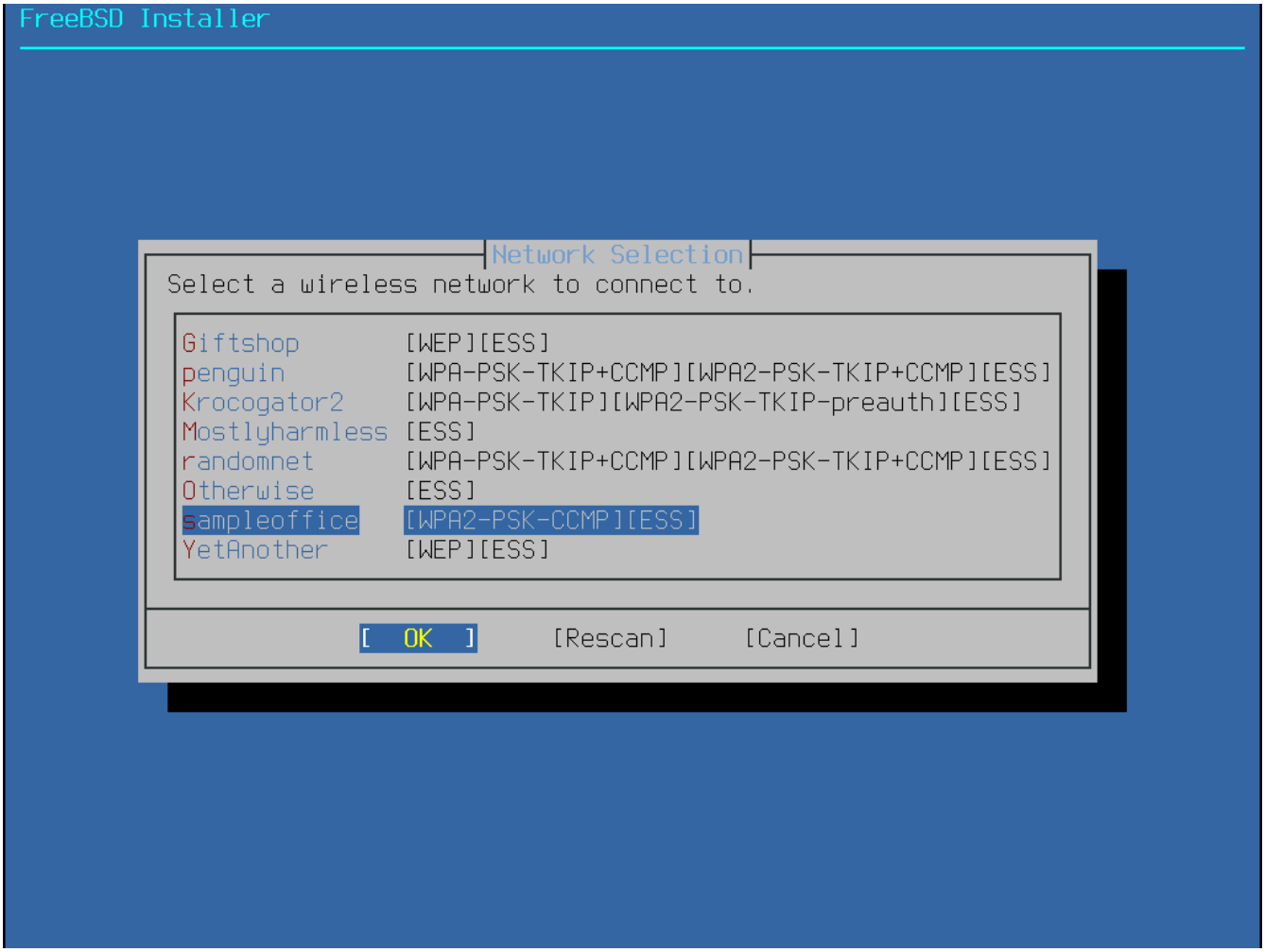
Ихэнх утасгүй сүлжээнүүд мэдээллийг дурын этгээд харахаас урьдчилан сэргийлэхийн тулд дамжуулсан өгөгдлийг шифрлэж нууцалдаг. WPA2-г ер нь илүүтэй зөвлөдөг. WEP гэх мэт хуучин шифрлэлтийн төрлүүд нь маш бага аюулгүй байдлыг хангадаг.

Утасгүй сүлжээнд холбогдох эхний алхам бол утасгүй сүлжээний холболтын цэгийг хайх явдал байдаг.



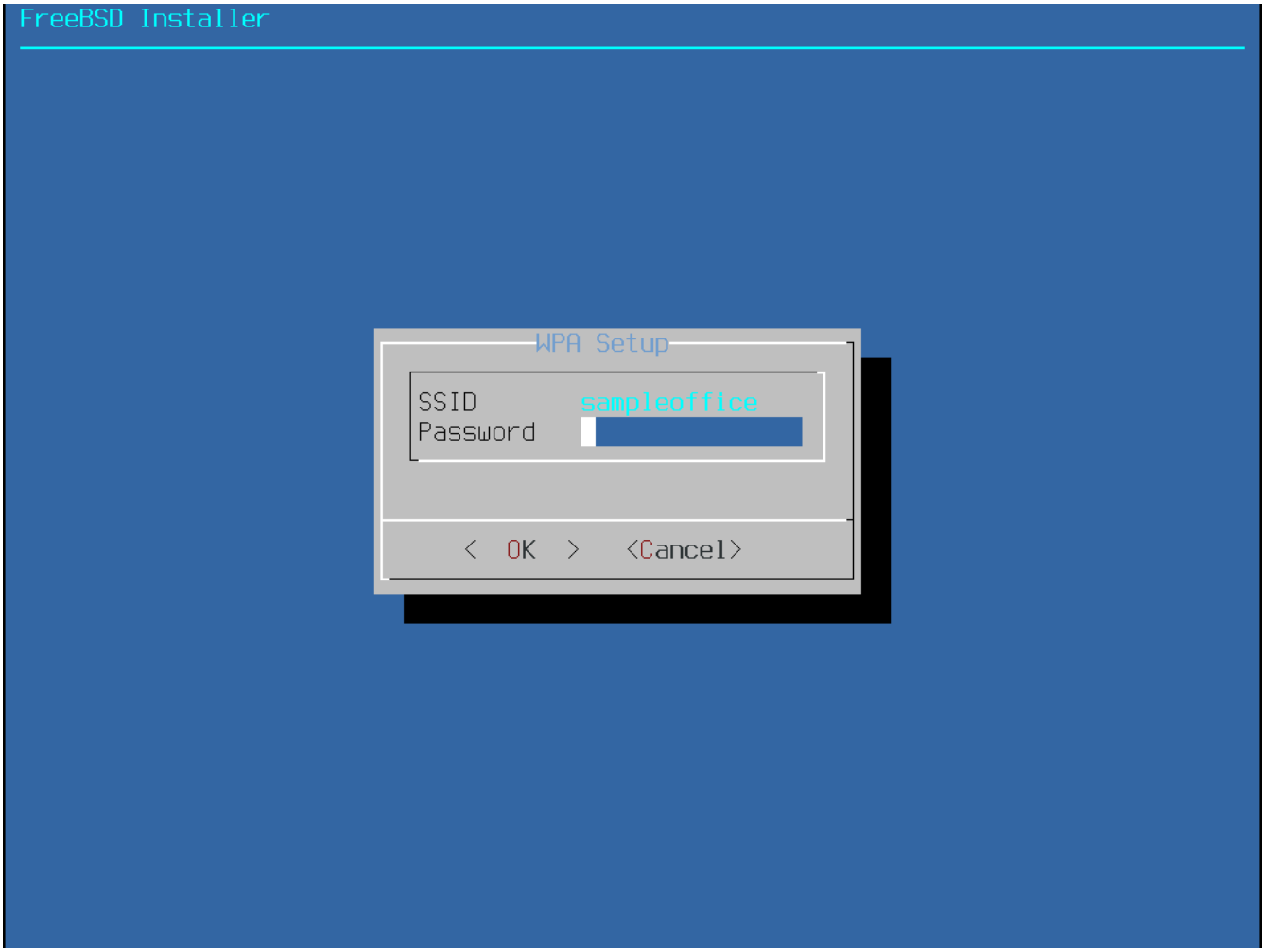

*Зураг 79. Утасгүй сүлжээний холболтын цэгийг хайх*

Хайх үед олдсон SSID -ууд болон тэдгээрийн шифрлэлтийн төрлийн талаарх тайлбар харагдах болно. Хэрэв SSID жагсаалтад харагдахгүй бол **[ Rescan ]**-г сонгож дахиж хайгаарай. Хэрэв хүссэн сүлжээ дахиад гарч ирэхгүй байгаа бол антенны холболтоо шалгах юм уу эсвэл холболтын цэг рүү компьютераа ойртуулж үзээрэй. Өөрчлөлт хийх болгондоо дахиж хайж үзээрэй.




*Зураг 80. Утасгүй сүлжээг сонгох нь*

Сүлжээг сонгосны дараа сонгосон утасгүй сүлжээндээ холбогдохдоо шифрлэлтийн мэдээллээ оруулна. WPA2 ашиглавал зөвхөн нууц үг (Урьдчилсан хуваалцсан түлхүүр буюу Pre-Shared Key, товчоор PSK) хэрэгтэй. Оруулах цонхон дээр бичсэн тэмдэгтүүд аюулгүй байдалтай холбоотойгоор од хэлбэрээр харагдана.

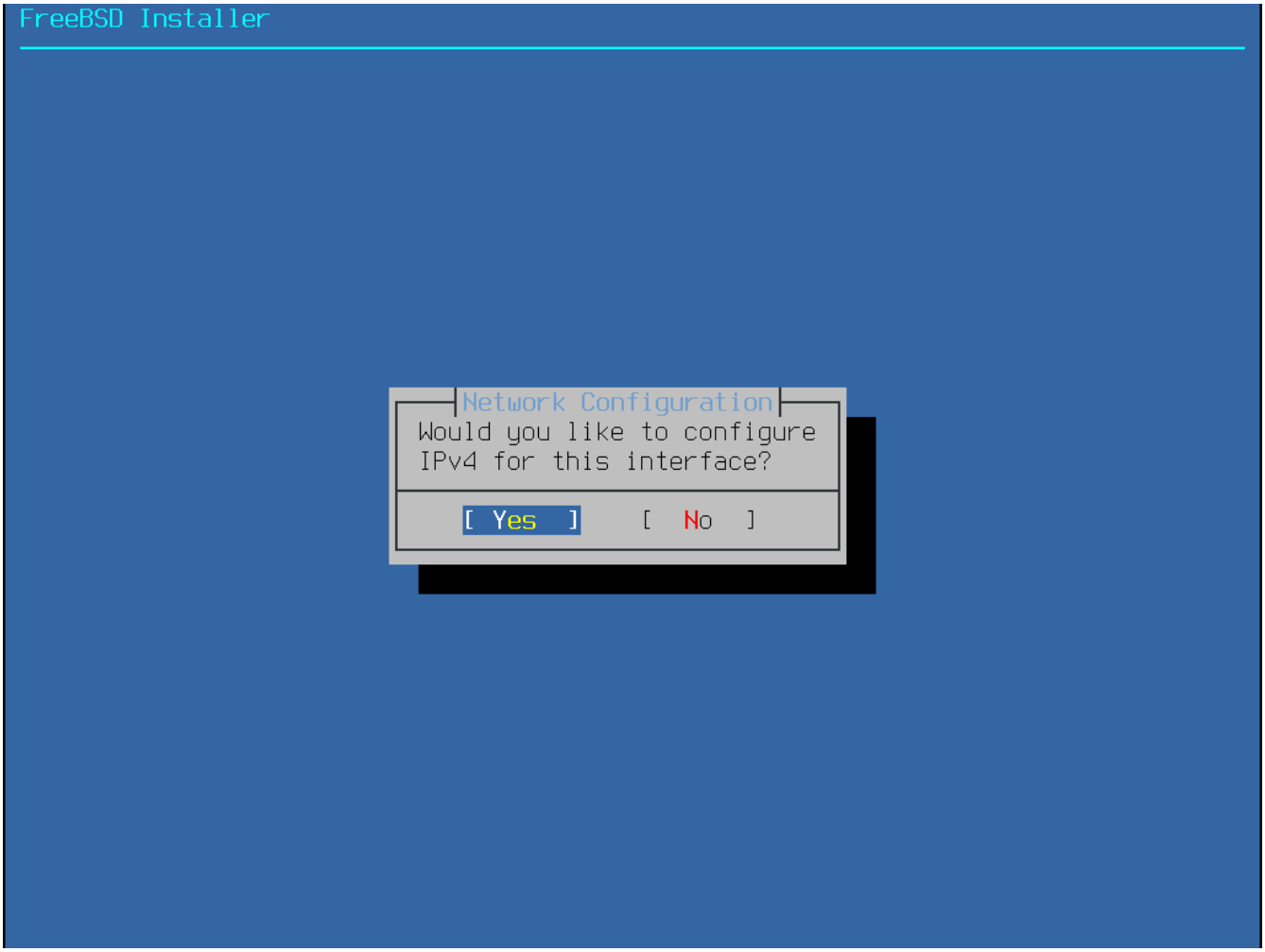



*Зураг 81. WPA2 тохируулга*

Утасгүй сүлжээг сонгож холболтын мэдээллийг оруулсны дараа сүлжээний тохиргоо үргэлжилнэ.

#### 3.9.2.2. IPv4 сүлжээг тохируулах нь

IPv4 сүлжээ ашиглагдах эсэхийг сонгох хэрэгтэй. Энэ сүлжээний хамгийн түгээмэл төрөл юм.



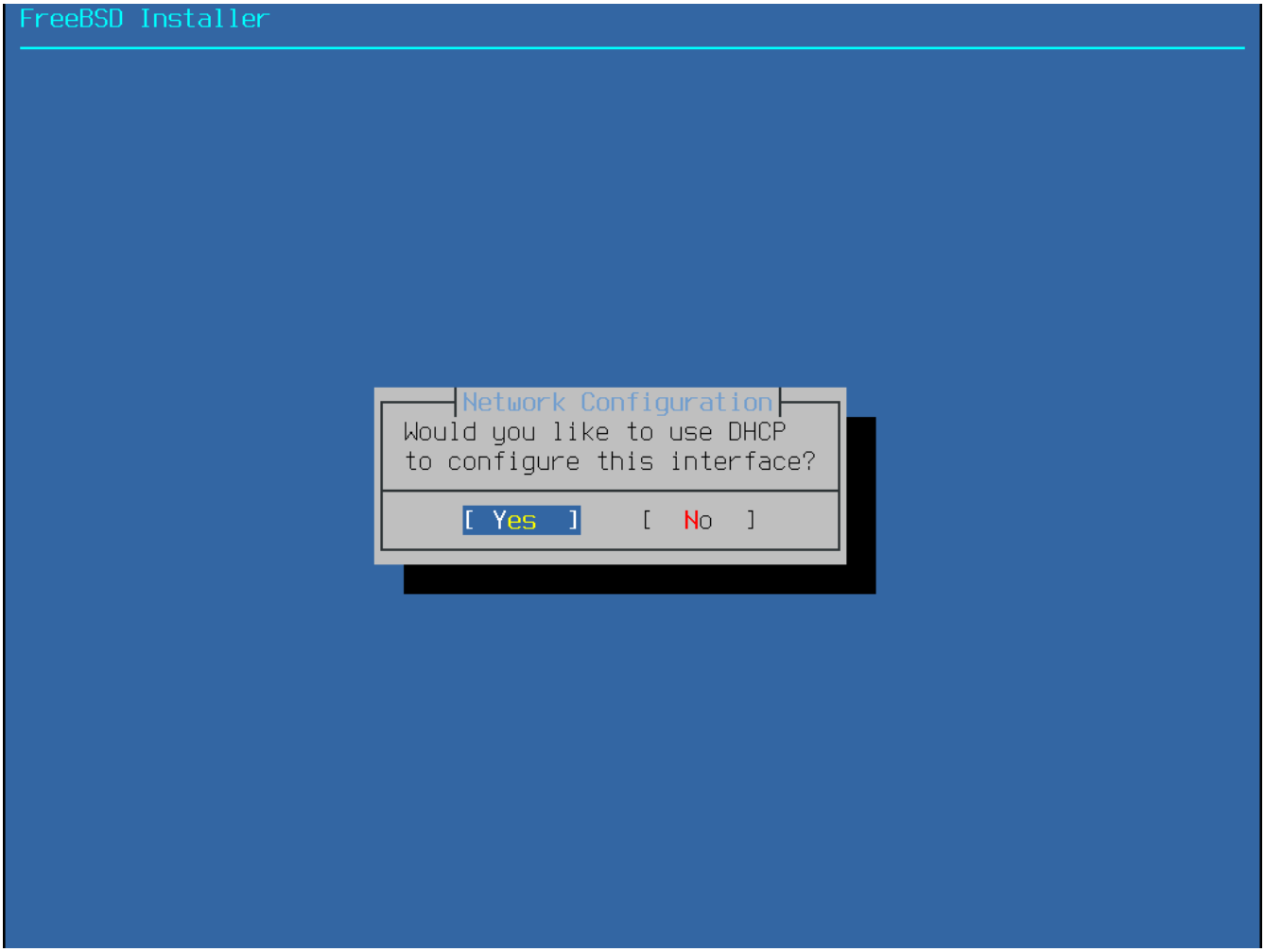

*Зураг 82. IPv4 сүлжээг сонгох нь*

IPv4 тохиргоог хийх хоёр арга байдаг. *DHCP* нь сүлжээний интерфэйсийг автоматаар тохируулах бөгөөд энэ аргыг илүүд үздэг. *Static* буюу тогтмол тохиргоо нь сүлжээний мэдээллийг гараас оруулах шаардлагыг бий болгодог.

Хамаагүй сүлжээний мэдээлэл оруулж болохгүй, учир нь ажиллахгүй. Сүлжээний администратор юм уу эсвэл үйлчилгээ үзүүлэгчээс Сүлжээний талаар мэдээлэл цуглуулах-д үзүүлсэн мэдээллийг авах хэрэгтэй.

#### 3.9.2.2.1. IPv4 DHCP сүлжээний тохиргоо

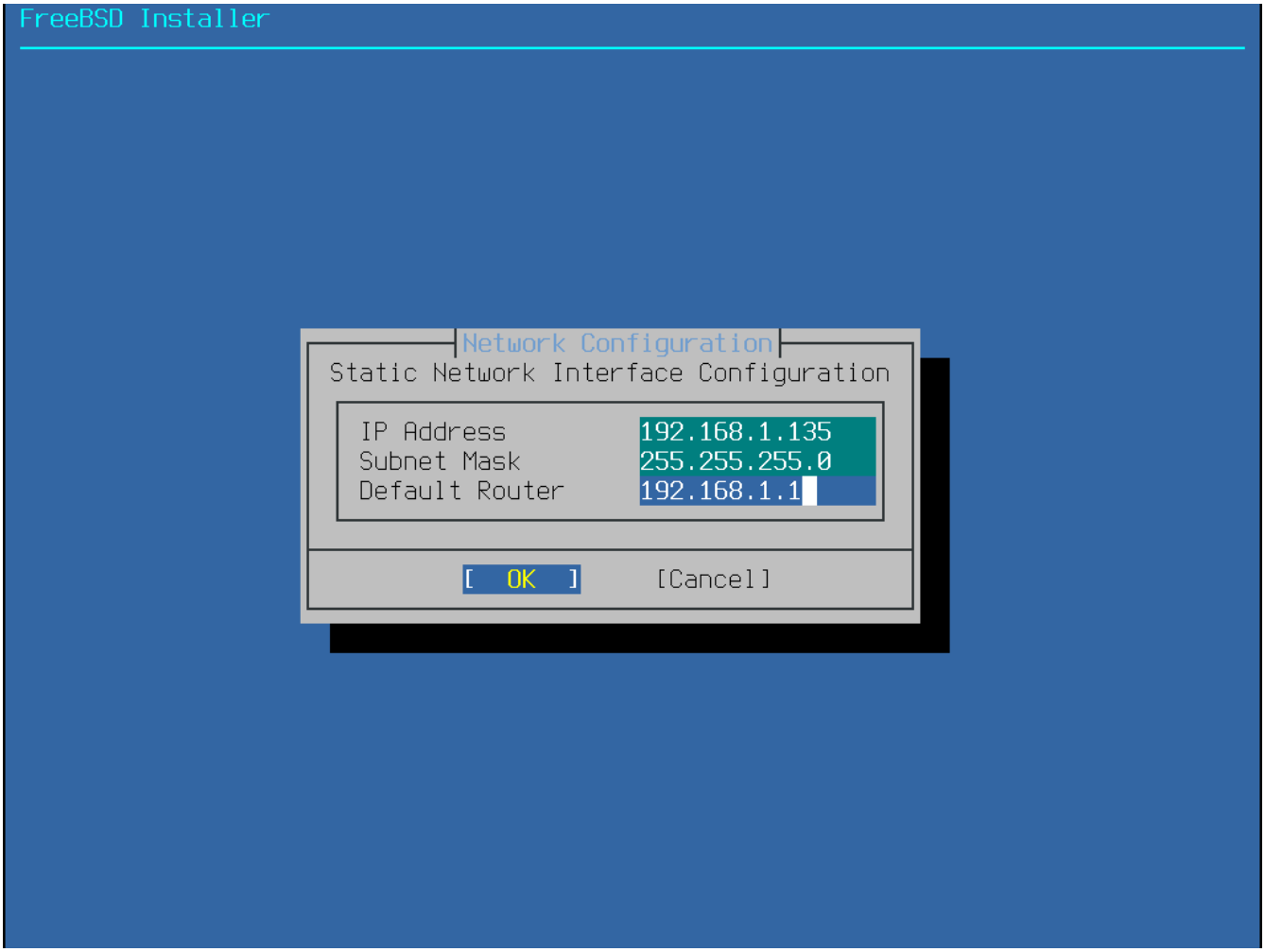
Хэрэв DHCP сервер байгаа бол автоматаар сүлжээгээ тохируулахын тулд **[ Yes ]**-г дарна.




*Зураг 83. IPv4 DHCP тохиргоог сонгох*

#### 3.9.2.2.2. IPv4 статик сүлжээний тохиргоо

Сүлжээний интерфэйсийн статик тохиргоо нь IPv4-н зарим мэдээлэл оруулахыг шаарддаг.

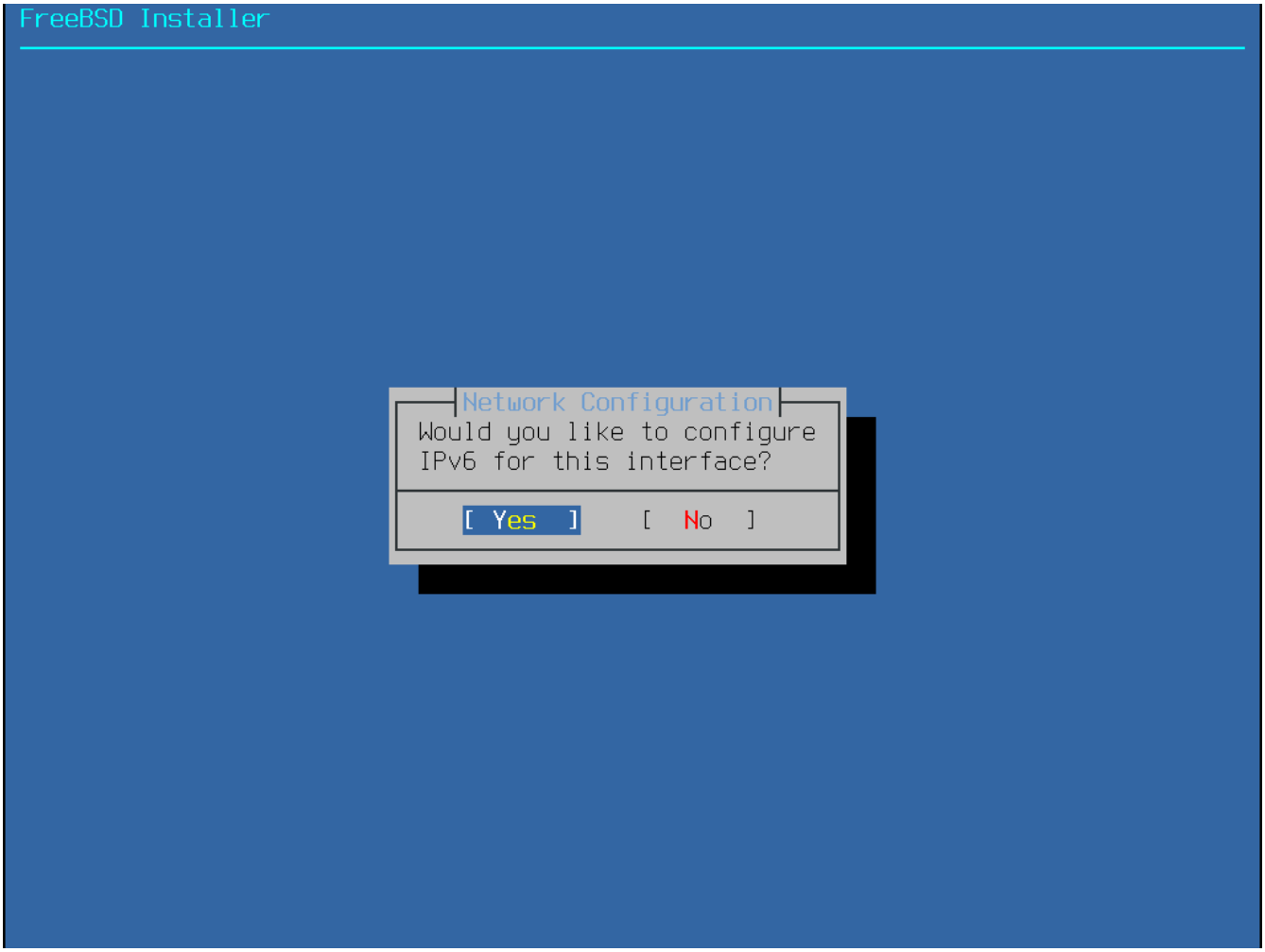



*Зураг 84. IPv4 статик тохиргоо*

- `IP хаяг` - Энэ компьютерт гараар зааж өгөх IPv4 хаяг. Энэ хаяг нь давхцах ёсгүй бөгөөд локал сүлжээн дээр өөр төхөөрөмж дээр ашиглагдаагүй байх ёстой.
- `Subnet Mask` - Локал сүлжээнд зориулсан сүлжээний маск. Ихэвчлэн `255.255.255.0` байдаг.
- `Default Router` - Энэ сүлжээн дэх анхдагч чиглүүлэгчийн IP хаяг. Ихэвчлэн локал сүлжээг Интернэт рүү холбодог чиглүүлэгч юм уу эсвэл өөр сүлжээний төхөөрөмжийн хаяг байдаг. Мөн *default gateway* буюу анхдагч гарц гэгддэг.

#### 3.9.2.3. IPv6 сүлжээг тохируулах нь

IPv6 нь сүлжээний тохиргооны шинэ арга юм. Хэрэв IPv6 байгаад түүнийг тохируулахыг хүсэж байгаа бол **[ Yes ]**-г сонгоно.

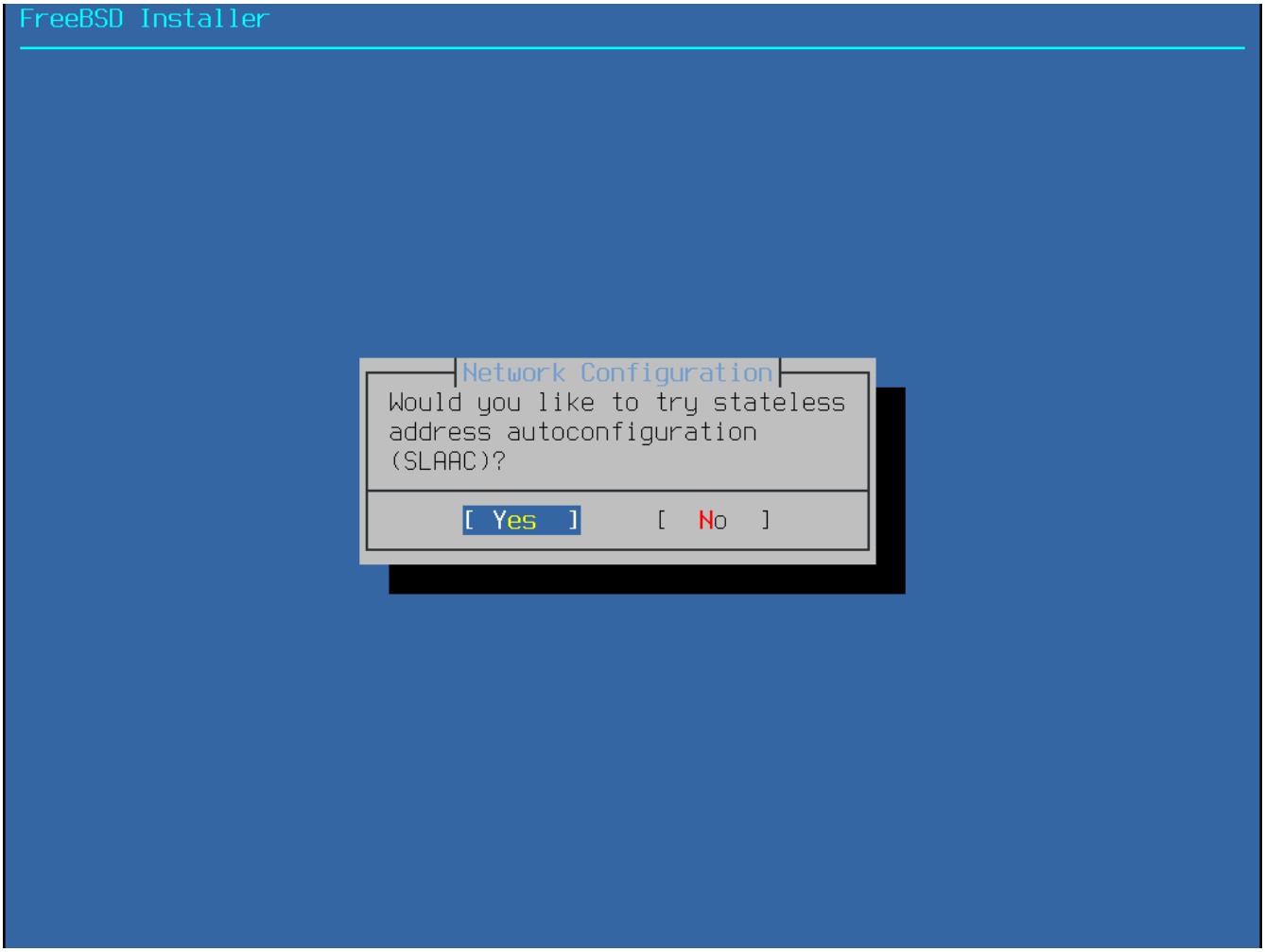



*Зураг 85. IPv6 сүлжээг тохируулахаар сонгох*

IPv6 нь бас тохируулах хоёр аргатай. *SLAAC* буюу *Төлөвт бус хаягийн автомат тохиргоо* нь сүлжээний интерфэйсийг автоматаар зөвөөр тохируулдаг. *Статик* тохиргоо нь сүлжээний мэдээллийг гараар оруулахыг шаарддаг.

#### 3.9.2.3.1. IPv6 Төлөвт бус хаягийн автомат тохиргоо

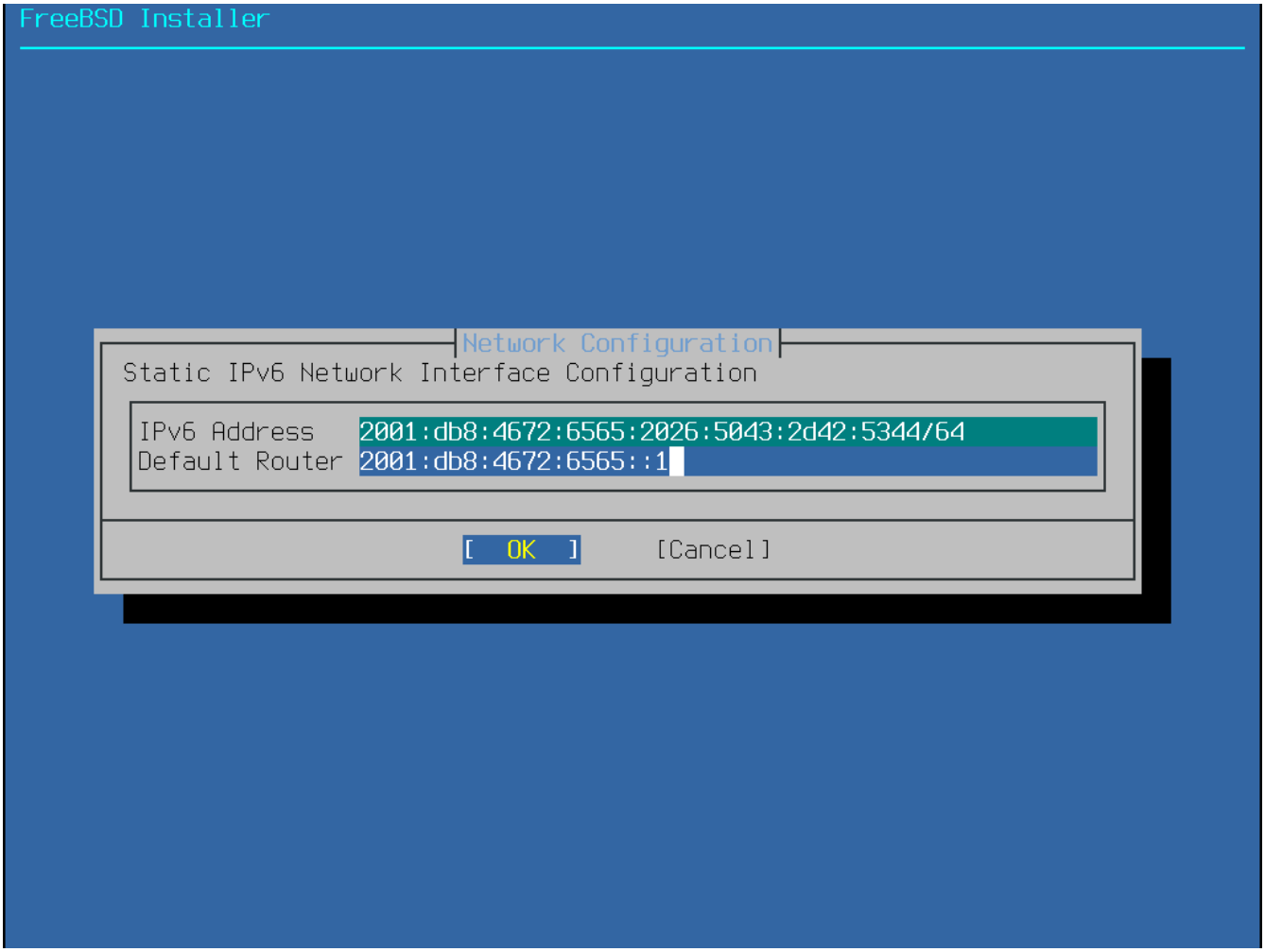
SLAAC нь локал чиглүүлэгчээс автомат тохиргооны мэдээлэл асуух боломжийг IPv6 сүлжээний төхөөрөмжид олгодог. RFC4862 хаягаас илүү мэдээллийг үзнэ үү.




*Зураг 86. IPv6 SLAAC тохиргоог сонгох*

#### 3.9.2.3.2. IPv6 статик сүлжээний тохиргоо

Сүлжээний интерфэйсийн статик тохиргоо нь IPv6 тохиргооны мэдээллийг оруулахыг шаарддаг.

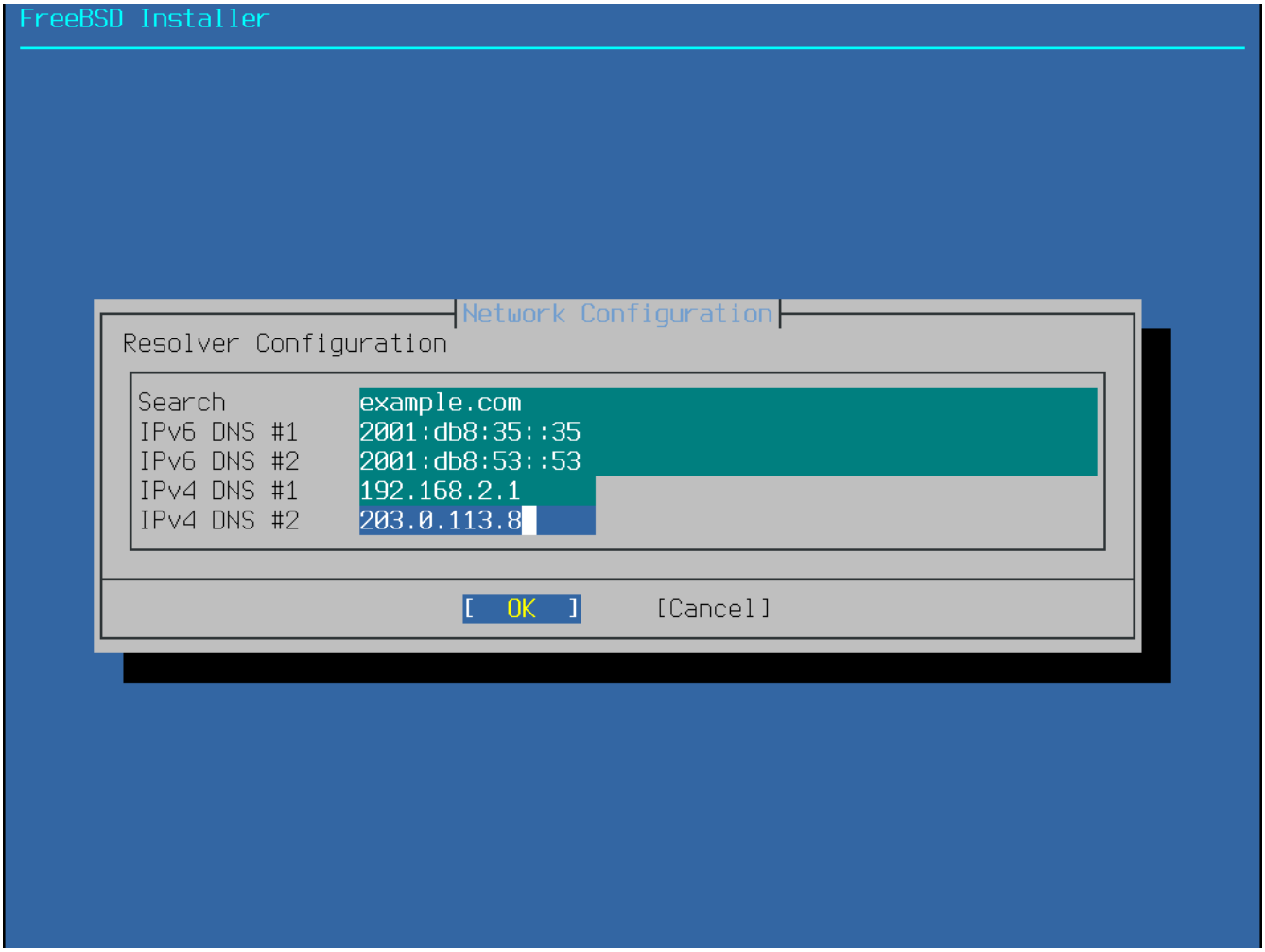



*Зураг 87. IPv6 статик тохиргоо*

- `IPv6 Address` - Гараар энэ компьютерт өгөх IP хаяг. Энэ хаяг нь давхцах ёсгүй бөгөөд локал сүлжээн дээр өөр төхөөрөмж дээр ашиглагдаагүй байх ёстой.
- `Default Router` - Энэ сүлжээн дэх анхдагч чиглүүлэгчийн IPv6 хаяг. Ихэвчлэн локал сүлжээг Интернэт рүү холбодог чиглүүлэгч юм уу эсвэл өөр сүлжээний төхөөрөмжийн хаяг байдаг. Мөн *default gateway* буюу анхдагч гарц гэгддэг.

#### 3.9.2.4. DNS-г тохируулах нь

*Domain Name System* ( *DNS*) танигч нь хостын нэрийг сүлжээний хаяг руу болон сүлжээний хаягаас хөрвүүлдэг. Хэрэв DHCP юм уу эсвэл SLAAC-г сүлжээний интерфэйсийг автоматаар тохируулахад ашиглаж байгаа бол Танигчийн тохиргооны утгууд аль хэдийн байж болох юм. Үгүй бол локал сүлжээний домен нэрийг хайх талбарт оруулна. DNS #1 ба DNS #2 нь локал DNS серверүүдийн IP хаягууд юм. Ядаж нэг DNS сервер хэрэгтэй байдаг.




*Зураг 88. DNS тохиргоо*

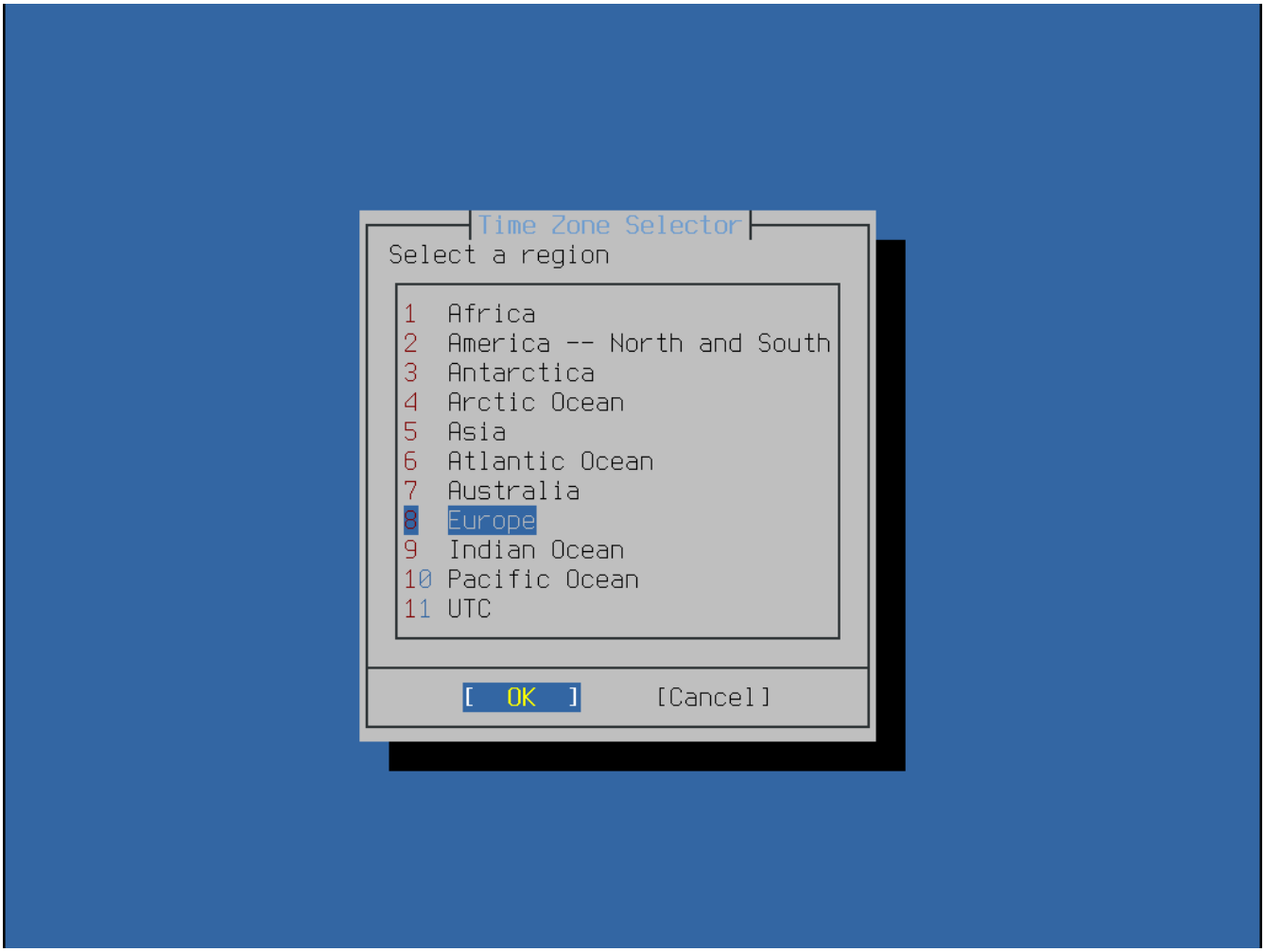
### 3.9.3. Цагийн бүсийг тохируулах нь

Цагийн бүсийг машиндаа зориулж тохируулах нь бүсийн цагийн өөрчлөлтүүдийг автоматаар дагах болон бусад цагийн бүстэй холбоотой функцуудыг зөв хийх боломжийг олгодог.

Энд байгаа жишээ нь АНУ-н Зүүн цагийн бүсэд байгаа машины хувьд юм. Таны сонголт газар зүйн бүсээс хамааран өөр өөр байх болно.

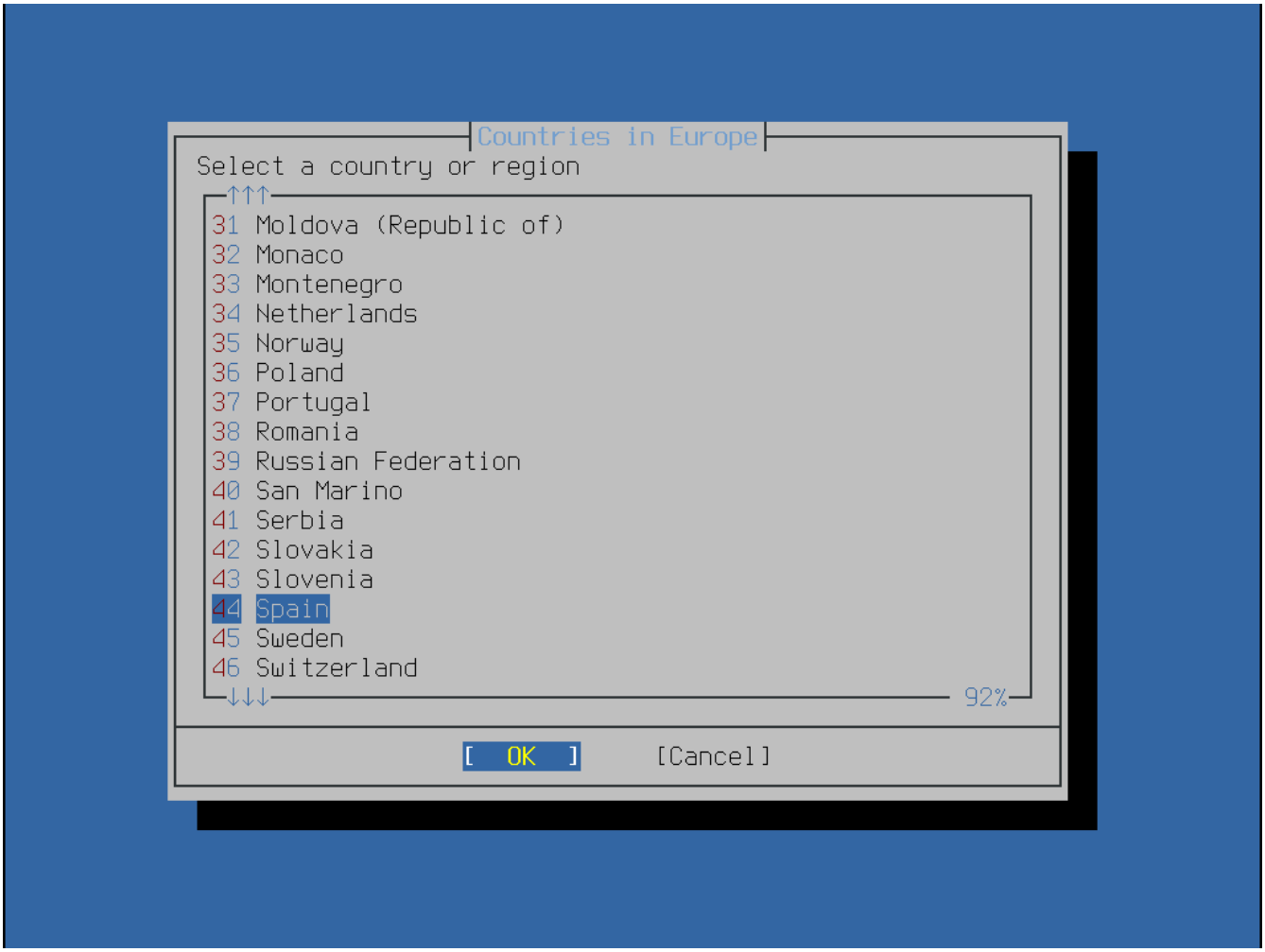
*Зураг 89. Локал эсвэл UTC цагийг сонгох*

Машины цаг хэрхэн тохируулагдсан байгаагаас хамаарч **[ Yes ]** эсвэл **[ No ]**-г сонгоод `Enter` дарна. Систем UTC эсвэл локал цаг алийг ашиглаж байгааг та мэдэхгүй байгаа бол **[ No ]**-г дарж хамгийн түгээмэл хэрэглэгддэг локал цагийг сонгоорой.



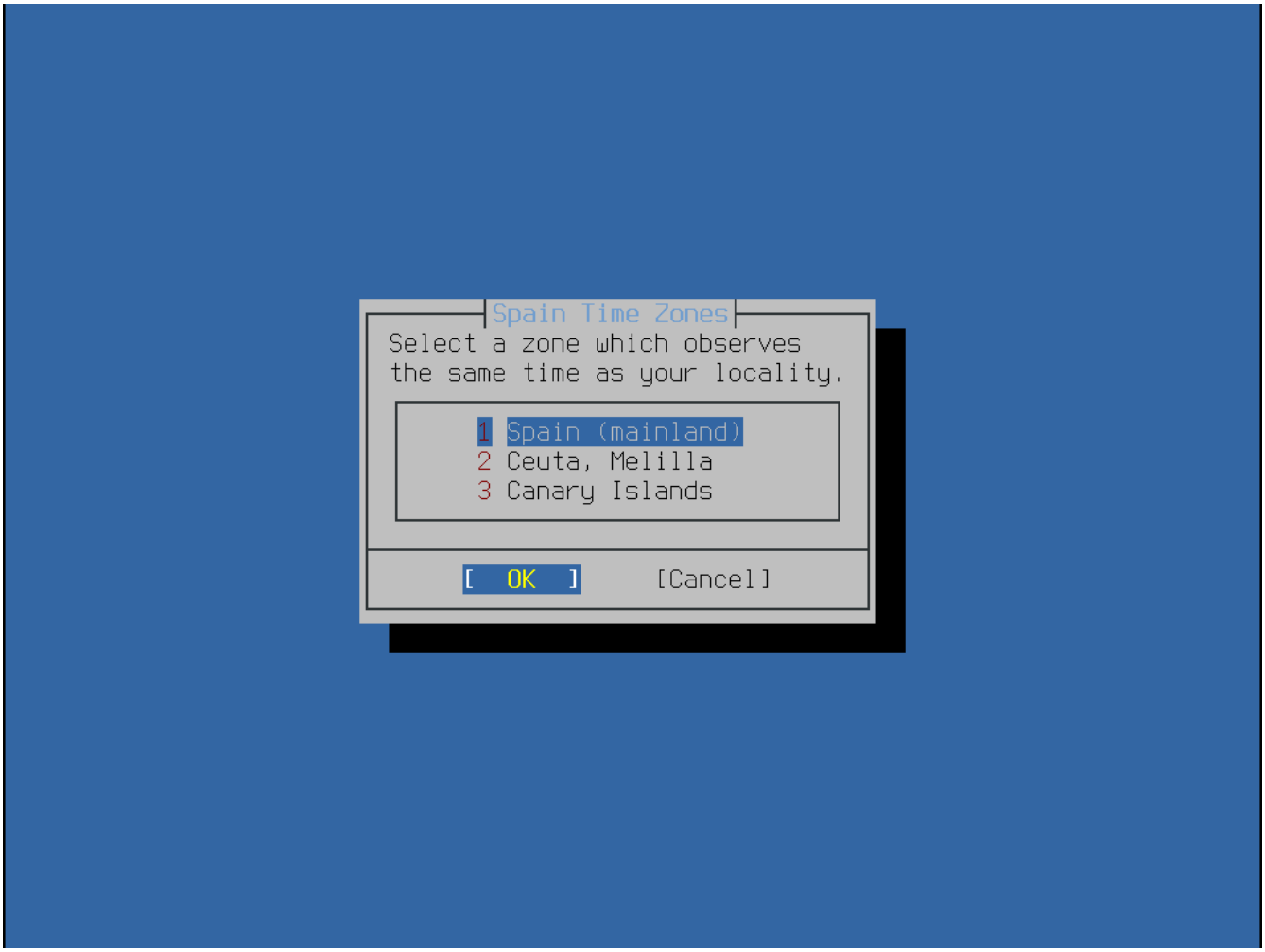

*Зураг 90. Бүсийг сонгох*

Сумнууд ашиглаж зохих бүсийг сонгоод `Enter` дарна.



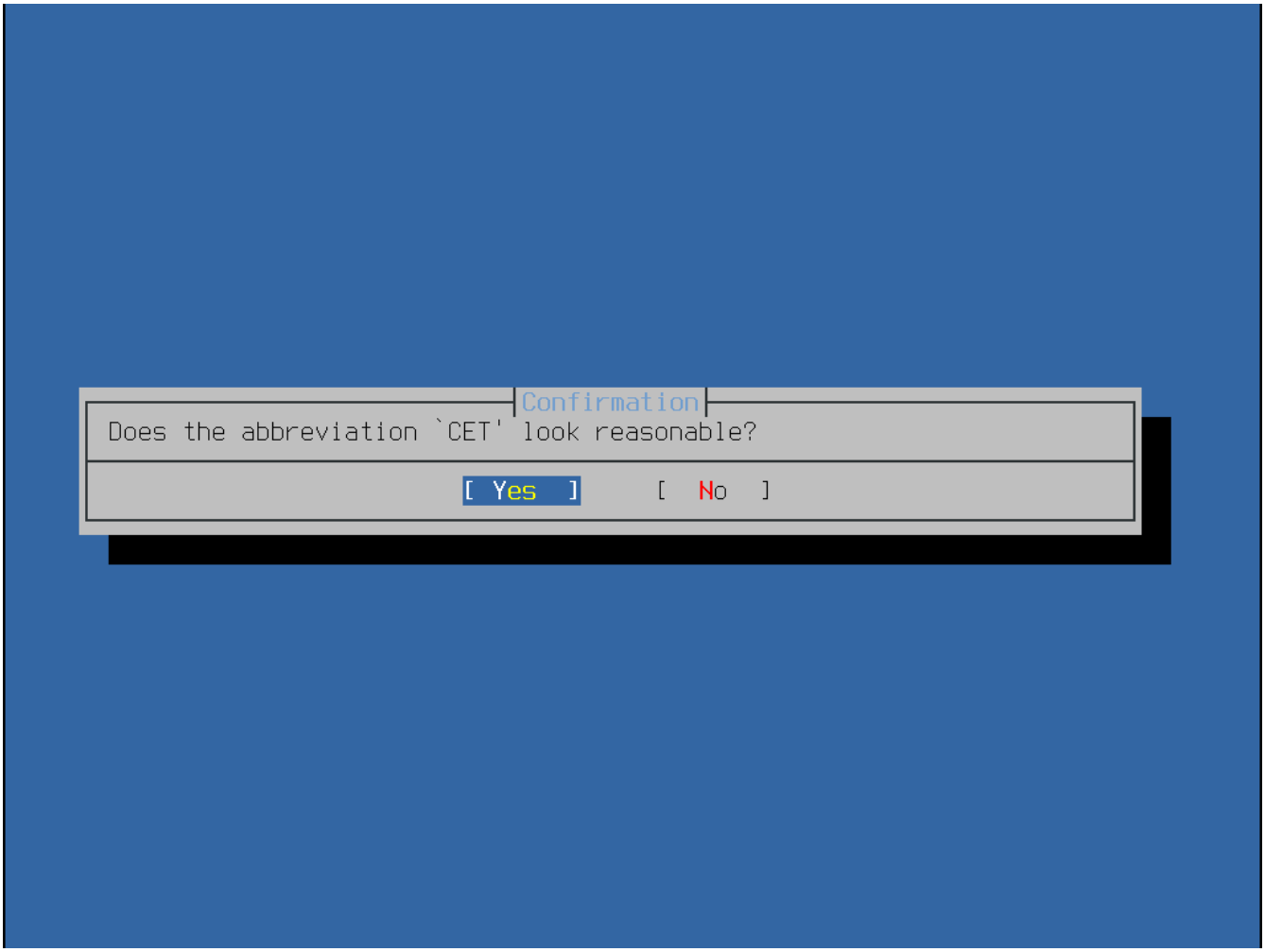

*Зураг 91. Улсыг сонгох*

Сум ашиглан зохих улсыг сонгоод `Enter` дарна.




*Зураг 92. Цагийн бүсийг сонгох*

Сум ашиглан зохих цагийн бүсийг сонгоод `Enter` дарна.

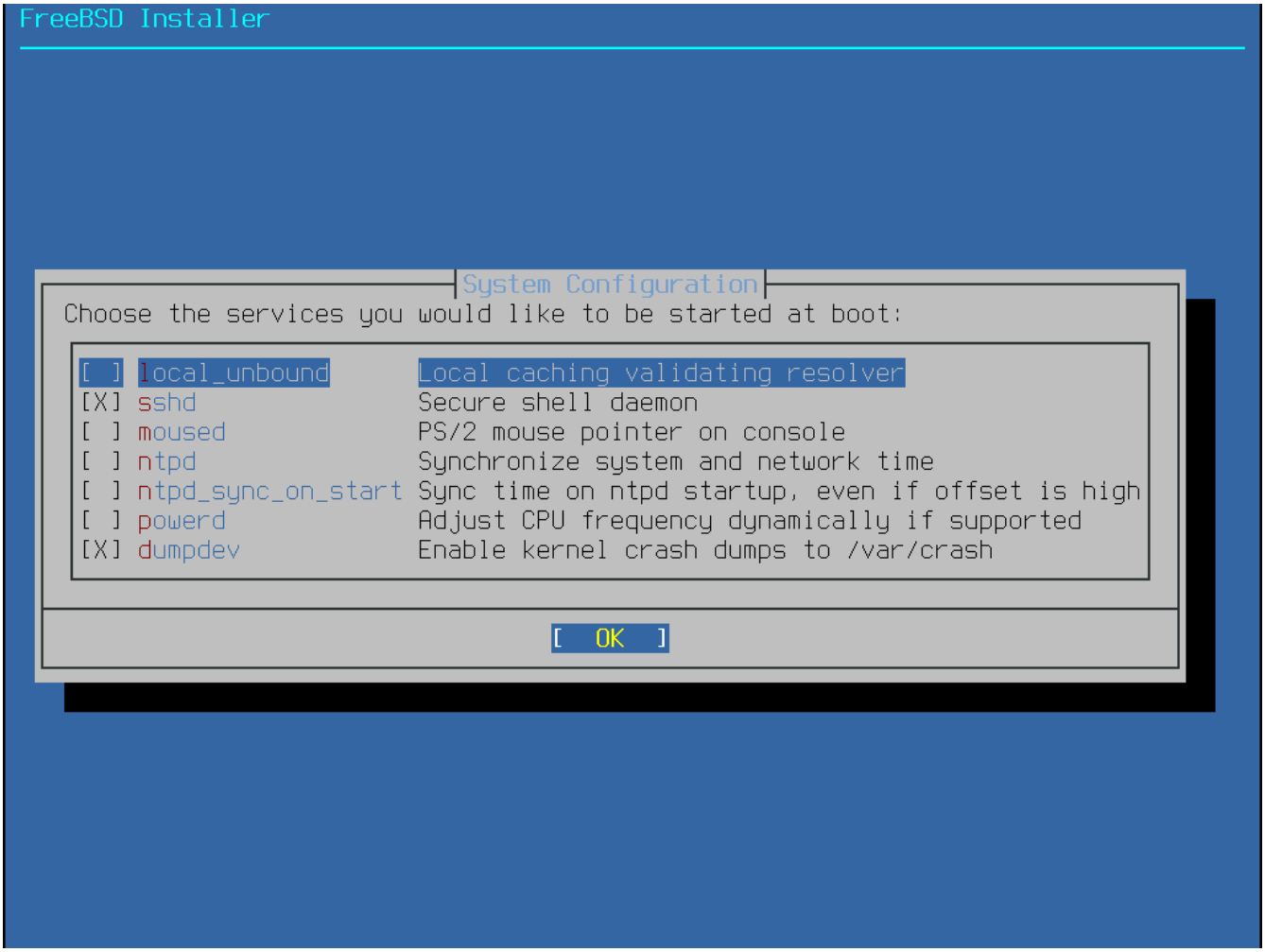



*Зураг 93. Цагийн бүсийг зөвшөөрөх*

Цагийн бүсийн товчлол зөв эсэхийг шалгаарай. Зөв байвал `Enter` дарж суулгалтын дараах тохиргоог үргэлжлүүлнэ.

### 3.9.4. Үйлчилгээ идэвхжүүлэхийг сонгох

Ачаалах үед ажиллах нэмэлт үйлчилгээнүүдийг идэвхжүүлж болно. Эдгээр үйлчилгээнүүд нь бүгд нэмэлтээр байдаг.



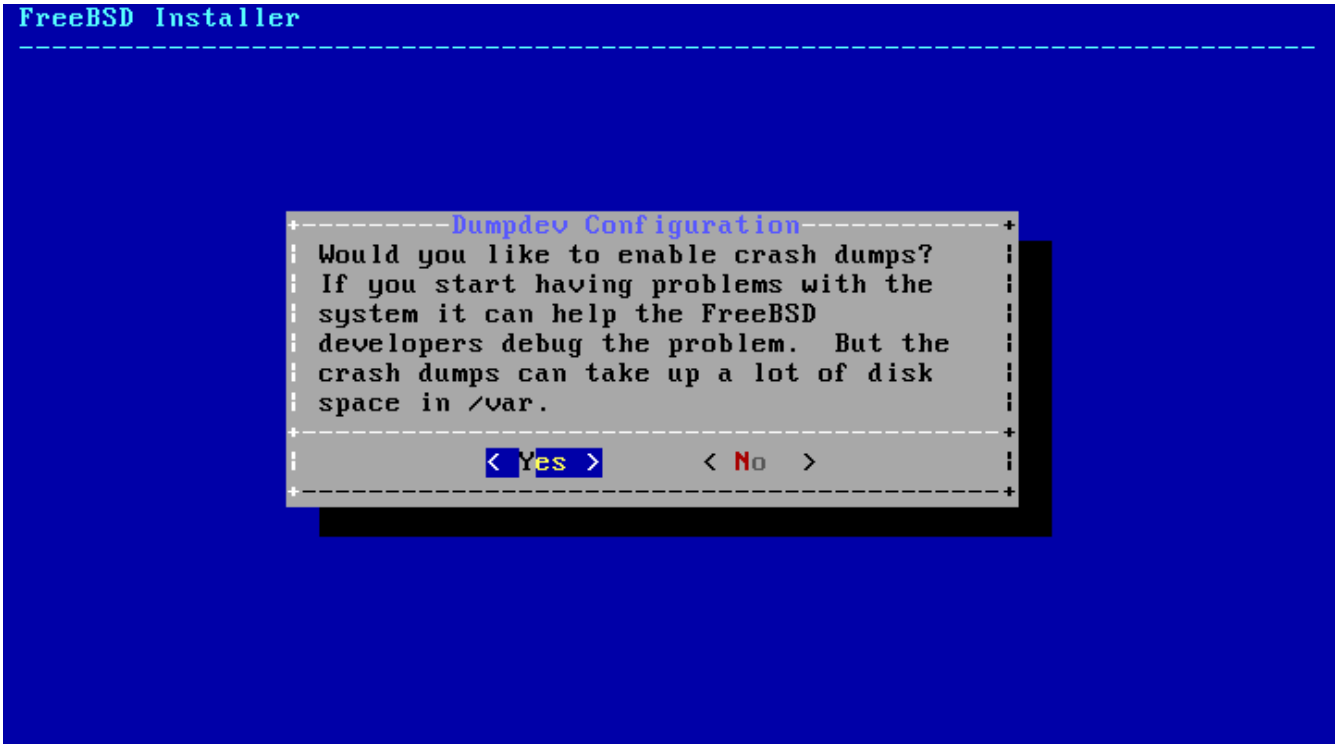

*Зураг 94. Нэмэлт үйлчилгээнүүдийг идэвхжүүлэхийг сонгох*

*Нэмэлт үйлчилгээнүүд*

- `sshd` - Аюулгүй алсын холболтод зориулсан аюулгүй бүрхүүл (SSH) демон.
- `moused` - Системийн консолийн хүрээнд хулганы хэрэглээг бий болгодог.
- `ntpd` - Цагийг автоматаар синхрончлох Сүлжээний Цагийн Протокол (NTP) демон.
- `powerd` - Тэжээлийн хяналт болон энерги зүй зохистой хэрэглэхэд зориулсан системийн тэжээлийн хяналт.

### 3.9.5. Сүйрлийн үеийн мэдээллийг хадгалахыг идэвхжүүлэх

bsdinstall нь систем дээр сүйрлийн үеийн мэдээллийг хадгалахыг идэвхжүүлэх эсэхийг асуудаг. Сүйрлийн үеийн мэдээлэл хадгалахыг идэвхжүүлэх нь систем дээр асуудлыг дибаг хийхэд маш хэрэгтэй байдаг бөгөөд хэрэглэгчдэд аль болох энэ боломжийг идэвхжүүлэхийг зөвлөдөг. **[ Yes ]** сонгож сүйрлийн үеийн мэдээллийг хадгалахыг идэвхжүүлэх юм уу эсвэл **[ No ]** дарж сүйрлийн үеийн мэдээллийг хадгалахыг идэвхжүүлэлгүйгээр үргэлжлүүлж болно.

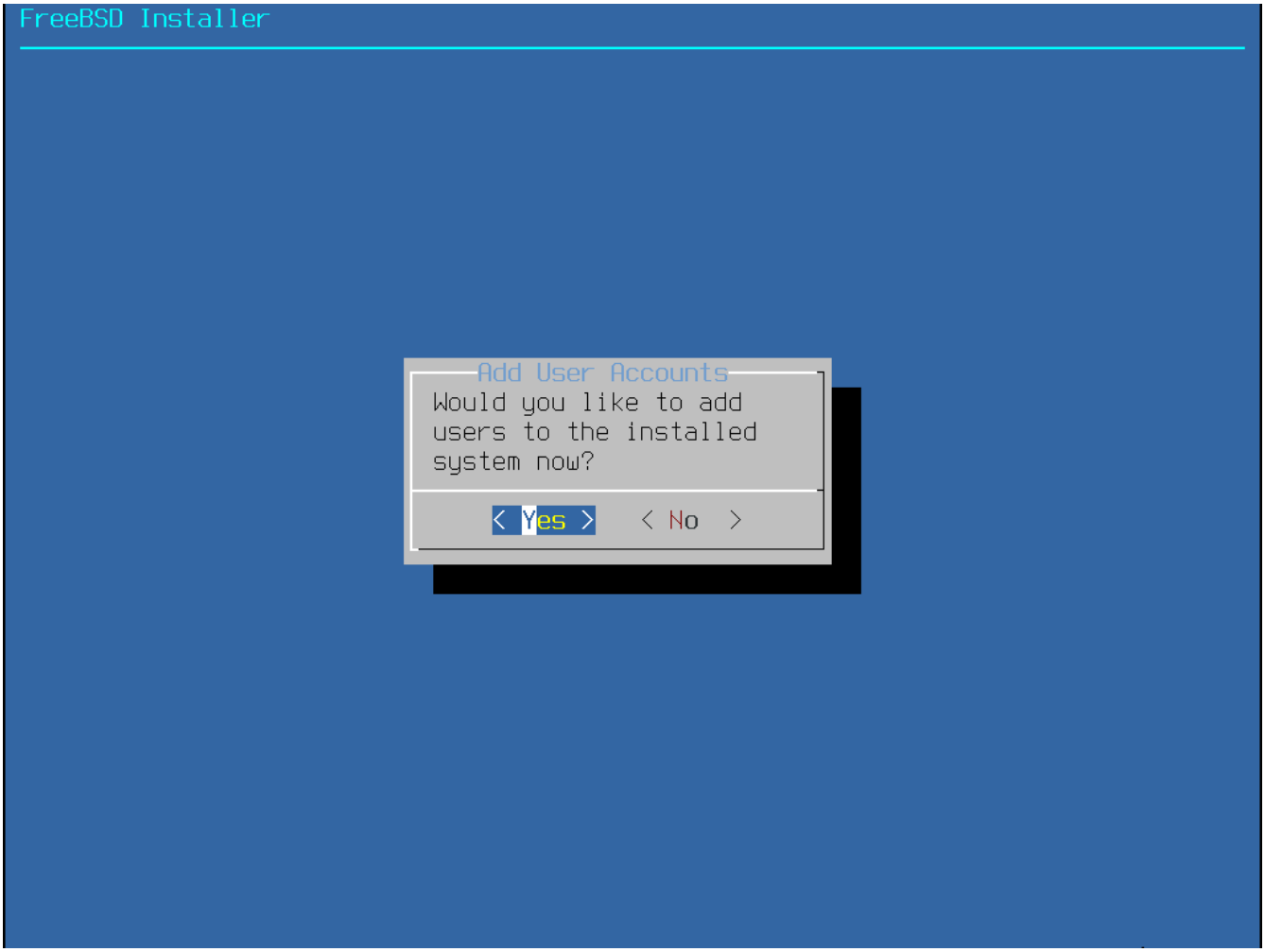



*Зураг 95. Сүйрлийн үеийн мэдээллийг хадгалахыг идэвхжүүлэх*

### 3.9.6. Хэрэглэгчид нэмэх

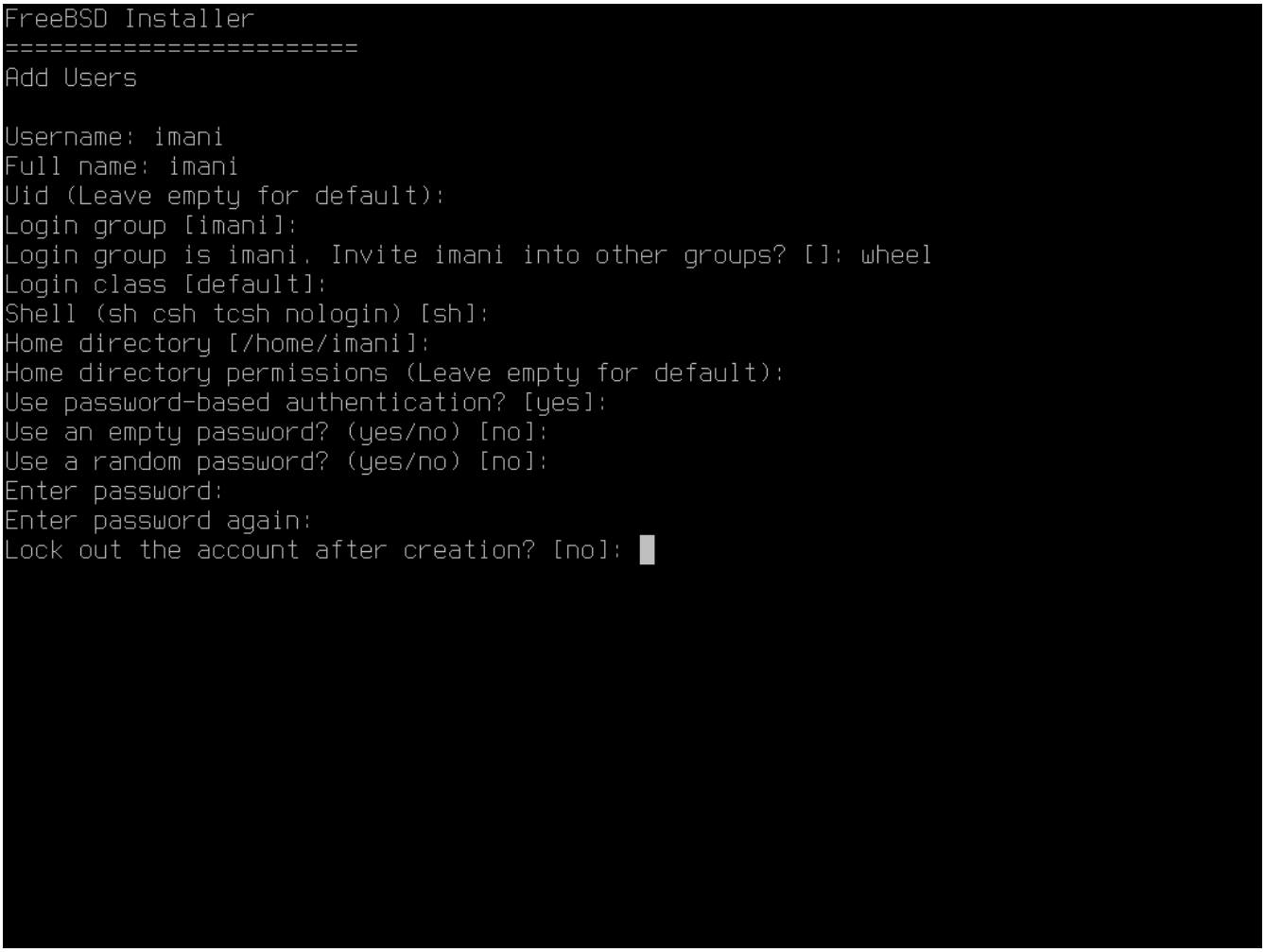
Суулгалтын явцад ядаж нэг хэрэглэгч нэмэх нь `root` хэрэглэгчээр системд оролгүйгээр системийг ашиглах боломжийг олгодог. `root` хэрэглэгчээр системд орвол хийх үйлдлийн хувьд ямар нэг хязгаарлалт аль эсвэл хамгаалалт байдаггүй. Нормал хэрэглэгчээр нэвтрэн орох нь илүү аюулгүй бөгөөд нууцлаг юм.

**[ Yes ]** дарж шинэ хэрэглэгч нэмнэ.




*Зураг 96. Хэрэглэгчийн бүртгэл нэмэх*

Нэмэх хэрэглэгчийн мэдээллийг оруулна.




*Зураг 97. Хэрэглэгчийн мэдээлэл оруулах*

*Хэрэглэгчийн мэдээлэл*

- `Username` - Хэрэглэгчийн нэвтрэх нэр. Ихэвчлэн нэрний эхний үсгийг овогтой нь цуг ашигласан байдаг.
- `Full name` - Хэрэглэгчийн бүтэн нэр.
- `Uid` - Хэрэглэгчийн ID. Систем үүнийг зааж өгөх тул ихэвчлэн хоосон үлдээдэг.
- `Login group` - Хэрэглэгчийн бүлэг. Анхдагч уттыг авахаар хоосон үлдээдэг.
- `Invite user into other groups?` - Хэрэглэгчийг гишүүнээр нэмж болох нэмэлт бүлгүүд.
- `Login class` - Анхдагч уттыг авахаар хоосон үлдээдэг.
- `Shell` - Хэрэглэгчид зориулсан интерактив бүрхүүл. Жишээн дээр csh(1) сонгосон байгаа.
- `Home directory` - Хэрэглэгчийн гэр сан. Анхдагч утга ихэвчлэн зөв байдаг.
- `Home directory permissions` - Хэрэглэгчийн гэрийн сангийн зөвшөөрлүүд. Анхдагч утга ихэвчлэн зөв байдаг.
- `Use password-based authentication?` - Ихэвчлэн "yes".
- `Use an empty password?` - Ихэвчлэн "no".
- `Use a random password?` - Ихэвчлэн "no".
- `Enter password` - Хэрэглэгчийн жинхэнэ нууц үг. Бичсэн тэмдэгтүүд дэлгэц дээр гарахгүй.
- `Enter password again` - Шалгахын тулд нууц үгийг дахин бичих ёстой.

- `Lock out the account after creation?` - Ихэвчлэн "no".

Бүгдийг оруулсны дараа ерөнхий дүгнэсэн мэдээлэл гарч ирэх бөгөөд систем зөв эсэхийг асууна. Хэрэв оруулж байх явцдаа алдаа хийсэн бол `no` гэж бичин дахин оролдоорой. Хэрэв бүгд зөв бол `yes` гэж бичин шинэ хэрэглэгч үүсгэнэ.

```
FreeBSD Installer
=======================
Add Users

Username: imani
Full name: imani
Uid (Leave empty for default):
Login group [imani]:
Login group is imani. Invite imani into other groups? []: wheel
Login class [default]:
Shell (sh csh tcsh nologin) [sh]:
Home directory [/home/imani]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]:
Use an empty password? (yes/no) [no]:
Use a random password? (yes/no) [no]:
Enter password:
Enter password again:
Lock out the account after creation? [no]:
Username   : imani
Password   : *****
Full Name  : imani
Uid        : 1001
Class      :
Groups     : imani wheel
Home       : /home/imani
Home Mode  :
Shell      : /bin/sh
Locked     : no
OK? (yes/no) [yes]:
adduser: INFO: Successfully added (imani) to the user database.
Add another user? (yes/no) [no]:
```

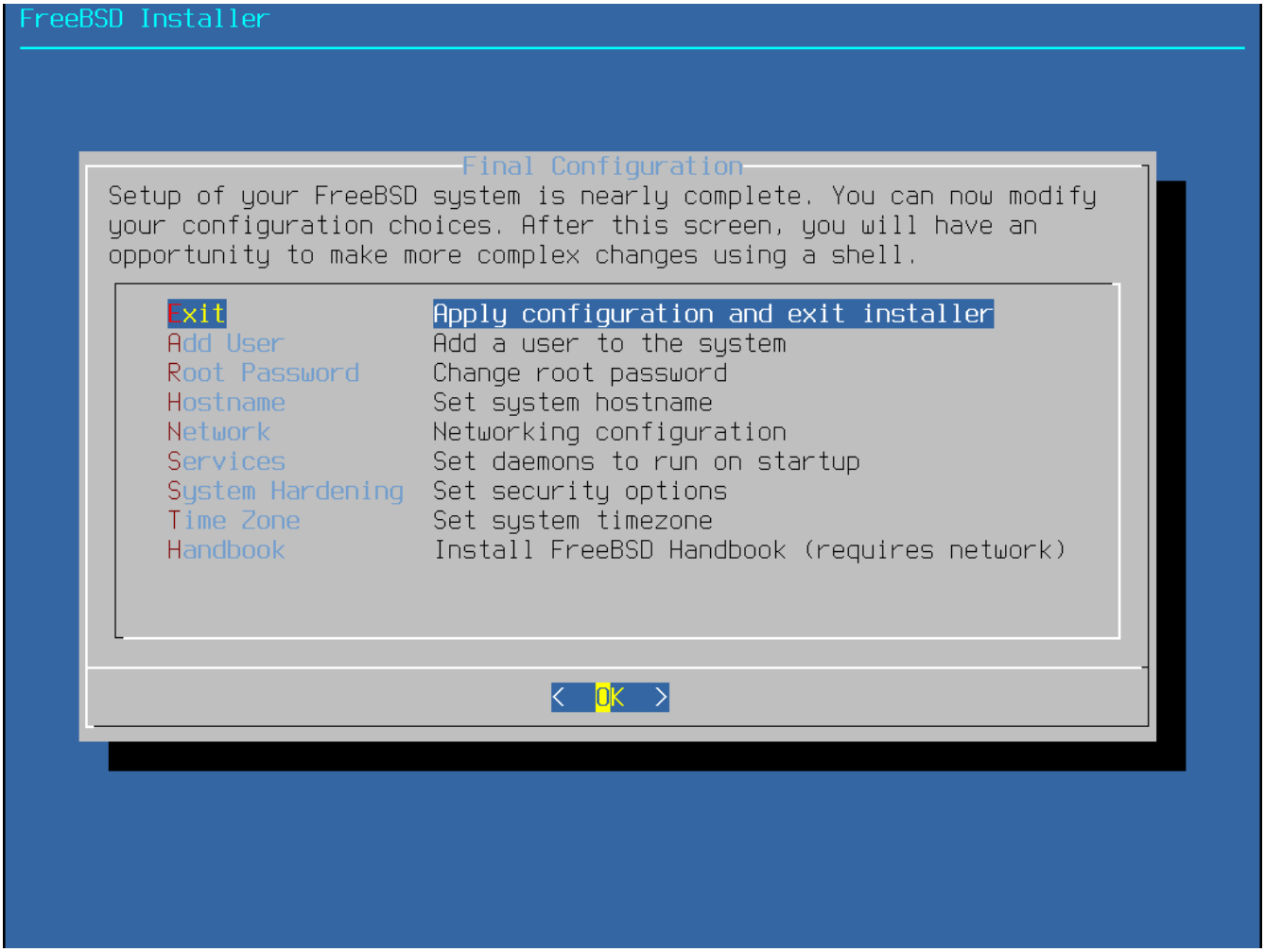
*Зураг 98. Хэрэглэгч ба бүлгийн удирдлагаас гарах*

Хэрэв өөр хэрэглэгч нэмэх шаардлагатай бол "Add another user?" гэсэн асуултад `yes` гэж хариулаарай. `no` гэж бичин хэрэглэгч нэмэхээ зогсоож суулгалтыг үргэлжлүүлнэ.

Хэрэглэгч нэмэх болон хэрэглэгчийн удирдлагын талаар илүү дэлгэрэнгүйг Хэрэглэгчид ба үндсэн бүртгэл зохицуулалт хэсгээс үзнэ үү.

### 3.9.7. Төгсгөлийн тохиргоо

Бүгдийг суулгаж тохируулсны дараа тохиргоонуудыг өөрчлөх сүүлчийн боломжийг систем олгоно.



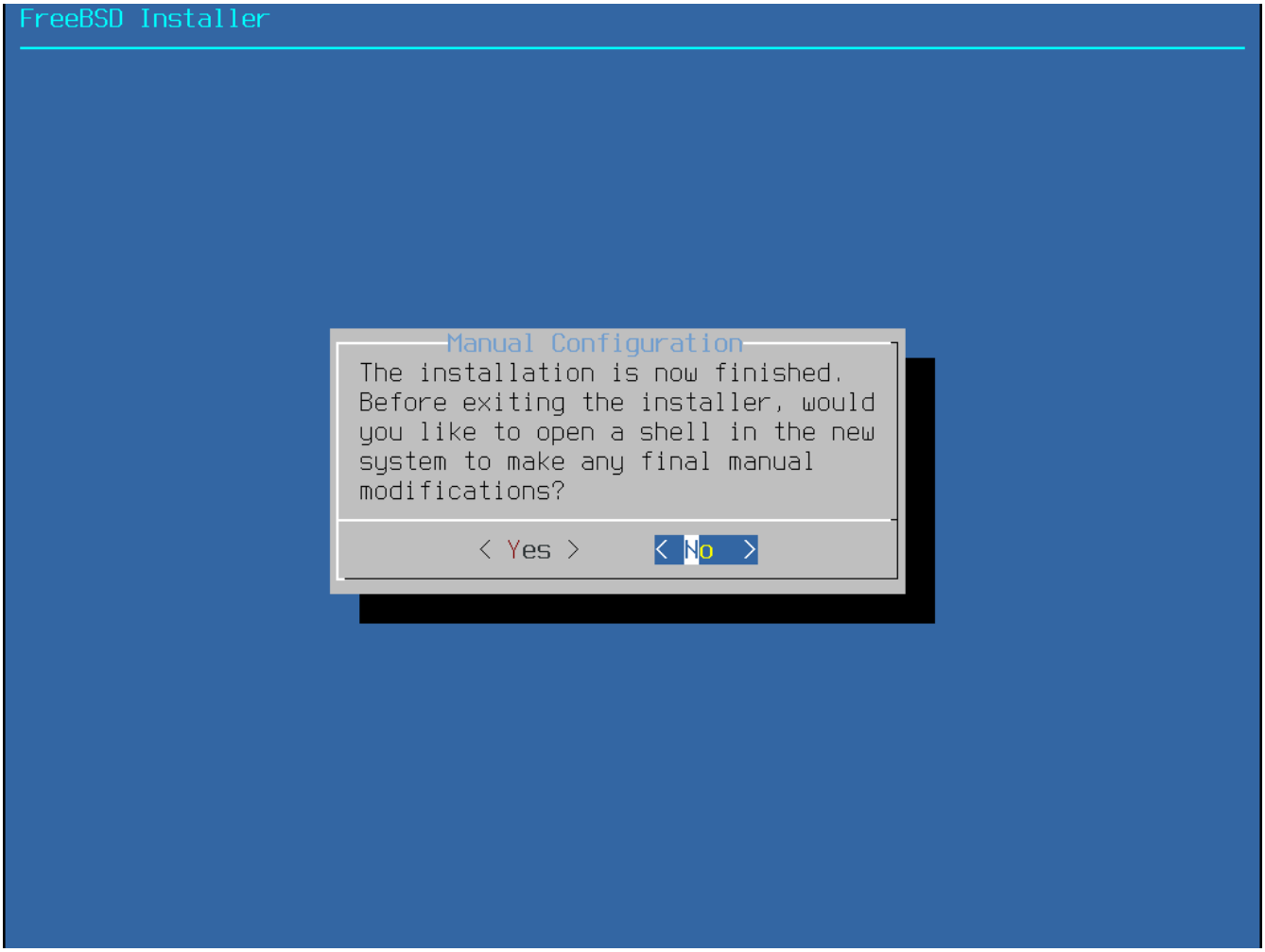

*Зураг 99. Төгсгөлийн тохиргоо*

Суулгалтыг гүйцээхээсээ өмнө энэ цэсийг ашиглан ямар нэг өөрчлөлт аль эсвэл нэмэлт тохиргоог хийгээрэй.

*Төгсгөлийн тохиргооны сонголтууд*

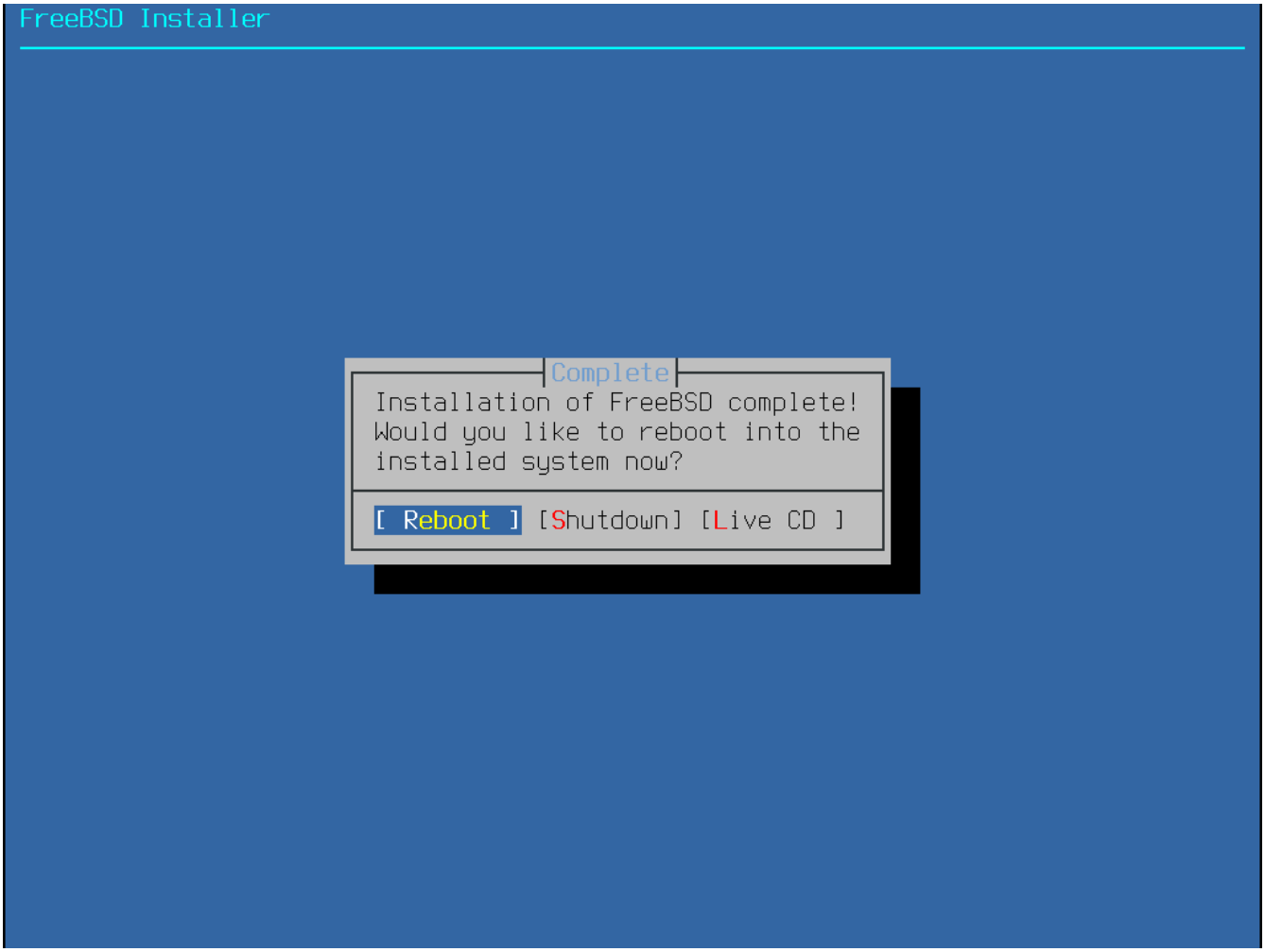
- `Add User` - Хэрэглэгчид нэмэх хэсэгт тайлбарласан байгаа.
- `Root Password` - `root` нууц үгийг тохируулах хэсэгт тайлбарласан байгаа.
- `Hostname` - Хостын нэрийг тохируулах нь хэсэгт тайлбарласан байгаа.
- `Network` - Сүлжээний интерфэйсүүдийг тохируулах нь хэсэгт тайлбарласан байгаа.
- `Services` - Үйлчилгээ идэвхжүүлэхийг сонгох хэсэгт тайлбарласан байгаа.
- `Time Zone` - Цагийн бүсийг тохируулах нь хэсэгт тайлбарласан байгаа.
- `Handbook` - FreeBSD гарын авлага (таны одоо уншиж байгаа энэ заавар) татаж суулгана.

Төгсгөлийн ямар нэг тохиргоо хийж дууссаны дараа **[ Exit ]**-г сонгож суулгалтаас гарна.




*Зураг 100. Гараар тохируулах*

bsdinstall нь шинэ систем рүү ачаалахаас өмнө өөр илүү нэмэлт тохиргоо шаардлагатай эсэхийг асууна. Шинэ системээс **[ Yes ]**-г сонгон дарж бүрхүүл рүү орох юм уу эсвэл **[ No ]**-г дарж суулгалтын сүүлийн алхам руу орно.




*Зураг 101. Суулгалтыг гүйцээх*

Хэрэв илүү тохиргоо эсвэл тусгай тохиргоо хэрэгтэй бол **[ Live CD ]**-г сонгосноор суулгалтын зөөвөрчлөгчийг Live CD горим руу ачаалах болно.

Суулгалт дууссаны дараа **[ Reboot ]**-г сонгож компьютерийг дахин ачаалж шинэ FreeBSD системийг эхлүүлнэ. FreeBSD суулгалтын CD, DVD эсвэл USB санах ойн дискийг авахаа мартуузай, тэгэхгүй бол компьютер тэр байгаа зөөвөрлөгчөөс дахин ачаалах болно.

## 3.9.8. FreeBSD ачаалах ба унтраах

### 3.9.8.1. FreeBSD/i386 ачаалах

FreeBSD ачаалах үед олон мэдээллийн чанартай зурвасууд дэлгэц дээр гардаг. Ихэнх нь дэлгэцээр урсан өнгөрөх бөгөөд энэ нь энгийн явдал юм. Систем ачаалж дууссаны дараа нэвтрэх хэсэг гарч ирдэг. Дэлгэц дээр гүйж өнгөрсөн зурвасуудыг үзэхийн тулд `Scroll-Lock` дарж *scroll-back buffer*-г идэвхжүүлнэ. `PgUp`, `PgDn` болон сумнуудыг ашиглан зурвасуудыг гүйлгэн харж болно. `Scroll-Lock` дарж дэлгэцийн түгжсэн горимоос гарч ердийн дэлгэц рүү эргэн орно.

`login:` цонхон дээр суулгалтын үед нэмсэн хэрэглэгчийн нэрийг оруулна. Жишээн дээр `asample` гэж буй. Шаардлагагүй тохиолдолд `root` хэрэглэгчээр орохоос зайлсхийгээрэй.

Дээр дурдсан зурвасуудыг гүйлгэж харах боломж нь хязгаарлагдмал бөгөөд бүх зурвасуудыг үзэх боложмгүй юм. Нэвтэрч орсны дараа тушаалын мөрөөс ихэнхийг нь үзэж болох бөгөөд

ингэхийн тулд `dmesg | less` тушаалуудыг ашиглана. Үзэж дууссаныхаа дараа `q` дарж тушаалын мөр рүү буцаж орно.

Ачаалах үеийн зурвасууд (хувилбарын мэдээллийг орхисон байгаа):

```
Copyright (c) 1992-2011 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.

    root@farrell.cse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC amd64
CPU: Intel(R) Core(TM)2 Duo CPU     E8400  @ 3.00GHz (3007.77-MHz K8-class CPU)
  Origin = "GenuineIntel"  Id = 0x10676  Family = 6  Model = 17  Stepping = 6
  Features
=0x783fbff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,APIC,SEP,MTRR,PGE,MCA,CMOV,PAT,PSE36,MMX
,FXSR,SSE,SSE2>
  Features2=0x209<SSE3,MON,SSSE3>
  AMD Features=0x20100800<SYSCALL,NX,LM>
  AMD Features2=0x1<LAHF>
real memory  = 536805376 (511 MB)
avail memory = 491819008 (469 MB)
Event timer "LAPIC" quality 400
ACPI APIC Table: <VBOX   VBOXAPIC>
ioapic0: Changing APIC ID to 1
ioapic0 <Version 1.1> irqs 0-23 on motherboard
kbd1 at kbdmux0
acpi0: <VBOX VBOXXSDT> on motherboard
acpi0: Power Button (fixed)
acpi0: Sleep Button (fixed)
Timecounter "ACPI-fast" frequency 3579545 Hz quality 900
acpi_timer0: <32-bit timer at 3.579545MHz> port 0x4008-0x400b on acpi0
cpu0: <ACPI CPU> on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci0: <ACPI PCI bus> on pcib0
isab0: <PCI-ISA bridge> at device 1.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <Intel PIIX4 UDMA33 controller> port 0x1f0-0x1f7,0x3f6,0x170-
0x177,0x376,0xd000-0xd00f at device 1.1 on pci0
ata0: <ATA channel 0> on atapci0
ata1: <ATA channel 1> on atapci0
vgapci0: <VGA-compatible display> mem 0xe0000000-0xe0ffffff irq 18 at device 2.0 on
pci0
em0: <Intel(R) PRO/1000 Legacy Network Connection 1.0.3> port 0xd010-0xd017 mem
0xf0000000-0xf001ffff irq 19 at device 3.0 on pci0
em0: Ethernet address: 08:00:27:9f:e0:92
pci0: <base peripheral> at device 4.0 (no driver attached)
pcm0: <Intel ICH (82801AA)> port 0xd100-0xd1ff,0xd200-0xd23f irq 21 at device 5.0 on
pci0
pcm0: <SigmaTel STAC9700/83/84 AC97 Codec>
ohci0: <OHCI (generic) USB controller> mem 0xf0804000-0xf0804fff irq 22 at device 6.0
```

```
on pci0
usbus0: <OHCI (generic) USB controller> on ohci0
pci0: <bridge> at device 7.0 (no driver attached)
acpi_acad0: <AC Adapter> on acpi0
atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0
atkbd0: <AT Keyboard> irq 1 on atkbdc0
kbd0 at atkbd0
atkbd0: [GIANT-LOCKED]
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: model IntelliMouse Explorer, device ID 4
attimer0: <AT timer> port 0x40-0x43,0x50-0x53 on acpi0
Timecounter "i8254" frequency 1193182 Hz quality 0
Event timer "i8254" frequency 1193182 Hz quality 100
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
atrtc0: <AT realtime clock> at port 0x70 irq 8 on isa0
Event timer "RTC" frequency 32768 Hz quality 0
ppc0: cannot reserve I/O port range
Timecounters tick every 10.000 msec
pcm0: measured ac97 link rate at 485193 Hz
em0: link state changed to UP
usbus0: 12Mbps Full Speed USB v1.0
ugen0.1: <Apple> at usbus0
uhub0: <Apple OHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0
cd0 at ata1 bus 0 scbus1 target 0 lun 0
cd0: <VBOX CD-ROM 1.0> Removable CD-ROM SCSI-0 device
cd0: 33.300MB/s transfers (UDMA2, ATAPI 12bytes, PIO 65534bytes)
cd0: Attempt to query device size failed: NOT READY, Medium not present
ada0 at ata0 bus 0 scbus0 target 0 lun 0
ada0: <VBOX HARDDISK 1.0> ATA-6 device
ada0: 33.300MB/s transfers (UDMA2, PIO 65536bytes)
ada0: 12546MB (25694208 512 byte sectors: 16H 63S/T 16383C)
ada0: Previously was known as ad0
Timecounter "TSC" frequency 3007772192 Hz quality 800
Root mount waiting for: usbus0
uhub0: 8 ports with 8 removable, self powered
Trying to mount root from ufs:/dev/ada0p2 [rw]...
Setting hostuuid: 1848d7bf-e6a4-4ed4-b782-bd3f1685d551.
Setting hostid: 0xa03479b2.
Entropy harvesting: interrupts ethernet point_to_point kickstart.
Starting file system checks:
/dev/ada0p2: FILE SYSTEM CLEAN; SKIPPING CHECKS
/dev/ada0p2: clean, 2620402 free (714 frags, 327461 blocks, 0.0% fragmentation)
Mounting local file systems:.
vboxguest0 port 0xd020-0xd03f mem 0xf0400000-0xf07fffff,0xf0800000-0xf0803fff irq 20
at device 4.0 on pci0
vboxguest: loaded successfully
Setting hostname: machine3.example.com.
Starting Network: lo0 em0.
```

```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
        options=3<RXCSUM,TXCSUM>
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3
        inet 127.0.0.1 netmask 0xff000000
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=9b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM>
        ether 08:00:27:9f:e0:92
        nd6 options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL>
        media: Ethernet autoselect (1000baseT <full-duplex>)
        status: active
Starting devd.
Starting Network: usbus0.
DHCPREQUEST on em0 to 255.255.255.255 port 67
DHCPACK from 10.0.2.2
bound to 192.168.1.142 -- renewal in 43200 seconds.
add net ::ffff:0.0.0.0: gateway ::1
add net ::0.0.0.0: gateway ::1
add net fe80::: gateway ::1
add net ff02::: gateway ::1
ELF ldconfig path: /lib /usr/lib /usr/lib/compat /usr/local/lib
32-bit compatibility ldconfig path: /usr/lib32
Creating and/or trimming log files.
Starting syslogd.
No core dumps found.
Clearing /tmp (X related).
Updating motd:.
Configuring syscons: blanktime.
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
10:a0:f5:af:93:ae:a3:1a:b2:bb:3c:35:d9:5a:b3:f3 root@machine3.example.com
The key's randomart image is:
+--[RSA1 1024]----+
|    o..          |
|   o . .         |
|  .   o          |
|       o         |
|    o   S        |
|   + + o         |
|o . + *          |
|o+ ..+ .         |
|==o..o+E         |
+-----------------+
Generating public/private dsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
7e:1c:ce:dc:8a:3a:18:13:5b:34:b5:cf:d9:d1:47:b2 root@machine3.example.com
```

```
The key's randomart image is:
+--[ DSA 1024]----+
|       ..     . .|
|      o  .   . + |
|     . ..   . E .|
|    . .  o o . . |
|     +  S = .    |
|    +  . = o     |
|     +  . * .    |
|    . .  o .     |
|      .o. .      |
+-----------------+
Starting sshd.
Starting cron.
Starting background file system checks in 60 seconds.

Thu Oct  6 19:15:31 MDT 2011

FreeBSD/amd64 (machine3.example.com) (ttyv0)

login:
```

RSA ба DSA түлхүүрүүдийг үүсгэх нь удаан машин дээр хугацаа зарцуулж магадгүй. Энэ нь sshd-г автоматаар эхлүүлэхээр тохируулсан тохиолдолд зөвхөн суулгасны дараа эхний удаа ачаалах явцад хийгддэг. Дараа дараагийн ачаалах явц хурдан байх болно.

FreeBSD нь анхдагчаар график орчин суулгадаггүй боловч суулгах боломжтой олон орчин байдаг. Дэлгэрэнгүй мэдээллийг X Цонхот систем хэсгээс үзнэ үү.

### 3.9.9. FreeBSD унтраах

FreeBSD компьютерийг зөв унтраах нь өгөгдөл болон бүр төхөөрөмжийг эвдрэлээс хамгаалахад тусалж болно. Тэжээлийг унтрааж болохгүй. Хэрэв хэрэглэгч `wheel` бүлгийн гишүүн бол тушаалын мөрөн дээр `su` гэж бичин `root` нууц үгийг оруулж супер хэрэглэгч болно. Үгүй бол `root` хэрэглэгчээр орж `shutdown -p now` гэж ажиллуулаарай. Систем цэвэрхэн хаагдаж өөрийгөө унтраах болно.

`Ctrl` + `Alt` + `Del` товчлуурын хослол системийг дахин ачаалахад ашиглагдах боловч ердийн ажиллагааны үед үүнийг зөвлөдөггүй.

## 3.10. Асуудлыг олж засварлах

Дараах хэсэг суулгалттай холбоотой хүмүүсийн тайлагнасан зарим нэг түгээмэл энгийн асуудлуудыг олж шийдвэрлэх талаар тайлбарлана.

### 3.10.1. Хэрэв ямар нэг юм буруу бол яах вэ

PC архитектурын төрөл бүрийн хязгаарлалтуудаас болоод төхөөрөмжүүдийг олж илрүүлэх нь 100% найдвартай биш байдаг боловч хэрэв тэгж чадахгүй байгаа бол та цөөн хэдэн зүйл

хийж болно.

Таны тоног төхөөрөмж дэмжигдсэн эсэхийг шалгахын тулд өөрийн FreeBSD хувилбарын Тоног төхөөрөмжийн тэмдэглэл баримтыг уншаарай.

Хэрэв таны тоног төхөөрөмж дэмжигдсэн боловч та гацах эсвэл бусад асуудлуудтай учраад байгаа бол өөрчлөн тохируулсан цөм бүтээх шаардлагатай болно. Энэ нь танд GENERIC цөмд байхгүй төхөөрөмжүүдийн дэмжлэгийг нэмэх боломжийг олгоно. Ачаалах дискнүүд дээрх цөм нь ихэнх төхөөрөмжүүдийг IRQ, IO хаягууд, DMA сувгуудын хувьд үйлдвэрийн анхдагч тохиргоотой байна гэж тооцон тохиргоо хийгдсэн байдаг. Хэрэв таны тоног төхөөрөмжид дахин тохиргоо хийгдсэн бол FreeBSD тэдгээрийг олохын тулд та цөмийн тохиргоогоо засаж дахин бүтээх хэрэгтэй болно.

Байхгүй төхөөрөмжийг хайж олохгүй байгаа нь дараа нь өөр байгаа төхөөрөмжийг бас олохгүйд хүргэж болох юм. Ийм тохиолдолд асуудал үүсгээд байгаа драйверуудыг хайхыг болиулах хэрэгтэй.

Төрөл бүрийн тоног төхөөрөмжийн бүрэлдэхүүнүүд ялангуяа эх хавтан дээр үйлдвэрээс сууж ирсэн програмуудыг шинэчилснээр суулгалтын үеийн зарим асуудлуудыг шийдэж болох юм. Эх хавтангийн програмыг ихэвчлэн BIOS гэдэг. Ихэнх эх хавтан болон компьютер үйлдвэрлэгчид шинэчлэлт болон шинэчлэх мэдээлэлд зориулсан вэбтэй байдаг.

Чухал шинэчлэлт зэрэг заавал хийх шинэчлэлт хийхээс бусдаар үйлдвэрлэгчид эх хавтангийн BIOS-г шинэчлэхийг зөвлөдөггүй. Шинэчлэх процесс BIOS-г бүрэн биш болгож компьютерийг ажиллагаагүй байдалд хүргэн буруу ажиллаж *болно*.

## 3.10.2. Алдааг олж шийдвэрлэх асуултууд ба хариултууд

### 3.10.2.1. Миний систем ачаалах явцдаа төхөөрөмж шалгаж байхдаа гацсан эсвэл суулгах явцад сонин байсан.

FreeBSD нь системийн тохиргоонд туслахын тулд i386, amd64, ба ia64 тавцангууд дээр системийн ACPI үйлчилгээг хэрэв илэрсэн бол ихээхэн ашигладаг. Харамсалтай нь ACPI драйвер болон системийн эх хавтан дотор, BIOS-н програм дээр зарим нэг алдаа байсаар байдаг. ACPI-г ачаалагч дуудагчийн гурав дахь шатан дээр `hint.acpi.0.disabled` хувьсагчийг зааж өгөн хааж өгч болно:

```
set hint.acpi.0.disabled="1"
```

Систем ачаалах болгонд энэ нь өөрчлөгдөх учир /boot/loader.conf файлд `hint.acpi.0.disabled="1"`-г нэмж өгөх хэрэгтэй. Ачаалагч дуудагчийн талаар илүү дэлгэрэнгүйг Ерөнхий агуулга хэсгээс үзэж болно.

# 3.11. Амьд CD ашиглах нь

FreeBSD-ийн амьд CD үндсэн суулгац програм байгаа CD дээр байдаг. Энэ нь хэрэглэгчдийн хувьд FreeBSD нь тэдний хүссэн үйлдлийн систем эсэхийг мэдэх болон зарим нэг боломжуудыг суулгахаасаа өмнө туршихад хэрэгтэй юм.

Амьд CD ашиглахаасаа өмнө дараах зүйлсийг анхаарах хэрэгтэй:

- Систем рүү хандахын тулд нэвтрэх эрх шаардлагатай. Хэрэглэгчийн нэр нь `root` ба нууц үг нь хоосон байна.
- Систем нь CD-с шууд ажиллах учир хатуу диск дээр суулгасан системээс хамаагүй удаан байна.
- Амьд CD нь тушаал хүлээх мөртэй бөгөөд график интерфэйсгүй байна.

# Chapter 4. Юниксийн үндэс

## 4.1. Ерөнхий агуулга

Дараах бүлгүүд нь FreeBSD үйлдлийн системийн үндсэн тушаал мөн гүйцэтгэлүүдийг хамрах болно. Энэ материалын ихэнх хэсэг нь UNIX®-төрлийн үйлдлийн системтэй хамаатай. Хэрэв та юникс төрлийн үйлдлийн системтэй танил бол энэ бүлгийг алгасаж болно. Харин та FreeBSD-г шинээр сурагч бол энэ бүлгийг тун анхааралтай унших хэрэгтэй.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD-н "виртуал консолыг" хэрхэн хэрэглэх.
- FreeBSD дээрх файлын тугийг ойлгосноор UNIX®-н файлын зөвшөөрөл хэрхэн ажилладгийг мэдэх.
- Үндсэн FreeBSD файл системийн байрлуулалтыг мэдэх.
- FreeBSD диск зохион байгуулалт.
- Файл системийг хэрхэн холбох болон салгах.
- Процесс, далд чөтгөр дэмон, мөн дохионууд гэж юу болох.
- Бүрхүүл гэж юу болох, мөн хэрхэн анхны нэвтрэх орчноо өөрчлөх.
- Үндсэн текст засварлагчийг хэрхэн хэрэглэх.
- Төхөөрөмж болох төхөөрөмжийн цэг гэж юу болох.
- FreeBSD-д ямар хоёртын хэлбэр хэрэглэгддэг болох.
- Нэмэлт мэдээллийг товч гарын авлагаас хэрхэн харах.

## 4.2. Виртуал консол болон терминал

FreeBSD нь олон аргаар хэрэглэгдэж чадна. Тэдний нэг нь тушаалуудыг текст терминал дээр бичиж хэрэглэх юм. Хэрэв та FreeBSD-г ингэж хэрэглэвэл, UNIX®-н маш олон боломж болон хүч таны гарт байна гэсэн үг. Энэ бүлэг танд, "терминал" болон "консол" гэж юу болох, мөн тэд нарыг FreeBSD дээр хэрхэн хэрэглэх талаар танд тодорхойлж өгөх болно.

### 4.2.1. Консол

Хэрэв та FreeBSD-г эхлэх үедээ автоматаар график горимд эхлүүлэхээр тохируулаагүй бол, систем нь эхлэлтийн гүйцэтгэдэг бичиглэлүүдээ дуусгасны дараа танд нэвтрэх хуудас харуулдаг. Та нэг иймэрхүү хуудас харах байх:

```
Additional ABI support:.
Local package initialization:.
Additional TCP options:.

Fri Sep 20 13:01:06 EEST 2002
```

```
FreeBSD/i386 (pc3.example.org) (ttyv0)

login:
```

Энд харуулсан текстүүд нь таны систем дээр жаахан өөр байж магадгүй ч, ерөнхийдөө үүнтэй ижил байх ёстой. Сүүлийн хоёр мөр нь бидэнд одоогоор чухал мөрүүд. Сүүлээсээ хоёр дахь мөр нь ингэж уншигдаж байна:

```
FreeBSD/i386 (pc3.example.org) (ttyv0)
```

Энэ мөр нь таны дөнгөж сая эхлүүлсэн системийн тухай товч мэдээллийг агуулсан байна. Та x86 архитектуртай тохиромжтой Интел юм уу тохирох систем дээр ажиллаж буй "FreeBSD" консолыг харж байна . Энэ машины нэр нь (UNIX® машин болгон өөрийн нэртэй ) `pc3.example.org`, бөгөөд та энэ системийн консолын - ttyv0 нэртэй терминалыг харж байна.

Эцэст нь, сүүлийн мөр бол үргэлж:

```
login:
```

Энэ хэсэг нь, таны FreeBSD уруу нэвтрэх "хэрэглэгчийн-нэрээ" бичдэг хэсэг. Дараагийн хэсэгт энэ талаар тодорхойлсон байгаа.

### 4.2.2. FreeBSD уруу нэвтрэх

FreeBSD бол олон хэрэглэгчийн бас олон үйлдлийн систем. Энэ нь, нэг машин уруу олон хүмүүс зэрэг холбогдож нэгэн зэрэг олон үйлдэл хийж болдог системийн ерөнхий тодорхойлолт юм.

Олон хэрэглэгчийн систем нь ямар нэгэн аргаар нэг "хэрэглэгчийг" бусдаас нь ялгаж чаддаг байх ёстой. FreeBSD дээр (мөн бусад UNIX®-төрлийн системүүд дээр) энэ нь, хэрэглэгч ямар нэгэн програм ажиллуулахаасаа өмнө систем уруу "нэвтрэх үйлдэл" хийж хэрэгждэг. Хэрэглэгч бүрд гоц нэр ( "хэрэглэгчийн-нэр") мөн хувийн нууц үг ( "нууц-үг") харгалзах ёстой. Хэрэглэгчийг програм хэрэглэж эхлэхээс өмнө FreeBSD энэ хоёрыг асуух болно.

Дөнгөж FreeBSD эхлэх үедээ эхлүүлэгч бичлэгүүдийг гүйцэтгэж дуусаад танаас зөв хэрэглэгчийн нэр оруулахыг лавлаж асууна:

```
login:
```

Энэ жишээнд зориулаад бүгдээрээ таны хэрэглэгчийн нэрийг `john` гэж бодоцгооё. Лавлаж асуусан мөрөнд `john` гэж бичээд `Enter` товч дарна. Ингээд дараа нь танаас "password" гэж нууц үг асуух болно:

```
login: john
Password:
```

john-ны нууц үгийг оруулаад Enter товч дарна. Нууц үгийг *танд харуулдаггүй!* Энэ талаар та одоохондоо санаагаа зовоох хэрэггүй. Нууцлалын шалтгаанаар ингэж байгаа юм гэж хэлэхэд хангалттай.

Хэрэв та нууц үгээ зөв оруулсан бол ингээд FreeBSD уруу нэвтрэх бөгөөд боломжтой бүх програмуудыг туршиж болохоор боллоо.

Та MOTD эсвэл тухайн өдрийн мэдээллийг тушаал бичигчийн араас харах болно ( тушаал бичигч нь #, $, эсвэл % гэсэн тэмдэгтүүдийн нэг нь байдаг). Энэ нь таныг FreeBSD уруу амжилттай нэвтэрснийг илэрхийлдэг.

### 4.2.3. Олон консолууд

UNIX® тушаалуудыг нэг консол дээр гүйцэтгэх нь ердийнх, гэхдээ FreeBSD олон програмыг нэг дор ажиллуулж чадна. FreeBSD олон програмыг нэг дор нэг цагт ажиллуулж чадаж байхад, нэг консол дээр олон тушаал өгч суух нь цаг алдахын л нэмэр. Ийм үед л "виртуал консолын" тусламж хэрэг болдог.

FreeBSD нь танд олон төрлийн виртуал консол хэрэглүүлэхээр тохируулагдаж чаддаг. Нэг виртуал консолоос нөгөөх уруу нь гарын хэдхэн товчлуур дараад л шилжиж болдог. Консол бүр өөрийн төрөл бүрийн гаралтын сувагтай бөгөөд FreeBSD нь таныг нэг консолоос нөгөө консол уруу сэлгэх үед гарын оруулга болон дэлгэцийн гаралтыг тухайн консолд тохируулж дамжуулж өгдөг.

Гарын тусгай товчлолууд нь FreeBSD дээр консол сэлгэхэд зориулж нөөцлөгдсөн байдаг . Та Alt + F1, Alt + F2, аас Alt + F8 хүртэлх товчлуурыг дарж FreeBSD дээр консолуудын хооронд шилжиж болно.

Таныг нэгээс нөгөөх уруу нь шилжих үед, FreeBSD таны дэлгэцийн гаралтыг хадгалж санаа тавьдаг. Ингэсний үр дүнд гарнаас тушаал оруулж програмуудыг ажиллуулж болдог маш олон "виртуал" дэлгэцүүдтэй мэт үзэгддэг. Нэг виртуал консол дээр таны ажиллуулсан програм нь уг консолоос сэлгээд өөр консол дээр сэлгэсэн ч гэсэн зогсохгүй ажиллаж л байдаг.

### 4.2.4. /etc/ttys Файл

FreeBSD анхны тохируулгаараа найман виртуал консолтой эхэлдэг. Энэ нь хатуу тогтоосон тохируулга биш бөгөөд хэрэв та хүсвэл олон юм уу цөөн болгож амархан өөрчлөх боломжтой. Виртуал консолын тоо болон тохируулга нь /etc/ttys файлд байдаг.

Та /etc/ttys файлыг хэрэглэж FreeBSD дээрх виртуал консолуудыг тохируулж болно. Энэ файлын тайлбар тавиагүй мөр болгон ( # тэмдгээр эхлээгүй мөрүүд) виртуал консол эсвэл ганц терминалын тохируулгыг агуулж байдаг. Энэ файл FreeBSD-тэй анх цуг ирэхдээ есөн виртуал консолтой гэж тохируулагдаж ирдэг бөгөөд тэдний наймыг нь хэрэглэж болохыг зөвшөөрсөн байдаг. Тэдгээр нь ttyv гэж эхэлсэн байдаг:

```
# name  getty                           type    status          comments
#
ttyv0   "/usr/libexec/getty Pc"         cons25  on  secure
```

```
# Virtual terminals
ttyv1   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv2   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv3   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv4   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv5   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv6   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv7   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv8   "/usr/X11R6/bin/xdm -nodaemon"  xterm   off secure
```

Виртуал консолыг тохируулдаг энэ файлын баганууд мөн нэмэлт сонголтуудын дэлгэрэнгүй тайлбарыг та ttys(5) гарын авлагаас харна уу.

### 4.2.5. Ганц хэрэглэгчийн горимын консол

"Ганц хэрэглэгчийн горим" гэж юу болох талаар дэлгэрэнгүй мэдээлэл Ганц-хэрэглэгчийн горим-д буй. FreeBSD дээр ганц хэрэглэгчийн горимд ажиллаж байхад зөвхөн нэг консол байдаг гэдгийг мэдэх хэрэгтэй. Энэ үед ямар ч виртуал консолууд боломжгүй. Ганц хэрэглэгчийн горимын тохируулгыг бас /etc/ttys файл дотор хийж болно. `console` гэж эхэлсэн мөрүүдийг харна уу:

```
# name  getty                           type    status          comments
#
# If console is marked "insecure", then init will ask for the root password
# when going to single-user mode.
console none                            unknown off secure
```

Дээрх `console` гэсэн мөрний дээд хэсэгт бичсэн тайлбарт бичсэний дагуу та `secure` гэдгийг `insecure` гэж өөрчилж болно. Хэрэв та ингэвэл, FreeBSD ганц хэрэглэгчийн горимд эхэлсэн ч гэсэн танаас `root` нууц үг асуух болно.

*Үүнийг* `_insecure` гэж солихдоо та тун болгоомжтой байх хэрэгтэй_. Хэрэв та хэзээ нэгэн цагт `root` нууц үгээ мартаад ганц хэрэглэгчийн горимд орвол жаахан хүндрэлтэй байх болно. Энэ нь гэхдээ шийдэж болохоор хүндрэл боловч FreeBSD-н эхлэх үйлдэл болон хамрагдсан програмуудад нь дасаагүй хүнд бол жаахан хүндрэлтэй байж магадгүй.

### 4.2.6. Консолын Видео горимуудыг өөрчлөх нь

FreeBSD-ийн консолын анхдагч видео горимыг 1024x768, 1280x1024, эсвэл таны график бичил схем болон дэлгэц дэмжиж чадах дурын хэмжээгээр тохируулж болох юм. Өөр видео горимыг ашиглахын тулд та `VESA` модулийг дуудна:

```
# kldload vesa
```

Дараа нь таны тоног төхөөрөмж ямар видео горимуудыг дэмждэгийг vidcontrol(1) хэрэгсэл

ашиглан та тодорхойлж болно. Дэмжигдсэн видео горимуудын жагсаалтыг авахын тулд доор дурдсан тушаалыг бичнэ:

```
# vidcontrol -i mode
```

Энэ тушаалын үр дүн нь таны тоног төхөөрөмжийн дэмждэг видео горимуудын жагсаалт байх болно. Та дараа нь `root` консол дээр шинэ видео горимыг сонгон vidcontrol(1) уруу өгч болно:

```
# vidcontrol MODE_279
```

Хэрэв шинэ видео горим боломжийн бол /etc/rc.conf файлд ачаалахад үүнийг үргэлж сонгодог байхаар тохируулж болно:

```
allscreens_flags="MODE_279"
```

## 4.3. Зөвшөөрлүүд

FreeBSD нь BSD UNIX®-ээс уламжилж байх үедээ, UNIX®-н үндсэн хэдэн бүтэц дээр суурилсан. Эхний бөгөөд хамгийн чухал тунхаглал нь FreeBSD бол олон хэрэглэгчийн үйлдлийн систем юм. Систем нь олон хэрэглэгчийн хоорондоо хамааралгүй олон үйлдлийг ачаалж чаддаг. Хэрэглэгч болгонд шаардагдах компьютерийн төхөөрөмж, санах ой мөн процессорын давтамжийг зөв шударга хуваарилахыг систем хариуцдаг.

Систем олон хэрэглэгчтэй ажиллах болохоор, системийн хийх ёстой зүйл нь хэн тухайн нөөцийг унших, бичих, гүйцэтгэх вэ гэдгийг зохицуулах юм. Эдгээр зөвшөөрлүүд нь гурав гурваараа нийлүүлэгдсэн гурван хэсэг тоо байдаг. Нэг нь файлын эзэмшигчид, нөгөөх нь файлын хамаарагддаг бүлэгт, үлдсэн нь хүн болгонд гэж хуваарилагдана. Энэ тоон дараалал нь дараах маягаар ажилладаг.:

| Утга | Зөвшөөрөл | Жагсаах харуулалт |
|---|---|---|
| 0 | Уншигдахгүй, бичигдэхгүй, гүйцэтгэгдэхгүй | `---` |
| 1 | Уншигдахгүй, бичигдэхгүй, гүйцэтгэгдэнэ | `--x` |
| 2 | Уншигдахгүй, бичигдэнэ, гүйцэтгэгдэхгүй | `-w-` |
| 3 | Уншигдахгүй, бичигдэнэ, гүйцэтгэгдэнэ | `-wx` |
| 4 | Уншигдана, бичигдэхгүй, гүйцэтгэгдэхгүй | `r--` |

| Утга | Зөвшөөрөл | Жагсаах харуулалт |
|---|---|---|
| 5 | Уншигдана, бичигдэхгүй, гүйцэтгэгдэнэ | `r-x` |
| 6 | Уншигдана, бичигдэнэ, гүйцэтгэгдэхгүй | `rw-` |
| 7 | Уншигдана, бичигдэнэ, гүйцэтгэгдэнэ | `rwx` |

Та ls(1) тушаалыг `-l` сонголттой хэрэглэж файлын эзэмшигч, бүлэг, мөн хүн болгонд хуваарилсан зөвшөөрлийг харуулсан баганатай дэлгэрэнгүй мэдээллийг харж болно. Жишээлбэл, `ls -l` тушаалыг нэг сан дотор гүйцэтгэвэл дараах маягаар харагдана:

```
% ls -l
total 530
-rw-r--r--  1 root  wheel     512 Sep  5 12:31 myfile
-rw-r--r--  1 root  wheel     512 Sep  5 12:31 otherfile
-rw-r--r--  1 root  wheel    7680 Sep  5 12:31 email.txt
...
```

Энд `ls -l` тушаалын эхний баганыг авч үзье:

```
-rw-r--r--
```

Хамгийн эхний (зүүн талын) тэмдэг нь үүнийг ердийн файл юм уу, эсвэл сан, онцгой тэмдэг төхөөрөмж, сокет, эсвэл ямар нэгэн холбоост файл мөн эсэхийг илэрхийлдэг. Энэ тохиолдолд `-` нь ердийн файлыг зааж байна. Дараагийн гурван тэмдэгт `rw-` нь энэ тохиолдолд энэ файлын эзэмшигчид зөвшөөрөл өгч байна. Дараагийн гурван тэмдэгт `r--` нь файлын хамаарах бүлэгт эрх өгч байна. Сүүлийн гурван тэмдэгт `r--` нь бусад бүх хүмүүст эрх өгч байна. Зураас нь зөвшөөрөл өгөөгүйг илэрхийлдэг. Энэ файлын тохиолдолд, файлыг унших бичих эрхийг эзэмшигчид нь өгөөд, бүлэг нь файлыг унших эрхтэй, бусад хүмүүс энэ файлыг зөвхөн унших эрхтэй гэж заагдсан байна. Дээр тайлбарласан хүснэгтийн дагуу, энэ файлын зөвшөөрөл нь `644` гэж дээрх гурван төрөлд заагдаж байна.

Энэ бол сайн бөгөөд сайхан хэрэг, гэхдээ төхөөрөмж рүү хандах зөвшөөрлийг систем яаж хянах вэ? FreeBSD бараг ихэнх төхөөрөмжийг нээж уншиж өгөгдөл бичдэг файл мэт хандаж уншиж бичдэг. Тэдгээр онцгой төхөөрөмжүүд нь /dev сан дотор байрладаг.

Сангууд ч гэсэн бас файл мэт хандагддаг. Тэд нарт бас уншигдах, бичигдэх, гүйцэтгэгдэх зөвшөөрлүүд байдаг. Сангийн гүйцэтгэгдэх тэмдэг нь файлаас арай өөрөөр хэрэгждэг. Хэрэв сан нь гүйцэтгэгдэнэ гэж тэмдэглэгдсэн бол, энэ сан нь дамжиж өнгөрүүлэгдэж болох буюу "cd" тушаалыг (сан сольдог) хэрэглэж ийшээ орж болно гэсэн үг. Энэ нь мөн сан дотор харагдаж байгаа файлууд уруу хандаж болно гэсэн үг. (бас нэг зүйл, мэдээжээр, файлууд бас өөр дээрээ зөвшөөрөл агуулсан байгаа).

Ерөнхийдөө бол, сан доторх файлуудыг жагсааж харахыг хүсвэл, уг санд уншигдах зөвшөөрөл суугдсан байх ёстой. Хэрэв санд байгаа файлыг устгана гэвэл бичигдэх *болон*

гүйцэтгэгдэх зөвшөөрлүүд уг санд суугдах ёстой.

Мөн нэмээд хэдэн зөвшөөрлийн битүүд байдаг, гэхдээ тэдгээр нь ажилладаг файлын setuid дугаар болон sticky сан зэрэг тусгай зориулалтаар хэрэглэгддэг. Файлын зөвшөөрлийн талаар нэмэлт мэдээлэл авахыг хүсвэл chmod(1) гарын авлага хуудаснаас хараарай.

### 4.3.1. Тэмдэгт зөвшөөрлүүд

Сан эсвэл файлд тоон утганы оронд заримдаа тэмдгэн зөвшөөрлийг хэрэглэдэг. Тэмдгэн зөвшөөрлийг бичихдээ (хэн) (үйлдэл) (зөвшөөрөл) гэсэн дарааллаар бичих бөгөөд дараах утгуудыг авдаг:

| Сонголт | Үсэг | Илэрхийлэл нь |
|---|---|---|
| (хэн) | u | Хэрэглэгч |
| (хэн) | g | Бүлгийн эзэмшигч |
| (хэн) | o | Бусад |
| (хэн) | a | Бүгд ("ертөнц") |
| (үйлдэл) | + | Зөвшөөрөл нэмэх |
| (action) | - | Зөвшөөрлийг устгах |
| (үйлдэл) | = | зөвхөн тухайн зөвшөөрлийг суулгах |
| (зөвшөөрөл) | r | Унших |
| (зөвшөөрөл) | w | Бичих |
| (зөвшөөрөл) | x | Гүйцэтгэх |
| (зөвшөөрөл) | t | Sticky бит |
| (зөвшөөрөл) | s | UID эсвэл GID суулгах |

Эдгээр утгууд нь өмнөх жишээ шиг chmod(1) тушаалтай хэрэглэгддэг бөгөөд гэхдээ үсэг хэрэглэнэ. Жишээлбэл, та *FILE* уруу бусад хэрэглэгчид хандахыг хориглохдоо:

```
% chmod go= FILE
```

Файлд нэгээс олон өөрчлөлт хийх шаардлага гарвал таслалаар тусгаарласан мөр бичиж болно. Жишээлбэл, дараах тушаал нь бүлэг болон "дэлхийг"*FILE* дээр бичих эрхийг нь аваад дараа нь хүн болгонд гүйцэтгэж болно гэсэн эрх өгч байна:

```
% chmod go-w,a+x FILE
```

### 4.3.2. FreeBSD файлын туг

Өмнө ярилцсан файлын зөвшөөрлөөс гадна FreeBSD нь "файлын туг" хэрэглээгээр хангагдсан байдаг. Эдгээр туг нь файлд нэмэлт нууцлалын болон хяналтын түвшин тогтоож

өгдөг, гэхдээ санд бол үгүй.

Эдгээр файлын тугнууд нь файлд нэмэлт түвшний хяналт тогтоож өгснөөрөө зарим тохиолдолд `root` хэрэглэгч хүртэл файлыг устгах юм уу өөрчилж чадахгүй болгож тусалдаг.

Файлын тугнууд нь энгийн загвартай chflags(1) багажаар ашиглагддаг. Жишээлбэл, file1 файл дээр устгагдахгүй гэсэн туг хатгахыг системд зөвшөөрүүлэхийн тулд дараах тушаалыг гүйцэтгэнэ:

```
# chflags sunlink file1
```

Хэрэв устгагдахгүй тугийг буцааж авна гэвэл өмнөх тушаал дээрээ `sunlink`-ын өмнө "no" залгаж ажиллуулна:

```
# chflags nosunlink file1
```

Энэ файлын тугийг харахын тулд ls(1) тушаалыг `-lo` сонголттой хамт ажиллуулна:

```
# ls -lo file1
```

Тушаалын гаралт нь дараах маягаар харагдах ёстой:

```
-rw-r--r--  1 trhodes  trhodes  sunlnk 0 Mar  1 05:54 file1
```

Нэлээн хэдэн тугнууд нь файлд зөвхөн `root` хэрэглэгчээр нэмэгдэж устгагдана. Бусад тохиолдолд файлын эзэмшигч нь тугийг суулгах боломжтой. Администраторуудыг chflags(1) болон chflags(2) гарын авлагуудыг уншихыг зөвлөж байна.

### 4.3.3. `setuid`, `setgid`, болон `sticky` буюу наалдамхай зөвшөөрлүүд

Өмнө хэлэлцсэн зөвшөөрлүүдээс гадна бүх администраторуудын мэдэх ёстой өөр гурван тусгай тохиргоо байдаг. Эдгээр нь `setuid`, `setgid` болон `sticky` буюу наалдамхай зөвшөөрлүүд юм.

Эдгээр тохиргоонууд нь ерөнхийдөө энгийн хэрэглэгчдэд зөвшөөрөгддөггүй ажиллагаагаар хангадаг бөгөөд UNIX®-ийн зарим нэг үйлдлүүдэд чухал байдаг. Эдгээрийг ойлгохын тулд жинхэнэ хэрэглэгчийн ID болон хүчинтэй хэрэглэгчийн ID-ийн ялгааг дурдах хэрэгтэй.

Жинхэнэ хэрэглэгчийн ID нь процессийг эзэмшдэг юм уу эсвэл процессийг эхлүүлдэг UID юм. Хүчинтэй хэрэглэгчийн UID нь процессийн ажиллаж байгаа тэр хэрэглэгчийн ID юм. Жишээ нь passwd(1) хэрэгсэл нь хэрэглэгч өөрсдийн нууц үгээ сольж байгаа болохоор жинхэнэ хэрэглэгчийн ID-аар ажиллах боловч нууц үгийн санд өөрчлөлт хийхийн тулд `root` хэрэглэгчийн хүчинтэй ID-аар ажилладаг. Ингэснээр энгийн хэрэглэгчдэд `Permission Denied` буюу зөвшөөрөл хаалттай гэсэн алдааг харуулалгүйгээр өөрсдийн нууц үгсийг солих боломжийг олгодог.

nosuid mount(8) тохиргоо нь эдгээр хоёртын файлуудыг дуугай амжилттүй болоход хүргэдэг. Энэ нь хэрэглэгчдэд мэдэгдэлгүйгээр ажиллагаа амжилттүй болно гэсэн үг юм. mount(8) гарын авлагын дагуу энэ тохиргоо нь nosuid гүйцэтгэл хялбаршуулагчийн хамгаалж болох шиг бүрэн найдвартай бас биш юм.

setuid зөвшөөрлийг зөвшөөрлийн цуглуулгын өмнө дөрвийн тоог (4) доорх жишээн дээрх шигээр тавьж тохируулж болно:

```
# chmod 4755 suidexample.sh
```

suidexample.sh файл дээрх зөвшөөрлүүд нь одоо доорх шиг харагдах ёстой:

```
-rwsr-xr-x   1 trhodes  trhodes    63 Aug 29 06:36 suidexample.sh
```

Энэ жишээн дээр s нь ажиллуулах битийг сольж файлын эзэмшигчид зориулсан зөвшөөрлүүдийн цуглуулгын хэсэг болж байгаа нь харагдах ёстой. Энэ нь passwd зэрэг дээшлүүлсэн зөвшөөрлүүдийг шаарддаг хэрэгслүүдийг зөвшөөрдөг.

Жинхнээр нь үүнийг харахын тулд хоёр терминал нээ. Нэг дээр нь энгийн хэрэглэгчээр passwd процессийг эхлүүл. Шинэ нууц үг хүлээж байхад нь процессийн хүснэгтийг шалгаад passwd тушаалын хэрэглэгчийн мэдээллийг хар.

Терминал A дээр:

```
Changing local password for trhodes
Old Password:
```

Терминал B дээр:

```
# ps aux | grep passwd
```

```
trhodes  5232  0.0  0.2  3420  1608   0  R+    2:10AM   0:00.00 grep passwd
root     5211  0.0  0.2  3620  1724   2  I+    2:09AM   0:00.01 passwd
```

Дээр харуулснаар passwd нь энгийн хэрэглэгчээр ажиллаж байгаа боловч root хэрэглэгчийн хүчинтэй UID ашиглаж байна.

setgid зөвшөөрөл нь setuid зөвшөөрөлтэй адил үүргийг гүйцэтгэдэг, гэхдээ бүлгийн тохиргоог өөрчилдөг. Програм юм уу эсвэл хэрэгсэл нь энэ тохиргоотойгоор ажиллахдаа процессийг эхлүүлсэн хэрэглэгч биш файлыг эзэмшиж байгаа бүлэг дээр тулгуурласан зөвшөөрлүүдийг олгодог.

Файл дээр setgid зөвшөөрлийг тохируулахдаа дараах жишээн дээрх шиг chmod тушаалыг

өмнөө хоёртой (2) өгнө:

```
# chmod 2755 sgidexample.sh
```

Шинэ тохиргоог өмнөх шигээ харж болох бөгөөд `s` нь одоо бүлгийн зөвшөөрлийн тохиргоонд зориулагдсан талбарт байгааг анхааараарай:

```
-rwxr-sr-x   1 trhodes  trhodes    44 Aug 31 01:49 sgidexample.sh
```

Эдгээр жишээнүүд дээр бүрхүүлийн скрипт нь хэдийгээр ажиллах боломжтой файл боловч өөр EUID буюу хүчинтэй хэрэглэгчийн ID-аар ажиллахгүй. Энэ нь яагаад гэвэл бүрхүүлийн скрипт нь setuid(2) системийн дуудлагуудад хандаж чадахгүйтэй холбоотой юм.

Бидний хэлэлцсэн эхний хоёр тусгай зөвшөөрлийн битүүд нь (`setuid` болон `setgid` зөвшөөрлийн битүүд) дээшлүүлсэн зөвшөөрлүүдийг зөвшөөрч системийн аюулгүй байдлыг доошлуулж болох юм. Системийн аюулгүй байдлыг чангатгаж чадах гурав дахь тусгай зөвшөөрлийн бит байдаг нь `sticky bit` буюу наалдамхай бит юм.

`sticky bit` нь санд тавигдсан үед файл устгалтыг зөвхөн файлыг эзэмшигчид зөвшөөрдөг. Энэ зөвшөөрлийн цуглуулга нь /tmp зэрэг нийтийн сангаас файлыг эзэмшдэггүй хэрэглэгч уг файлыг устгахаас хамгаалахад ашиг тустай байдаг. Энэ зөвшөөрлийг ашиглахын тулд зөвшөөрлийн урд нэгийг (1) тавьж өгнө. Жишээ нь:

```
# chmod 1777 /tmp
```

Одоо үр дүнг `ls` тушаал ашиглан харах боломжтой:

```
# ls -al / | grep tmp
```

```
drwxrwxrwt  10 root  wheel         512 Aug 31 01:49 tmp
```

`sticky bit` буюу наалдамхай бит зөвшөөрөл нь цуглуулгын хамгийн сүүлд байгаа `t`-ээс ялгаатай юм.

## 4.4. Сангийн бүтэц

FreeBSD ийн сангийн бүтэц нь уг системийг ойлгоход тун чухал үүрэг гүйцэтгэнэ. Хамгийн чухал бөгөөд үндсэн ойлголт бол root буюу хамгийн дээд эх сан "/" юм. Энэ сан нь систем эхлэх үед хамгийн түрүүнд танигдах ёстой бөгөөд олон хэрэглэгчид зориулсан үйлдлийн системийг бэлдэхэд чухал үүрэгтэй. Энэ дээд эх сан нь олон хэрэглэгчийн ажиллагаанд шилжих үед холбогдох бусад файлын системүүдийн холболтын цэгүүдийг бас агуулж байдаг.

Холболтын цэг нь үндсэн файл систем уруу (ихэвчлэн эх root файл систем) нэмэлт файлын системүүдийг холбож нэмдэг цэг юм. Энэ тухай цаашид Диск зохион байгуулалт-д тайлбарласан байгаа. Үндсэн холбох цэгүүд нь /usr, /var, /tmp, /mnt, мөн /cdrom сангуудыг агуулж байдаг. Эдгээр сангууд нь голдуу /etc/fstab файлд заагдсан байдаг. /etc/fstab файл нь төрөл бүрийн файл системүүдийг системд ойлгуулах гэж бичсэн хүснэгт бичлэг. /etc/fstab файлд байгаа ихэнх файл системүүд компьютер эхэлж ачаалагдах үед автоматаар rc(8) гүйцэтгэгддэг бичлэгээс таниулагддаг. Гэхдээ хэрэв `noauto` гэсэн сонголтыг агуулсан бол автоматаар таниулагддаггүй. Дэлгэрэнгүй мэдээлэл fstab файл хэсэгт буй.

Файл системийн бүтцийн тухай бүрэн мэдээллийг hier(7)-с харж болно. Одоохондоо байнга хэрэглэгддэг сангуудыг товч тайлбарлахад хангалттай.

| Сан | Тодорхойлолт |
|---|---|
| / | Файл системийн Root буюу хамгийн дээд эх сан. |
| /bin/ | Ганц болон олон хэрэглэгчийн орчны үндсэн хэрэгслийн сан. |
| /boot/ | Үйлдлийн систем эхлэж ачаалагдах үеийн програмууд болон тохируулга файлууд. |
| /boot/defaults/ | Анхдагч ачаалах үеийн тохируулгын файлууд; дэлгэрэнгүйг loader.conf(5) хуудаснаас харна уу. |
| /dev/ | Төхөөрөмжүүд байдаг сан; intro(4) хуудаснаас харна уу. |
| /etc/ | Системийн тохируулгын файлууд болон гүйцэтгэх бичлэгүүд. |
| /etc/defaults/ | Анхдагч системийн тохируулгын файлууд; дэлгэрэнгүйг rc(8) хуудаснаас хараарай. |
| /etc/mail/ | sendmail(8) мэтийн захиа илгээгчдийн тохируулгын файлууд. |
| /etc/namedb/ | `named`-н тохируулгын файл; named(8) хуудаснаас дэлгэрэнгүйг харна уу. |
| /etc/periodic/ | cron(8)-г ашиглаж өдрөөр, сараар, эсвэл жилээр гэх мэт давтамжаар ажиллаж чаддаг бичлэгүүд ; дэлгэрэнгүйг periodic(8) хуудаснаас харна уу. |
| /etc/ppp/ | `ppp` тохируулга файлууд; ppp(8)-с харна уу. |
| /mnt/ | Администратор голдуу түр зуур бусад файл систем эсвэл төхөөрөмжийг таниулахад хэрэглэдэг хоосон сан. |
| /proc/ | Процесийн файл систем; procfs(5) болон mount_procfs(8)-с харна уу. |

| Сан | Тодорхойлолт |
| --- | --- |
| /rescue/ | Эвдэрч гэмтсэн үед сэргээхэд хэрэглэгддэг программууд rescue(8)-с харна уу. |
| /root/ | `root` хэрэглэгчийн гэрийн сан. |
| /sbin/ | Ганц болон олон хэрэглэгчийн орчинд хэрэглэж болохоор системийн програм ба администраторын үндсэн хэрэгслүүд. |
| /tmp/ | Түр зуурын файлууд. /tmp сан доторх файлууд нь систем шинээр ачаалагдах үед хадгалагдалгүйгээр устгагдана. Санах ойгоос ажилладаг файл системүүд голцуу /tmp санд таниулагдаж байрлуулагддаг. Энэ үйлдэл нь rc.conf(5)-тай холбоотой tmpmfs-төрлийг ашиглаж автоматжуулагдаж болдог (эсвэл /etc/fstab дотор буй оруулгууд бүгдээрээ; mdmfs(8)-с харна уу). |
| /usr/ | Програм болон хэрэглэгчийн хэрэглэдэг үндсэн хэрэгслүүд. |
| /usr/bin/ | Үндсэн хэрэгслүүд, програмын багажууд, мөн программууд. |
| /usr/include/ | Стандарт C-ийн include файлууд. |
| /usr/lib/ | Програмын шахаж бэлдсэн багцууд. |
| /usr/libdata/ | Төрөл бүрийн хэрэгслийн өгөгдлийн файлууд. |
| /usr/libexec/ | Системийн далд чөтгөр буюу дэмон болон системийн хэрэгслүүд (бусад програмаас дуудагдаж ажилладаг). |
| /usr/local/ | Дотоод гүйцэтгэгддэг болон програмын сангууд гэх мэт программууд байдаг. Мөн FreeBSD-н портуудын анхны байрлуулах газар болж өгдөг. /usr/local -ийн доторх нь, hier(7)-аар /usr сангийн дотор байгааг байрлуулсан шиг зохион байгуулагдах хэрэгтэй. Гэхдээ энэнд хамаарахгүй сангууд нь man гэж /usr/local/share сан дотор биш /usr/local дотор байрладаг сан, мөн share/doc/port дотор байдаг портуудын бичиг баримт байдаг сан юм. |
| /usr/obj/ | Архитектураасаа хамаараад /usr/src санг хөрвүүлэх үед бүтээгддэг сан.. |
| /usr/ports/ | FreeBSD Портын цуглуулга (сонгомол). |

| Сан | Тодорхойлолт |
| --- | --- |
| /usr/sbin/ | Системийн дэмон болон системийн хэрэгслүүд (хэрэглэгчдээр гүйцэтгэгдэнэ). |
| /usr/shared/ | Архитектуртаа хамаатай файлууд. |
| /usr/src/ | BSD болон/эсвэл дотоод эх файлууд. |
| /usr/X11R6/ | X11R6 цогцолборын гүйцэтгэж болдог програмууд, програмын сан зэргүүд (сонгомол). |
| /var/ | Олон зорилгоор хэрэглэгддэг бүртгэл бичлэг, түр зуурын, дараалал зэргийн файлууд. Санах ойд тулгуурлаж ажилладаг зарим файлын системүүд /var дотор танигдаж үүсгэгддэг. Энэ үйлдэл нь rc.conf(5)-тай холбоотой varmfs-төрлийг ашиглаж автоматжуулагдаж болдог (эсвэл /etc/fstab дотор буй оруулгууд бүгдээрээ; mdmfs(8)-с харна уу). |
| /var/log/ | Системийн төрөл бүрийн бүртгэл бичлэгийн файлууд. |
| /var/mail/ | Хэрэглэгчийн ирсэн захиаг хадгалах файлууд. |
| /var/spool/ | Төрөл бүрийн хэвлэгч болон захианы системийн дарааллыг удирдах сангууд. |
| /var/tmp/ | Түр зуурын файлууд. Энэ санд буй файлууд нь голдуу системийг эхлэх үед зориулагдахаар нөөцлөгдсөн байдаг. Гэхдээ, хэрэв /var нь санах ойд тулгуурласан файлын систем бол өмнө хэлсэн зүйл хүчингүй. |
| /var/yp/ | NIS maps. |

## 4.5. Диск зохион байгуулалт

FreeBSD нь файл олохын тулд хэрэглэдэг хамгийн жижиг хэрэгсэл бол файлын нэр юм. Файлын нэрний том болон бага үсэгнүүд нь бас ялгаатай. Энэ нь юу гэсэн үг вэ гэвэл readme.txt болон README.TXT гэсэн хоёр файл нь хоорондоо ялгаатай файл гэсэн үг. FreeBSD нь файлын төрлийг програм, бичиг баримт, эсвэл бусад төрлийн файл байна гэж ялгахын тулд (.txt) гэх мэтийн өргөтгөл хэрэглэдэггүй.

Файлууд сан дотор хадгалагддаг. Сан нь зуу зуун файл агуулж болох бөгөөд эсвэл хоосон байж болно. Сан нь сангаа агуулж бас болох ба ингэж сангийн угсарсан модлог бүтэц үүсгэж болно. Ингэвэл та файлуудаа илүү амархан зохицуулах болно.

Файл ба сангууд нь / тэмдгийн араас шаардлагатай бол сангуудын нэрийг бичиж өгөгдсөн нэрээрээ хандагдана. Хэрэв танд foo нэртэй сан нь bar санг агуулдаг бөгөөд уг сан дотор

readme.txt файл байгаа бол, файлын хандах бүтэн нэр буюу *зам* нь foo/bar/readme.txt гэж бичигдэнэ.

Сан болон файлууд нь файл системд хадгалагддаг. Файл систем болгон хамгийн эхний дээд хэсэгт заавал нэг сан агуулдаг бөгөөд, түүнийг уг файл системийн *root сан* буюу дээд эх сан гэж нэрлэдэг. Тэгээд энэ эх сан нь цаашаагаа өөр сангуудыг агуулж явдаг.

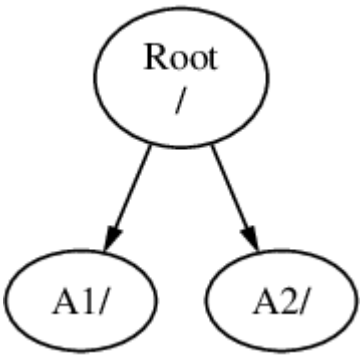
Энэ хүртэл таны уншсан зүйлүүд нь таны бусад мэддэг үйлдлийн системтэй ижил байж магадгүй. Гэхдээ жаахан ялгаанууд бий. Жишээлбэл MS-DOS® үйлдлийн систем нь файл болон санг тусгаарлахдаа `\` тэмдэг хэрэглэдэг байхад Mac OS® үйлдлийн систем нь `:` тэмдгийг хэрэглэдэг.

FreeBSD дискэнд үсэглэж нэр өгдөггүй бөгөөд үсгээр нэр өгөгдсөн файлын замыг хэрэглэдэггүй. Та FreeBSD дээр c:/foo/bar/readme.txt гэж бичиж болохгүй.

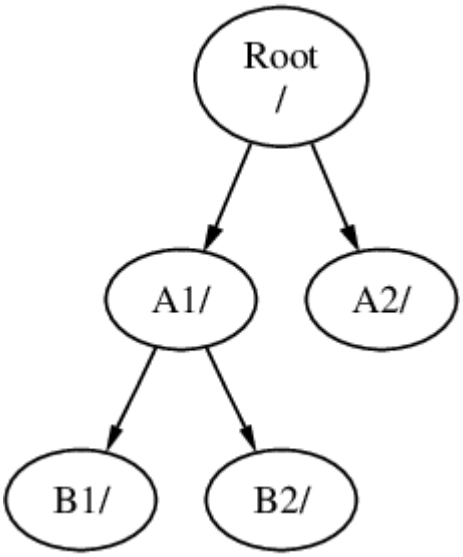
Харин түүний оронд нэг файл системийг нэг *root файл систем* гэж ангилдаг. Уг root буюу эх файл системийн эх сан нь `/` гэж хандагдана. Бусад өөр файл системүүд энэ root буюу эх файл систем дотор *холбогддог*. Та FreeBSD систем дээрээ хэдэн ч дисктэй байсан, сан болгонууд нь нэг дискний хэсэг мэт харагддаг.

Жишээлбэл `A`, `B`, мөн `C` гэсэн гурван файл систем танд байна гэж бодъё. Файл систем бүр нь өөртөө хоёр сан агуулсан тус тусын эх сантай гэж үзье. Тэдгээр сангууд нь дараах нэртэй байг. `A1`, `A2` (гэх мэтчилэн `B1`, `B2` мөн `C1`, `C2`).

`A`-г эх сан гэж үзнэ. Хэрэв та `ls` тушаалаар энэ сангийн дотор байгааг харвал, `A1` ба `A2` гэсэн хоёр санг та харах болно. Энэ сангийн модлог загвар нь ингэж харагдаж байна:

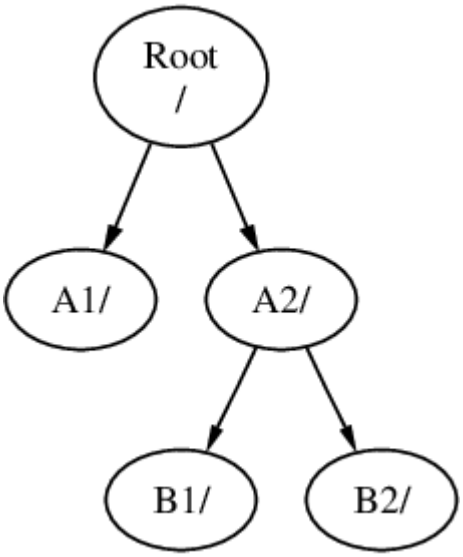



Хэрэв файл систем өөр файл системд холбогдохоор бол, холбогдож байгаа системийнхээ нэг сан дор холбогдоно. Тэгвэл одоо `B` файл системийг `A1` санд холбоно гэж үзье. Тэгвэл `B`-ийн эх сан нь `A1`-ээр орлуулагдаж, `B` доторх файлууд дараах маягаар харагдана:



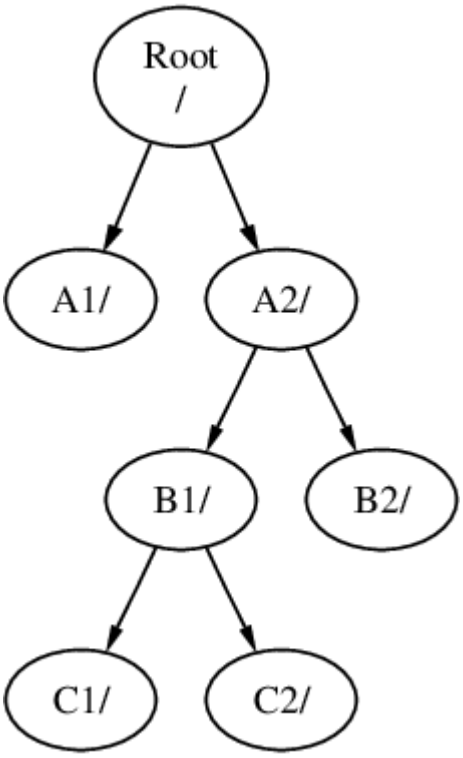

Хэрэв `B1` эсвэл `B2` сан доторх файлууд шаардлага гаран хандагдахаар бол /A1/B1 эсвэл /A1/B2 гэсэн зам хэрэглэгдэж бичигдэнэ. Хэрэв /A1 дотор файлууд байсан бол тэдгээрийг түр зуур нуудаг. `B` файл систем A системээс *салгагдсан* үед л тэр файлууд харагдана.

Хэрэв `B` файл систем `A2` дор холбогдсон бол дараах маягаар харагдана:

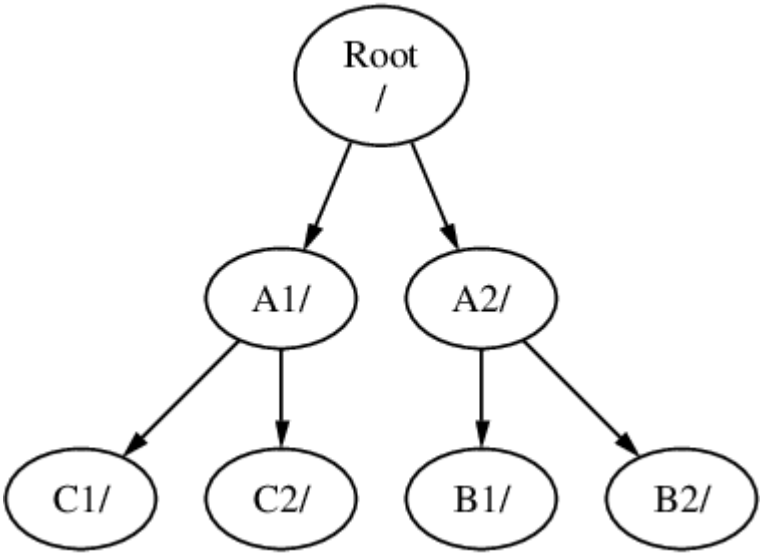



мөн файл уруу хандах зам нь /A2/B1 болон /A2/B2 гэж тус тусдаа хандагдах болно.

Файл системүүд нэг нэгнийхээ дээр холбогдож болдог. Сүүлийн жишээгээ үргэлжлүүлээд `C` файл системийг `B` файл систем доторх `B1` сангийн дээд хэсэгт холбож өгвөл дараах зохион байгуулалт үүсэж байна:




Эсвэл `C` файл систем `A` файл систем дотор `A1` санд холбогдож болно:




Хэрэв та MS-DOS® системийг гадарладаг бол энэ нь `join` тушаалтай төсөөтэй боловч яг адилхан биш.

Угтаа бол энэ нь тийм их анхаарлаа хандуулаад байхаар зүйл биш. Ердийн үед та FreeBSD суулгах үедээ нэг файл систем үүсгээд хаана холбохыг нь шийдэж холбоод, шинэ диск нэмэхгүй л бол түүнийгээ хэзээ ч өөрчлөх шаардлага гардаггүй.

Өөр файл систем үүсгэлгүйгээр нэг бүхэл эх файл систем үүсгэж болдог. Ийм үед зарим сул талууд гарч ирдэг бөгөөд нэг л давуу тал үүснэ.

*Олон файл системүүдийн давуу талууд*

- Олон төрлийн файл системүүд нь олон төрлийн *холбох нөхцөлтэй*. Жишээлбэл, сайн

төлөвлөсний дараагаар, эх файл систем нь зөвхөн уншигдахаар холбогдож, ингэснээр санамсаргүй юм уу алдаа ослын шалтгаанаар чухал файлуудыг устгахаас сэргийлж болно. Хэрэглэгчдээр бичигдэж болдог файл систем, жишээ нь /home мэтийн бусад системээс тусгаарлаж *nosuid* төрлийн гэж холбож болно; энэ сонголт нь файл систем дээр буй гүйцэтгэж болдог файлд *suid*/*guid* бит утга тавигдахаас сэргийлж аюулгүй байдлыг хангаж өгдөг.

- FreeBSD файл систем ямар зорилгоор хэрэглэгдэхээс нь хамаараад файл систем дээр файлын зохион байгуулалтыг автоматаар хийдэг. Тийм болохоор байнга бичигдэж байдаг олон жижигхэн файлуудын байгаа файл систем дээр цөөхөн бичигддэг том файл агуулсан файл системийг бодвол олон бичигдэхэд зориулж арай өөр файлын зохион байгуулалт хийгддэг. Ганц том эх файл системд ийм зохион байгуулалт хийх боломжгүй.
- FreeBSD-н файл систем нь цахилгаан тэжээлээс огцом салгагдсан үед ч дискний алдаа үүсгэдэггүй найдвартай байдаг. Гэвч маш ноцтой үед цахилгаан тэжээлээс огцом салгагдахад файл системийн бүтэц эвдэрч болзошгүй юм. Өгөгдлүүдээ олон файл системд хуваж байрлуулах нь дараа нь эвдрэлээс буцааж сэргээхэд амар байдаг.

*Нэг файл системтэй байх үеийн давуу тал*

- Энэ файл систем нь тогтсон хэмжээтэй байдаг. Хэрэв та FreeBSD-ийг суулгах үедээ тодорхой зааж өгсөн файл систем үүсгээд, сүүлд нь түүнийгээ өргөжүүлэхийг хүсвэл, энэ нь тийм амархан биелэхгүй. Та эхлээд байгаа файл системийнхээ файлуудыг нөөцөлж хадгалаад дараа нь файл системдээ шинэ хэмжээ өгч өргөтгөөд дараа нь нөөцөөсөө файлуудаа буцааж сэргээж хадгалах болно.

FreeBSD-н growfs(8) тушаал нь дээрх хүндрэлийг алга болгож, файлыг нөөцлөх шаардлагагүйгээр шууд файл системийг өргөтгөж болдог болсон.

Дискний хуваалт дотор файл систем оршиж байдаг. FreeBSD нь юниксээс уламжилсан болохоор диск хуваалтын ойлголт нь ердийн хэрэглээтэй ижилхэн биш (жишээлбэл, MS-DOS® диск зохион байгуулалт). Дискний хуваагдал бүр `a` үсэгнээс эхлээд `h` хүртэл үсгээр тэмдэглэгддэг. Хуваагдал бүр зөвхөн ганцхан файл систем агуулах ёстой. Ийм болохоор файл систем нь агуулж байгаа диск хуваалтынхаа үсгээр илэрхийлэгдэх бөгөөд хэрэв өөр файл системд холбогдвол, холбосон сан нь уг файл системийг илэрхийлнэ.

FreeBSD мөн *swap _-д зориулж дискэнд зай бэлддэг. Swap хэмжээ нь FreeBSD-н _виртуал санах ой* юм. Ингэснээр таны компьютер байгаа бодит хэмжээнээсээ илүү санах ой хэрэглэж байгаа мэт ажилладаг. Хэрэв зарим програм FreeBSD дээр ажиллаад санах ойноос хэтрэх хүндрэл гарвал, хэрэглэгдэхгүй байгаа хэсгийг swap зай руу зөөж, шаардлагатай үед буцааж санах ой руу зөөх зарчмаар хэрэглэгддэг.

Зарим дискний хуваалт нь тогтсон журамтай байдаг.

| Хуваалт | Тогтсон журам |
|---|---|
| `a` | Ихэнхдээ root файл системийг агуулж байдаг |
| `b` | Ихэнхдээ swap хэмжээг агуулдаг |

| Хуваалт | Тогтсон журам |
| --- | --- |
| c | Ихэнх үед агуулж буй зүсмэлийнхээ хэмжээтэй ижил хэмжээтэй байдаг. Энэ нь ямар нэгэн хэрэгслүүдийг (жишээлбэл, дискний эвдэрсэн хэсгийг шалгагч) c хуваалт дээрх зүсмэл дээр бүхэлд нь ажиллаж болох зөвшөөрөл өгдөг. Ердийн үед та үүн дээр файл систем үүсгэх шаардлага байхгүй. |
| d | Урьд нь d хуваалт онцгой үүрэгтэй байсан боловч одоо тийм биш, харин ердийн хуваалт шиг хэрэглэгдэж болно. |

FreeBSD дээр файл систем агуулж байгаа хуваалтыг *зүсмэл* гэж нэрлэдэг. Хуваалтын ерөнхий нэр нь FreeBSD дээр зүсмэл гэж яригдах бөгөөд FreeBSD-ийн UNIX® гаралд цаад утга учир нь бий. Зүсмэл нь 1 -ээс эхлээд 4 хүртэлх тоогоор дугаарлагдана.

Зүсмэлийн дугаар нь төхөөрөмжийн нэрний араас s үсгээр эхэлж бичигддэг. Тэгэхээр "da0_s1_" гэдэг нь, эхний SCSI диск дээрх эхний зүсмэлийг илэрхийлж байна. Диск дээр физик чанараараа зөвхөн дөрвөн ширхэг зүсмэл байрлуулж болно. Харин логик зүсмэлийг та физик зүсмэл дотор дурын хэмжээтэй үүсгэж болно. Ингэж өргөтгөж нэмсэн зүсмэлүүд нь 5 гэсэн дугаараар эхэлж цаашаагаа тоологддог. Тэгэхээр "ad0_s5_" гэдэг нь эхний IDE диск дээрх эхний өргөтгөсөн зүсмэлийг хэлж байна. Ингэж өргөтгөсөн зүсмэлүүд нь файл систем агуулж, систем дээр ердийн зүсмэл мэт харагдаж ажиллана.

Зүсмэлүүд нь физик диск дээр "аюултай зориулалт"аар буюу өөрөөр хэлбэл хүчээр байрлуулагддаг. Харин бусад дискнүүд нь a -с эхлээд h хүртэл нэрлэсэн *хуваалт* агуулж болдог. Эдгээр үсэгнүүд нь төхөөрөмжийн ард залгагдаж бичигддэг ба "da0_a_" гэдэг нь эхний da диск дээр байгаа а хуваалтыг илэрхийлж байна. "ad1s3_e_" бол хоёр дахь IDE диск дээр байгаа гурав дахь зүсмэлийн тав дахь хуваалтыг илэрхийлж байна.

Эцэст нь хэлэхэд, диск болгон системд танигдах ёстой. Дискний нэр нь дискний төрлийг илэрхийлсэн үсгээр эхлээд тэгээд араас нь хэд дэх диск вэ гэдгийг нь илэрхийлсэн дугаартай байдаг. Зүсмэлээс ялгарах зүйл нь, дискний дугаар 0 -ээс эхэлдэг. Ерөнхий хэрэглээний жишээг Диск төхөөрөмжийн нэрнүүд хүснэгтээс харна уу.

Хуваалт уруу хандах үед FreeBSD уг хуваалтыг агуулсан зүсмэл болон дискийг тодорхойлохыг шаарддаг. Тэгээд зүсмэл рүү хандах үед зүсмэлийг агуулсан дискний нэрийг шаардах болно. Тэгэхээр та дискний нэр, s, зүсмэлийн дугаар, тэгээд хуваалтын үсэг гэсэн дарааллаар нэрлэх нь байна. Жишээнүүдийг Диск, Зүсмэл, Хуваалтын нэрлэх жишээ-д харуулав.

Танд ойлгоход тань дөхөм болгож Дискний тогтсон загвар жишээн дээр дискний зохион байгуулалтын тогтсон загварыг харуулж байна.

FreeBSD суулгахын тулд та эхлээд дискний зүсмэлийг тохируулна, дараа нь зүсмэл дотор FreeBSD-н хэрэглэх хуваалт үүсгээд, дараа нь хуваалт бүрд файл систем (эсвэл swap зайг) үүсгэж эцэст нь хаана холбогдохыг нь (mount) зааж өгдөг.

*Хүснэгт 6. Диск төхөөрөмжийн нэрнүүд*

| Нэр | Утга |
|---|---|
| ad | ATAPI (IDE) disk |
| da | SCSI direct access disk |
| acd | ATAPI (IDE) CDROM |
| cd | SCSI CDROM |
| fd | Floppy disk |

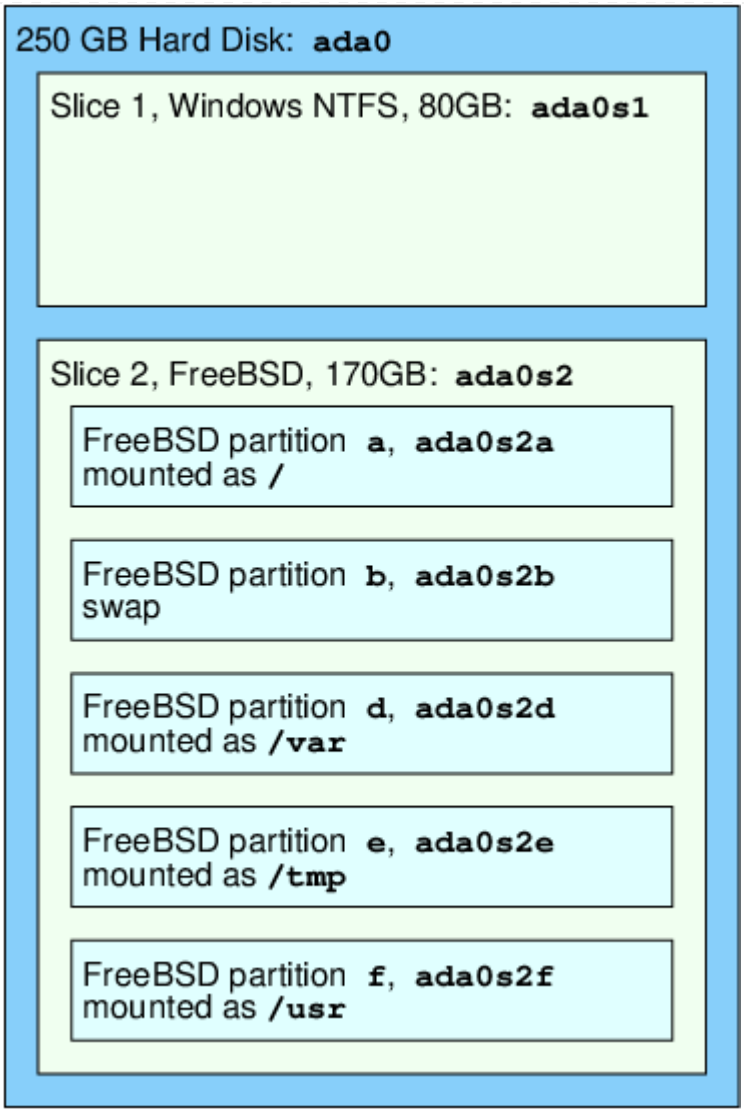
*Жишээ 6. Диск, Зүсмэл, Хуваалтын нэрлэх жишээ*

| Нэр | Утга |
|---|---|
| `ad0s1a` | Эхний IDE диск (`ad0`) дээрх эхний зүсмэлийн (`s1`) эхний хуваалт (`a`). |
| `da1s2e` | Хоёр дахь SCSI диск (`da1`) дээрх хоёр дахь (`s2`) зүсмэлийн тав дахь (`e`) хуваалт. |

*Жишээ 7. Дискний тогтсон загвар*

Дараах загвар нь системд буй IDE диск FreeBSD дээр хэрхэн харагдаж байгааг харуулж байна. Дискний хэмжээг 4 ГБ-н хэмжээтэй гэж үзээд хоёр ширхэг 2 ГБ зүсмэл байна (нэг зүсмэл дээр нь MS-DOS® хуваалт байгаа). Эхний зүсмэл нь MS-DOS®-н C: диск агуулсан, харин хоёр дахь зүсмэл дээр FreeBSD суугдсан. Энэ жишээн дээр FreeBSD нь гурван өгөгдлийн хуваалт мөн swap хуваалт хэрэглэж байна.

Гурван хуваалт нь тус тусдаа файл систем агуулж байгаа. `a` хуваалт root файл системд зориулагдаж, `e` хуваалт /var санд, мөн `f` хуваалт /usr санд тус тус зориулагдсан.




# 4.6. Файл системийг холбох болон салгах

Файл систем нь / гэсэн эхээс эхлэн модлог хэлбэрээр маш сайн харагддаг. /dev, /usr, мөн бусад сангууд нь root буюу эх сангаасаа салбарласан салаа мөчир бөгөөд цаашаа уг мөчир нь бас /usr/local гэж салаалах зэргээр өргөжиж салаалж болдог.

Эдгээр сангуудын заримыг нь өөр файл системд байрлуулах маш олон шалтгаан бий. /var сан нь жишээлбэл log/, spool/, гэх зэрэг янз бүрийн түр зуурын файлуудыг агуулдаг бөгөөд түргэн дүүрэх магадлалтай. Эх сан буюу root файл систем түргэн дүүрнэ гэдэг бол тийм ч сайн юм биш. Тийм болохоор /var санг / сангаас тусад нь өөр газар байрлуулах нь тун хэрэгтэй.

Тодорхой хэдэн сангуудыг тусад нь өөр файл систем дээр байрлуулах хүндтэй шалтгаан бол, хэрвээ уг сангууд нь өөр физик диск дээр, тусдаа виртуал диск дээр, Сүлжээний файл систем дээр, эсвэл CDROM дээр байх явдал юм.

## 4.6.1. fstab файл

/etc/fstab файлд жагсаагдсан файл системүүд нь системийн ачаалах явцад автоматаар холбогддог (гэхдээ хэрэв тэдгээр нь `noauto` сонголт хэрэглээгүй үед).

/etc/fstab нь дараах маягийн жагсаалтыг агуулж байдаг:

```
төхөөрөмж       /холбох-цэг файл-системийн-төрөл     сонголтууд      dumpfreq
passno
```

`төхөөрөмж`

Төхөөрөмжийн нэр (заавал байх ёстой). Төхөөрөмжийн нэрс хуудсанд тайлбарласан байгаа.

`mount-point буюу холбох цэг`

Файл системийг байрлуулах сангийн нэр (заавал байх ёстой).

`файл-системийн-төрөл`

mount(8) хуудсанд тайлбарласан байгаа тул алгасав. FreeBSD-н анхдагч файл систем бол `ufs`.

`сонголтууд`

Уншиж бичигдэх файл системийг заасан `rw` эсвэл зөвхөн уншигдах файл системд зориулсан `ro` сонголт байж болох бөгөөд цаашаагаа мөн нэмэлт сонголтууд агуулж болно. Ердийн сонголт бол `noauto` бөгөөд ингэснээр систем эхлэх үед уг файл систем холбогддоггүй. Бусад сонголтууд mount(8) гарын авлагад тодорхой бичигдсэн байгаа.

`dumpfreq`

Энэ сонголт нь dump(8) хэрэгслээр хэрэглэгддэг бөгөөд аль файл системд нөөц хадгалалт хийх вэ гэдгийг тодорхойлж өгдөг. Хэрэв энэ сонголтыг бичээгүй бол хоосон уттыг агуулж байдаг.

`passno`

Энэ сонголтыг ашиглан ямар дарааллаар файл системүүдийг шалгах вэ гэдгийг зааж өгдөг. Шалгалгүйгээр алгасна гэсэн файл системүүд нь `passno` талбардаа 0 утгатай байх ёстой. root буюу эх файл системийн (энэ файл систем нь хамгийн түрүүнд шалгагддаг) `passno` талбарын утга нь 1 байдаг бөгөөд бусад системийн `passno` нь нэгээс их байх ёстой. Хэрэв нэгээс илүү олон файл системүүд `passno` талбартаа ижил утгатай бол fsck(8) нь тэдгээр файл системийг хэрэв боломжтой зэрэг шалгахыг оролдох болно.

/etc/fstab файлын бүтэц болон тохируулдаг сонголтуудын тухай дэлгэрэнгүй мэдээллийг fstab(5) гарын авлагаас харна уу.

### 4.6.2. `mount` тушаал

mount(8) тушаал нь файл системийг холбоход хэрэглэгддэг цорын ганц тушаал юм.

Таны хамгийн өргөн хэрэглэх хэлбэр бол:

```
# mount төхөөрөмж-буюу-диск холбох-цэг
```

mount(8)-н гарын авлагад зааснаар бол маш олон сонголт байдаг бөгөөд өргөн хэрэглэгддэг

нь:

*Холбох тохируулгууд*

`-a`

/etc/fstab файлд жагсааж бичсэн бүх файл системийг холбоно. Гэхдээ "noauto" гэж тэмдэглэснийг, `-t` гэж туг хатгасныг, эсвэл өмнө нь холбогдчихсон файл системүүдийг холбохгүй.

`-d`

Дуудаж ажиллуулж байгаа файл системээсээ бусдад нь дурын үйлдлээ хийнэ. Энэ сонголтыг `-v` тугтай хамт хэрэглэж mount(8)-г юу хийж байгааг нь харж болдог.

`-f`

Бохир файл систем (аюултай), эсвэл файл системийн унших-бичигдэх төлвөөс зөвхөн-унших төлөв рүү шилжүүлэн бичих эрхийг хасаж холбох үед хүчээр холбодог.

`-r`

Файл системийг зөвхөн-унших төлвөөр холбоно. Энэ нь `-o` сонголтыг `ro` утгатай хэрэглэсэнтэй ижил.

`-t` ***fstype***

Өгөгдсөн файл системийг өгөгдсөн төрлөөр холбоно. Эсвэл `-a` сонголт өгөгдсөн бол зөвхөн өгөгдсөн төрлөөр холбоно.

"ufs" нь анхдагч файл систем юм.

`-u`

Файл системд холболтын сонголтыг шинэчилнэ.

`-v`

Болж байгаа үйл явцыг харуулж мэдэгдэнэ.

`-w`

Файл системийг унших-бичих төлвөөр холбоно.

`-o` сонголт нь дараах зүйлүүдийг таслалаар тусгаарлаж хэрэглэж болно:

**noexec**

Энэ нь файл систем дээр хоёрлосон файл буюу ачаалагдаж ажиллаж болдог файл ажиллахыг үл зөвшөөрнө. Энэ нь мөн аюулгүй байдлын зорилгоор хэрэгтэй.

**nosuid**

Файл систем дээр setuid эсвэл setgid тугуудыг бүү хэрэглэ. Мөн аюулгүй байдлын шалтгаанаар.

### 4.6.3. `umount` тушаал

umount(8) тушаал нь ардаа холбосон цэг, төхөөрөмжийн нэр зэргийг авдаг бөгөөд мөн `-a`

сонголт эсвэл бас `-A` сонголтуудыг авч болдог.

Бүх хэлбэрүүд нь `-f` сонголтыг хүчээр салгах үед болон `-v` сонголтыг явцын мэдээллийг харахад хэрэглэдэг. `-f` сонголт нь тийм сайн санаа биш гэдгийг анхааруулмаар байна. Хүчээр файл системийг салгах нь файл эсвэл системийн эвдрэлд хүргэж болзошгүй.

`-a` болон `-A` сонголтууд нь холбогдсон бүх файл системийг салгахад хэрэглэгддэг. Эсвэл мөн `-t` сонголтоор өөрчлөгдсөн файл системүүдийг салгахад хэрэглэнэ. Гэхдээ `-A` сонголт нь root файл системийг салгах гэж оролддоггүй.

# 4.7. Процесс буюу програмын явц

FreeBSD бол олон үйлдэл зэрэг хийх чадвартай систем. Энэ нь олон програм нэг дор зэрэг ажиллана гэсэн үг. Програмын тухайн үед ажиллаж байгааг нь *процесс буюу програмын явц* гэдэг. Таны ажиллуулсан тушаал болгон шинэ процесс эхлүүлдэг бөгөөд систем дээр бүхэлдээ тэр чигээрээ процессууд үргэлж ажиллаж системийн ажиллагааг бүрэлдүүлж байдаг.

Програмын явц бүр *процесс ID*, эсвэл *PID* гэсэн дугаар агуулж байдаг бөгөөд файлтай бас ижилхэн өөрийн гэсэн эзэмшигч болон бүлэгт харъяалагдаж байдаг. Эзэмшигч ба бүлгийн мэдээлэл нь уг процесс файлд хандах эрхийг нь хэрэглэж файлтай хэрхэн хандах вэ гэдгийг тодорхойлж өгдөг. Ихэнх процесс нь мөн эцэг процесстой байдаг. Эцэг процесс нь тэдгээр процессийг эхлүүлж өгдөг. Жишээлбэл, хэрэв та тушаалуудыг бүрхүүлд бичиж гүйцэтгэвэл, бүрхүүл ч процесс, тушаалууд ч процесс болдог бөгөөд таны бичсэн тушаал болгон процесс болдог. Ийм маягаар ажиллуулаад байвал бүрхүүл нь тэдгээр процессуудын эцэг нь болно. Үүнээс өөр онцгой процесс байдаг ба түүнийг init(8) процесс гэж нэрлэдэг. `init` бол хамгийн эхний процесс байдаг бөгөөд үүний PID дугаар үргэлж 1 байдаг. FreeBSD эхлэх үед `init` процесс автоматаар эхэлдэг.

Системд гүйцэтгэгдэж байгаа процессуудыг хардаг хоёр тушаал бол ps(1) ба top(1) хоёр юм. `ps` тушаал нь одоо ажиллаж байгаа процессуудыг PID дугаартай нь харуулж, мөн хичнээн санах ойн хэмжээ хэрэглэж байгаа, ямар тушаалаар гүйцэтгэгдэж байгаа зэргийг нь харуулдаг. `top` тушаал нь ажиллаж байгаа бүх процессуудыг харуулдаг бөгөөд хэдэн секунд тутамд шинэчлэгдэж байдаг. Ингэснээр таны компьютер тухайн үед юу хийж байгааг харж болно.

Анхдагч горимондоо, `ps` тушаал нь зөвхөн таны эзэмшиж байгаа процессуудыг харуулдаг. Жишээлбэл:

```
% ps
  PID  TT  STAT      TIME COMMAND
  298  p0  Ss     0:01.10 tcsh
 7078  p0  S      2:40.88 xemacs mdoc.xsl (xemacs-21.1.14)
37393  p0  I      0:03.11 xemacs freebsd.dsl (xemacs-21.1.14)
48630  p0  S      2:50.89 /usr/local/lib/netscape-linux/navigator-linux-4.77.bi
48730  p0  IW     0:00.00 (dns helper) (navigator-linux-)
72210  p0  R+     0:00.00 ps
  390  p1  Is     0:01.14 tcsh
 7059  p2  Is+    1:36.18 /usr/local/bin/mutt -y
```

```
 6688  p3  IWs    0:00.00 tcsh
10735  p4  IWs    0:00.00 tcsh
20256  p5  IWs    0:00.00 tcsh
  262  v0  IWs    0:00.00 -tcsh (tcsh)
  270  v0  IW+    0:00.00 /bin/sh /usr/X11R6/bin/startx -- -bpp 16
  280  v0  IW+    0:00.00 xinit /home/nik/.xinitrc -- -bpp 16
  284  v0  IW     0:00.00 /bin/sh /home/nik/.xinitrc
  285  v0  S      0:38.45 /usr/X11R6/bin/sawfish
```

Дээрх жишээнд харуулснаар, ps(1) тушаалын гаралт нь хэд хэдэн баганаар харуулагдаж байна. PID бол өмнө ярьсны дагуу процессийн ID дугаар. PID дугаарууд нь 1 гэж эхлээд 99999 хүртэл дугаарлагддаг бөгөөд хэтрээд ирэхээрээ эхнээсээ эхэлж тоологддог (PID нь ашиглагдаж байгаа бол дахин олгогддоггүй). TT багана нь, уг програмын ажиллаж байгаа tty буюу терминалыг нь харуулдаг бөгөөд одоохондоо хэрэгсэх шаардлага байхгүй. STAT нь програмын төлвийг харуулдаг бөгөөд мөн одоохондоо хэрэгсэхгүй байж болно. TIME нь процессор дээр хэр удаан ажиллаж байгааг нь заадаг бөгөөд гэхдээ энэ нь програм эхэлснээс хойш тооцож эхэлсэн цаг биш. Ихэнх програм нь процессор дээр гүйцэтгэх гэж хэсэг хугацаа зарцуулж хүлээдэг. Эцэст нь, COMMAND нь тухайн програмыг ажиллуулсан тушаалыг харуулдаг.

Харуулах мэдээллээ өөрчилж болдог хэд хэдэн нэмэлт сонголт ps(1) тушаалд бий. Байнга хэрэглэгддэг сонголтуудын багц бол auxww юм. a сонголт нь зөвхөн өөрийн эзэмшдэг процесс биш харин ажиллаж байгаа бүх процессийг харуулдаг. u сонголт нь процессийг эзэмшиж байгаа хэрэглэгчийн нэрийг харуулдаг бөгөөд мөн хэрэглэж байгаа санах ойг нь харуулдаг. x сонголт нь далд ажиллаж байгаа буюу дэмон эсэхийг нь харуулдаг. ww сонголт нь процессуудыг ажиллуулсан тушаалын бүтэн нэрийг нь харуулдаг. Бусад үед, хэрэв тушаалын нэр нь дэлгэцэнд багтахгүй урт байх тохиолдолд хасаж харуулдаг.

top(1) тушаалын гаралт нь өмнөх жишээтэй ижилхэн. Жишээ болгож нэг гаралтыг харцгаая:

```
% top
last pid: 72257;  load averages:  0.13,  0.09,  0.03    up 0+13:38:33  22:39:10
47 processes:  1 running, 46 sleeping
CPU states: 12.6% user,  0.0% nice,  7.8% system,  0.0% interrupt, 79.7% idle
Mem: 36M Active, 5256K Inact, 13M Wired, 6312K Cache, 15M Buf, 408K Free
Swap: 256M Total, 38M Used, 217M Free, 15% Inuse

  PID USERNAME PRI NICE  SIZE    RES STATE    TIME   WCPU    CPU COMMAND
72257 nik       28   0  1960K  1044K RUN      0:00 14.86%  1.42% top
 7078 nik        2   0 15280K 10960K select   2:54  0.88%  0.88% xemacs-21.1.14
  281 nik        2   0 18636K  7112K select   5:36  0.73%  0.73% XF86_SVGA
  296 nik        2   0  3240K  1644K select   0:12  0.05%  0.05% xterm
48630 nik        2   0 29816K  9148K select   3:18  0.00%  0.00% navigator-linu
  175 root       2   0   924K   252K select   1:41  0.00%  0.00% syslogd
 7059 nik        2   0  7260K  4644K poll     1:38  0.00%  0.00% mutt
...
```

Тушаалын гаралт хоёр хэсэгт хуваагдсан байна. Толгой хэсэг (эхний таван мөр) нь сүүлд

ажиллаж байгаа PID дугаарыг, системийн ачаалалтын дунджууд (энэ нь системийг хэр завгүй байгааг илтгэдэг), системийн асаалттай байгаа хугацаа (унтраалгүй хэр удсан эсэхийг) мөн одоогийн цаг зэргийг харуулдаг. Бусад илэрхийллүүд нь, хичнээн процессууд ажиллаж байгаа (энэ тохиолдолд 47 ), хичнээн санах ойн хэмжээ болон swap хэмжээ хэрэглэгдэж байгаа, мөн хичнээн хугацааг процессорын бусад төлөв байдалд зарцуулсан зэргийг илэрхийлж байна.

Доорх багануud нь ps(1) тушаалтай ойролцоо мэдэгдлүүдийг харуулж байна. Эхлээд PID дугаар дараа нь хэрэглэгчийн нэр, процессороос хэрэглэсэн хугацаа, ажиллуулсан тушаал гэх мэт. top(1) тушаал нь анхдагч тохируулгаараа, процессийн хэрэглэж байгаа санах ойн хэмжээг харуулдаг. Тэр хэсэг нь хоёр баганад харуулагддаг бөгөөд эхнийх нь нийт хэмжээ дараагийх нь тухайн үеийнх нь хэмжээ юм. Нийт хэмжээ гэдэг нь програм ажиллахад хэд хэрэгтэйг хэлдэг бөгөөд тухайн үеийнх нь хэмжээ нь одоогоор хэдийг хэрэглэж байгааг заадаг. Энэ жишээн дээр getenv(3) програм бараг 30 МБ хэмжээг RAM санах ойд шаардлагатай гэсэн боловч одоогоор 9 МБ хэмжээ хэрэглэж байна гэж заасан байна.

top(1) нь энэ харуулалтыг хоёр секунд тутам шинэчилдэг бөгөөд үүнийг `s` тохируулгаар өөрчилж болно.

# 4.8. Далд чөтгөр буюу дэмонууд, дохионууд, мөн процессуудыг зогсоох нь

Та хэрэв ямар нэг текст засварлагч дээр ажиллаж байхдаа файл нээх, хаах, хадгалах гэх мэт уг програмыг сайн хянаж чаддаг. Та яагаад тэгж чадаж байна вэ гэвэл, уг програм нь *терминал* дээр холбогдон ажиллаж танд тийм боломж олгож байгаа билээ. Зарим програм тэгэхэд үргэлж хүн гарнаас оруулалт хийх шаардлагагүй зориулалтаар бүтээгдсэн байдаг бөгөөд хамгийн эхний боломж гарангуут терминалаас салангид ажилладаг. Жишээлбэл вэб серверүүд өдөржин хүмүүсээс ирсэн хүсэлтэд хариулт өгч байдаг бөгөөд ердийн үед танаас оруулга шаарддаггүй. Ийм төрлийн бас нэг програм бол захиа илгээгч програм юм.

Бид ийм програмыг *далд чөтгөр буюу дэмон* гэж нэрлэдэг. Дэмонууд нь Грекийн домогт байдаг сайн муугийн аль нь ч биш бөгөөд жижигхэн мөртлөө хүмүүст хэрэгтэй юм хийж байдаг сүнсийг хэлдэг. Вэб сервер болон захианы серверүүд үүнтэй ижил хүмүүст тустай юм хийдэг. Тийм болохоор BSD нь спорт шаахайтай, жижигхэн, сэрээ барьсан чөтгөрийг олон жилээр дуртайяа өөрийнхөө билэг тэмдэг болгож байгаа билээ.

Далд ажиллаж байгаа буюу дэмон болж ажиллаж байгаа програмын нэрний ард "d" үсэг залгаж бичдэг зарчим бий. BIND програмын бүтэн нэр нь Berkeley Internet Name Daemon бөгөөд үндсэн далд ажилладаг програмын нэр нь `named`, мөн Apache вэб серверийн далд ажилладаг програмын нэр нь `httpd`, хэвлэх дарааллыг далд ажиллаж зохицуулж байдаг програмын нэр нь `lpd` гэх мэт нэртэй байдаг. Энэ нь ерөнхийдөө ингэж зарчим гаргасан болохоос хатуу тогтоосон дүрэм биш; жишээлбэл захиа илгээгч үндсэн програм Sendmail-н далд ажилладаг програмыг та `maild` гэж төсөөлж байгаа бол эндүүрэх бөгөөд харин `sendmail` гэж нэрлэдэг.

Заримдаа та эдгээр дэмон процессуудтай холбогдож харилцах хэрэг гарна. Ингэх нэг арга нь түүн рүү (эсвэл бусад ажиллаж байгаа процесс уруу) *дохио* гэгддэг зүйл илгээх явдал юм. Маш олон төрлийн илгээж болох дохионууд байдаг - зарим дохионууд онцгой зориулалттай

нийтэд нь хэрэгждэг, зарим нь тухайн програмдаа зориулж өөр өөрөөр хөрвүүлэгддэг бөгөөд програмын заавар дээр ямар дохиог яаж хөрвүүлэн ойлгох вэ гэдгийг заасан байдаг. Та өөрийнхөө эзэмшиж байгаа процесс уруугаа дохио илгээж болно. Хэрэв та бусдын эзэмшдэг процесс уруу kill(1) эсвэл kill(2) гэх зэрэг дохио илгээвэл таны эрх дутаж хэрэгждэггүй. Гэхдээ ийм эрхээр дутагддаггүй хэрэглэгч бол `root` хэрэглэгч бөгөөд хүн болгоны процесс уруу дохио илгээж чаддаг.

Мөн FreeBSD-ээс програм уруу зарим тохиолдолд дохио илгээдэг. Жишээлбэл, хэрэв муу зохиогдсон нэг програм санах ойг зориулсан хэмжээнээсээ илүү хэрэглээд эхэлбэл FreeBSD уг програм уруу *Хэсгийн буруу хэрэглээ* дохио илгээдэг (`SIGSEGV`). Мөн хэрэв ямар нэгэн програм нь alarm(3) гэдэг системийн сануулах програмыг ашигласан бол тухайн хугацаа нь хэтрэх үед Сэрүүлэг дохио уг програмд ирэх (`SIGALRM`) зэрэг олон дохио бий.

Процесийг зогсоох хоёр дохио байдаг, `SIGTERM` ба `SIGKILL` хоёр. `SIGTERM` нь арай эелдэг аргаар процессийг зогсоодог; процесс нь эхлээд дохиог *хүлээж авааd* өөрийг нь хаах гэж байгаад мэдээд нээлттэй байгаа бүртгэл бичлэг файлуудаа хаагаад тэгээд хийж байгаа ерөнхий үйлдлээ зогсоодог. Зарим тохиолдолд уг процесс нь таслагдаж болохооргүй үйлдэл хийж байх үедээ `SIGTERM` дохиог хэрэгсэхгүй байдал үүсдэг.

`SIGKILL` дохиог ямар ч процесс хэрэгсэхгүй байж чаддаггүй. Өөрөөр хэлбэл энэ нь "Чиний юу хийж байх нь надад хамаагүй, одоо шууд зогсоо" гэсэн дохио юм. Хэрэв та `SIGKILL` дохиог процесс уруу илгээвэл FreeBSD уг процессийг зогсоодог .

Таны хэрэглэж болохоор бусад дохионууд нь `SIGHUP`, `SIGUSR1`, мөн `SIGUSR2`. Эдгээр дохио нь ерөнхий зориулалтаар хэрэглэгддэг бөгөөд эдгээр дохиог хүлээж авсан програмууд тус тусдаа өөр өөр хариу үйлдэл хийдэг.

Жишээлбэл, та вэб серверийнхээ тохируулгын файлд өөрчлөлт хийгээд уг серверийг тохируулгын файлаа дахин шинээр уншуулахыг хүссэн гэж бодъё. Та `httpd` дэмоноо зогсоогоод дахин шинээр эхлүүлж болох боловч ажиллаж байгаа вэб серверийг зогсоож болохгүй нөхцөл байж болно. Ихэнх дэмонууд нь `SIGHUP` дохиог хүлээж авбал өөрийнхөө тохируулгынхаа файлыг уншина гэж тохируулагдсан байдаг. Тэгэхээр `httpd` дэмоноо зогсоогоод шинээр ачаалж байхын оронд `SIGHUP` дохиог илгээхэд хангалттай. Учир нь энэ дохионд тэгж хариулна гэсэн тогтоосон арга зам байхгүй тул дэмон болгон өөр өөр үйлчлэл үзүүлдгийг ойлгож тухайн дэмоны заавар бичгийг нь судлах хэрэгтэй.

дохио илгээхдээ kill(1) тушаалыг ашигладаг.

**Procedure: Процесс уруу дохио илгээх**

Энэ жишээ нь inetd(8) уруу хэрхэн дохио илгээхийг харуулах болно. `inetd` -н тохируулгын файл /etc/inetd.conf гэж байрласан бөгөөд `inetd`-ийг тохируулгын файлаа дахин уншуулахын тулд `SIGHUP` дохиог илгээх болно.

1. Илгээх процессийн процесс ID дугаарыг мэдэх хэрэгтэй. Ингэхийн тулд pgrep(1) тушаалыг ашиглана.

```
% pgrep -l inetd
```

```
  198  inetd -wW
```

За тэгэхээр, inetd(8)-ийн PID дугаар нь 198 гэж энэ тохиолдолд хэлж байна. Зарим тохиолдолд `grep inetd` гэсэн тушаал нь өөрөө энэ гаралтад байж байдаг. Учир нь, ps(1) програм нь уг тушаалыг ажиллуулж байгаа процессийг олсон үед тэгж гаргаж харуулдаг.

2. kill(1) тушаалыг хэрэглэж дохио илгээнэ. inetd(8) нь `root` хэрэглэгчээр гүйцэтгэгдэж байгаа болохоор, та эхлээд su(1) тушаалыг ашиглан `root` хэрэглэгч болох хэрэгтэй.

```
% su
Password:
# /bin/kill -s HUP 198
```

Ихэнх UNIX® системийн тушаалуудын адил, kill(1) тушаал нь хэрэв амжилттай хэрэгжвэл ямар нэгэн төлвийн мэдээлэл харуулдаггүй. Хэрэв та өөрийнхөө эзэмшдэггүй процесс уруу дохио илгээвэл `kill: PID: Operation not permitted` буюу энэ үйлдлийг хийх эрх байхгүй байна гэсэн мэдэгдэл гарч ирнэ. Хэрэв та PID дугаараа буруу бичих, эсвэл буруу процесс уруу дохио илгээвэл, азгүй тохиолдолд уг дохио нь тухайн процесс уруу илгээгдэх бөгөөд, уг процесс нь азаар байхгүй байгаа тохиолдолд `kill: PID: No such process` буюу тийм процесс алга байна гэсэн мэдэгдэл гарч ирнэ.

*Яагаад `/bin/kill` тушаалыг хэрэглэх хэрэгтэй вэ?*

Ихэнх shells буюу бүрхүүлүүд `kill` тушаалыг өөртөө агуулсан байдаг бөгөөд /bin/kill тушаалыг бичихийн оронд шууд бүрхүүлд буй тушаалыг нь гүйцэтгэх нь шулуухан байдаг. Энэ нь амарчилсан ашигтай арга боловч бүрхүүл болгон өөрсдийн илгээх дохионы өөр өөр нэртэй байдгийг мэдэх хэрэгтэй. Тийм болохоор бүрхүүл болгоны дохионы бичлэгийг судлахын оронд шууд `/bin/kill ⋯` тушаалыг хэрэглэх нь зөв арга юм.

Бусад дохио илгээх үйлдлүүд нь үүнтэй тун ижил бөгөөд `TERM` эсвэл `KILL` дохионуудын оронд шаардлагатай дохиогоо бичих хэрэгтэй.

Санаанд орсон тоотой дохио болгоныг устгана гэдэг бол буруу санаа юм. init(8) процесс ялангуяа 1 гэсэн процесс ID байх нь онцгой тохиолдол. Тийм болохоор `/bin/kill -s KILL 1` гэвэл системийг шууд унтраана. kill(1) тушаалыг гүйцэтгэхээсээ өмнө `Return` товч дарахынхаа *өмнө_үргэлж* ямар процесс уруу ямар дохио илгээж байгаагаа давхар шалгах хэрэгтэй.

## 4.9. Shell буюу бүрхүүл

FreeBSD дээр маш олон ажил тушаал бичиж оруулдаг shell буюу бүрхүүл хэмээх орчинд хийгддэг. Бүрхүүлийн гол үүрэг нь гарнаас оруулсан тушаалыг гүйцэтгэх юм. Өдөр болгон

гүйцэтгэхэд шаардагддаг файл зохицуулах, тушаал оруулагч мөрийг засварлах, тушаалын багц, орчны хувьсагч зэрэг туслах тушаалуудыг олон бүрхүүлүүд агуулж байдаг. FreeBSD нь `sh` буюу Bourne Shell, мөн `tcsh` буюу сайжруулсан C-shell зэрэг багц бүрхүүлүүдтэй хамт ирдэг. `zsh` болон `bash` зэрэг маш олон бүрхүүлүүд FreeBSD-ийн портын цуглуулганд ирдэг.

Та ямар бүрхүүл хэрэглэдэг вэ? Энэ бол тухайн хүний дур сонирхолтой хамаатай. Хэрэв та C програмын хэл дээр програм бичдэг бол C-тэй адилхан бичигддэг `tcsh` төрлийн бүрхүүлд дуртай болж болох юм. Хэрэв та Linux системээс ирсэн юм уу эсвэл UNIX®-ийн тушаалуудтай дөнгөж танилцаж байгаа шинэ хүн бол `bash` бүрхүүлээр эхлэх нь амар байж болох юм. Хамгийн гол нь бүрхүүл болгон өөрийн гэсэн онцгой шинж чанартай болохоор өөрийнхөө хүсэл зорилгод тааруулж бүрхүүлээ сонгож дадах нь чухал.

Бүрхүүлийн хамгийн энгийн чадвар бол файлын нэрийн автомат гүйцэтгэл юм. Тушаал юм уу файлын эхний хэдхэн үсгийг бичээд гарын `Tab` товчийг дарахад уг үсгээр эхэлсэн файлын нэрийг танд гүйцээж бичиж өгдөг. Энд нэг жишээ авч үзье. Танд foobar болон foo.bar гэсэн хоёр файл байгаа гэж бодъё. Та foo.bar файлыг устгахыг хүсэв. Тэгвэл та компьютерийнхаа гаран дээр ингэж бичих болно: `rm fo[Tab].[Tab]`.

Бүрхүүл танд ингэж харуулна `rm foo[BEEP].bar`.

[BEEP] гэдэг нь консолоос гаргаж байгаа хонхны дуу бөгөөд ингэж эхэлсэн файлын нэр нэгээс олон байгааг танд мэдэгдэж байгаа юм. foobar болон foo.bar хоёр хоёулаа `fo` гэж эхэлж байгаа ч гэсэн `foo` гэж танд дүүргэж өгдөг. Хэрэв та `.` гэж бичээд `Tab` дахин дарвал бүрхүүл танд уг хүссэн файлын тань нэрийг гүйцээж өгдөг.

Бүрхүүлийн бас нэг шинж чанар бол орчны хувьсагчийн хэрэглээ юм. Орчны хувьсагч гэдэг нь бүрхүүлийн орчинд хадгалагддаг, нэрэнд утга өгсөн хослол байдаг. Энэ бүрхүүлийн орчин нь уг бүрхүүлээс гүйцэтгэгдэж байгаа бүх програм болон програмын тохируулгад харагдаж байдаг. Байнга хэрэглэгддэг ерөнхий орчны хувьсагчдыг дор жагсааж тайлбарлав:

| **Хувьсагч** | **Тодорхойлолт** |
|---|---|
| `USER` | Холбогдсон байгаа тухайн хэрэглэгчийн нэр. |
| `PATH` | Ачаалагдаж болдог хоёртын файлуудыг агуулдаг сангуудыг таслалаар тусгаарлаж бичсэн жагсаалт. |
| `DISPLAY` | Хэрэв холбогдох боломжтой бол, X11 дэлгэцийн сүлжээгээр холбогдох цэг. |
| `SHELL` | Хэрэглэж байгаа shell буюу бүрхүүл. |
| `TERM` | Хэрэглэгчийн терминалын төрлийн нэр. Терминалын шинж чанарыг тодорхойлоход хэрэглэгддэг. |
| `TERMCAP` | Төрөл бүрийн терминалуудад алгасах ёстой кодуудын өгөгдлийн бааз. |
| `OSTYPE` | Үйлдлийн системийн төрөл. жишээ нь, FreeBSD. |

| Хувьсагч | Тодорхойлолт |
|---|---|
| MACHTYPE | Системийн ажиллаж байгаа процессорын архитектур. |
| EDITOR | Хэрэглэгчийн эрхэмлэж хэрэглэгддэг текст засварлагч. |
| PAGER | Хэрэглэгчийн эрхэмлэж хэрэглэдэг пэйжер. |
| MANPATH | Таслалаар тусгаарлаж жагсаасан, гарын авлагын хуудсуудыг агуулсан хайх сангууд. |

Бүрхүүл болгонд орчны хувьсагчаа өөр өөр тохируулдаг. Жишээлбэл, C-маягийн бүрхүүл tcsh болон csh дээр, та setenv тушаалыг ашиглаж орчны хувьсагчийг зааж өгдөг. Bourne төрлийн бүрхүүлүүд болох sh болон bash дээр, та export тушаалыг ашиглаж орчны хувьсагчийг зааж өгнө. Жишээ нь, орчны хувьсагч EDITOR-г өөрчлөх юм уу зааж өгөхийн тулд csh юм уу эсвэл tcsh бүрхүүл дээр EDITOR хувьсагчид /usr/local/bin/emacs утгыг өгөхийн тулд:

```
% setenv EDITOR /usr/local/bin/emacs
```

Bourne бүрхүүлүүд дээр:

```
% export EDITOR="/usr/local/bin/emacs"
```

Ихэнх бүрхүүлийн орчны хувьсагчийн утгыг харахын тулд хувьсагчийн нэрний урд $ тэмдгийг хэрэглэж харна. Жишээ нь, echo $TERM гэсэн тушаал нь $TERM хувьсагчид ямар утга байгааг консол дээр харуулна. Учир нь, бүрхүүл $TERM дотор буй утгыг echo тушаалд дамжуулж өгснөөр консол дээр харуулагддаг.

Бүрхүүл маш олон тэмдэгтийг тусгай зориулалтаар өгөгдлийг илэрхийлэхдээ хэрэглэдэг бөгөөд тэдгээр тэмдэгтийг мета-тэмдэгтүүд гэж нэрлэдэг. Байнга хэрэглэгддэг тэмдэгтийн нэг нь юм. Энэ тэмдэгт нь файлын нэрэнд байгаа тэмдэгтүүдийг хэд ч байсан хамаагүй орохыг илэрхийлдэг. Тэдгээр мета-тэмдэгтүүд нь файлын нэрийг орлуулахад байнга хэрэглэгддэг. Жишээлбэл, echo * гэсэн тушаал нь ls тушаалтай ижилхэн үүрэг гүйцэтгэх бөгөөд, яагаад гэвэл бүрхүүл нь тэмдэгтэд тохирох бүх файлуудыг авч echo тушаал руу дамжуулж харуулдаг.

Бүрхүүл дээр мета-тэмдэгтүүдийг хэрэв тухайн онцгой тохиолдлоор нь биш харин зүгээр тэмдэгт хэлбэрээр нь хэрэглэхийг хүсвэл, уг тэмдэгтийн урд ташуу зураасыг (\) бичиж өгдөг. echo $TERM гэсэн тушаал нь ямар терминал заагдсан байгааг харуулна. Харин echo \$TERM тушаал нь ердөө $TERM гэж харуулна.

### 4.9.1. Өөрийнхөө бүрхүүлийг солих

Бүрхүүлээ буюу shell-ээ солих хамгийн амархан арга бол chsh тушаалыг хэрэглэх юм. chsh тушаал нь таныг EDITOR орчны хувьсагчид заагдсан засварлагч уруу оруулдаг бөгөөд хэрэв энэ хувьсагчид утга заагдаагүй байвал шууд vi засварлагчийг ажиллуулдаг. Тэгээд та "Shell:" мөрөнд буй утгыг өөрчилж бүрхүүлээ өөрчилнө.

Та мөн chsh тушаалыг -s сонголттой хэрэглэж болох бөгөөд ингэсэн үед нэмэлт засварлагч нээлгүйгээр шууд бүрхүүлийг өөрчилдөг. Жишээ нь та өөрийнхөө бүрхүүлийг bash бүрхүүлээр солихыг хүсвэл дараа тушаалыг өгөх болно:

```
% chsh -s /usr/local/bin/bash
```

Таны хэрэглэхийг хүсэж байгаа бүрхүүл тань /etc/shells файл дотор *заавал* байх ёстой. Хэрэв та портын цуглуулгаас бүрхүүл суулгасан бол энэ нь автоматаар хийгдчихдэг. Харин та гар аргаар ямар нэг бүрхүүл суулгавал энэ файлд өөрөө нэмэх хэрэгтэй.

Жишээ нь та bash бүрхүүлийг гар аргаар суулгаад /usr/local/bin байршилд байрлуулсан бол дараах тушаалыг өгөх хэрэгтэй:

```
# echo "/usr/local/bin/bash" >> /etc/shells
```

Тэгээд дараа нь chsh тушаалыг хэрэглэх хэрэгтэй.

## 4.10. Текст засварлагчид

FreeBSD дээр текст файлуудыг засварлаж маш олон тохиргоонуудыг хийдэг. Тийм болохоор та текст засварлагч дээр гаршиж сурах нь чухал. FreeBSD нь үндсэн хэдэн засварлагчтай хамт ирдэг бөгөөд портын цуглуулга дээрээ бол олон зуун засварлагчтай.

Хамгийн амархан бөгөөд сурахад хялбар засварлагч бол ee бөгөөд easy editor буюу хялбар засварлагч гэдгийг товчилсон нэр юм. ee-г эхлүүлэхийн тулд тушаал бичих мөрөндөө ee файлын-нэр гэж бичих бөгөөд *файлын-нэр* нь засварлагдах файлын нэр. Жишээ нь, /etc/rc.conf файлыг засварлахын тулд ee /etc/rc.conf гэж бичнэ. ee засварлагч дотроо дээд хэсэгт нь програмыг хэрэглэх заавар нь бичээстэй байдаг. ^ гэсэн тэмдэг нь гарын Ctrl товчийг илэрхийлдэг бөгөөд ^e гэдэг нь Ctrl + e гэсэн гарын товчлол юм. ee програмаас гарахын тулд Esc товч дараад leave editor буюу засварлагчаас гарна гэдгийг сонгох хэрэгтэй. Хэрэв файлд өөрчлөлт орсон бол гарахаас өмнө хадгалах эсэхийг лавлаж асуух болно.

Мөн FreeBSD нь vi гэсэн хүчирхэг засварлагчийг системийн үндсэн хэсэгтэй цуг зөөвөрлөдөг бөгөөд бас Emacs болон vim зэрэг засварлагчдыг FreeBSD Портын цуглуулгадаа багтаасан байдаг (editors/emacs болон editors/vim). Эдгээр засварлагчид нь ажиллах хүчин чадвараараа илүү боловч сурахад арай илүү төвөгтэй байдаг. Гэвч та текст файлыг засварлахад маш их хөдөлмөр гаргахаар бол vim эсвэл Emacs програмуудыг сурснаар таны цаг болон хөдөлмөрийг цаашид улам илүү хөнгөвчлөх болно.

Файлууд засварладаг эсвэл бичихийг шаарддаг олон програм текст засварлагчийг автоматаар нээдэг. Ашиглагдах анхдагч засварлагчийг өөрчлөхийн тулд EDITOR орчны хувьсагчийг тохируулах хэрэгтэй. Дэлгэрэнгүйг Бүрхүүлүүд хэсгээс үзнэ үү.

# 4.11. Төхөөрөмж ба төхөөрөмжийн цэгүүд

Төхөөрөмж гэдэг ойлголт нь голдуу системд буй төхөөрөмж болох диск, хэвлэгч, график карт, мөн гар зэрэг ордог. FreeBSD эхэлж ачаалах үедээ, гол чухал таньсан төхөөрөмжүүдээ харуулдаг. Ингэж эхлэхдээ харуулсан бичлэгийг та дахин харахыг хүсвэл /var/run/dmesg.boot файлыг хараарай.

Жишээ нь, acd0 гэдэг нь эхний IDE CDROM төхөөрөмж байхад, kbd0 гэдэг нь гарыг илэрхийлж байдаг.

UNIX® үйлдлийн систем нь эдгээр төхөөрөмж уруу хандахдаа төхөөрөмжийн цэг гэж нэрлэгдэх тусгай файл уруу ханддаг. Эдгээр төхөөрөмжийн цэгүүд нь /dev санд байдаг.

## 4.11.1. Төхөөрөмжийн цэг үүсгэх

Хэрэв системд шинэ төхөөрөмж нэмэгдвэл, эсвэл нэмэлт төхөөрөмжид зориулсан шаардлага гарвал шинэ төхөөрөмжийн цэг үүсгэх ёстой.

### 4.11.1.1. DEVFS (DEVice File System буюу төхөөрөмжийн файл систем)

Төхөөрөмжийн файл систем буюу DEVFS нь ерөнхий файлын системийн нэрийн талбарын цөм дахь төхөөрөмжийн нэрийн талбарт хандах боломжийг өгдөг. Төхөөрөмжийн цэгийг үүсгэх эсвэл өөрчлөх зэрэг үйлдлийг DEVFS нь бидэнд хийж өгч амар болгож өгдөг.

devfs(5) гарын авлагаас нэмэлт мэдээллийг харна уу.

# 4.12. Хоёртын хэлбэрүүд

FreeBSD яагаад elf(5) хэлбэр хэрэглэдгийг ойлгохын тулд, та ачаалагдаж ажилладаг файлын төрлөөс UNIX® дээр "ноёлдог" гурван хэлбэрийг мэдэх ёстой:

- a.out(5)

  UNIX®-н хамгийн хуучин бөгөөд "сонгодог" ачаалагддаг файлын хэлбэр. Энэ нь эхэн хэсэгтээ өөрийнхөө хэлбэрийг таниулах зориулалттай шидэт дугаар агуулж байдаг (a.out(5) хуудаснаас дэлгэрэнгүй мэдээлэл авна уу). Ачаалагдсан үедээ санах ойд гурван хэсэгт хуваагддаг: .text, .data, мөн .bss бөгөөд дээрээс нь хэрэглэгдэх обьектуудыг агуулсан хүснэгт мөн мөрийн хүснэгтийг агуулж байдаг.

- COFF

  SVR3 обьект хэлбэр. Толгой хэсэгтээ тодорхой зориулалттай хүснэгт агуулж байдаг. Тийм болохоор зөвхөн .text, .data, болон .bss хэсгүүдээс гадна нэмэлт зүйлс агуулж чадна.

- elf(5)

  COFF-н дараагийн үе. Энэ нь олон хэсэг агуулахаас гадна 32-бит эсвэл 64-битийн утга агуулах чадвартай. Нэг муу тал бий: ELF нь тухайн системийн архитектурт зөвхөн ганцхан ABI байгаа гэж авч үздэг. SYSV ертөнц (хамгийн багадаа гурван ABI агуулж байдаг: SVR4, Solaris, SCO) байсаар байхад ингэж авч үзэх нь буруу юм.

> FreeBSD нь энэ хүндрэлийг, ABI-н мэдээлэл агуулсан ачаалагддаг ELF файлуудыг зохицуулдаг *branding* хэрэгслийг ашиглаж сайжруулахыг боддог. Нэмэлт мэдээллийг brandelf(1) хуудаснаас харна уу.

FreeBSD нь хуучны "сонгодог" отгоос салбарлаж гарсан тул a.out(5) хэлбэрийг хэрэглэж байсан бөгөөд энэ хэлбэрээ 3.X салбар гарах хүртэл маш олон BSD хувилбартаа ашиглаж байжээ. Хэдийгээр FreeBSD дээр өмнө нь ELF хоёртын хэлбэрийг хөрвүүлж мөн ажиллуулж (цөм дээр ч гэсэн) болдог байсан ч, FreeBSD нь анхнаасаа ELF хэлбэрийг анхдагч хэлбэрээ болгохыг "таттгалзсан" билээ. Яагаад? Учир нь, Линукс систем нь хуваалцдаг кодын сан буюу "Shared-Libraries" -д зориулсан үсэрч ажилладаг хүснэгт, мөн түүнийг хөгжүүлэгчид болон байгууллагад хүндрэлтэй байдаг шалтгаанаар a.out хэлбэрээс зайлсхийж ELF хэлбэр рүү шилжих гэж нүсэр хүнд хөдөлмөр зарсан юм. ELF хэлбэр нь хуваалцдаг кодын сан буюу "Shared-Libraries" хүндрэлийг давах боломж олгосон хэрэгслүүдийг санал болгосон бөгөөд тэгээд ч хөгжлийн явцад "нэг алхам урд нь" явж байгааг бодож мөн нэг хэлбэрээс нөгөө хэлбэрт шилжүүлэх үйл явцад гарах хүнд зардал байсан ч шилжүүлэхээр шийдсэн юм. FreeBSD-н кодын санг хуваалцах зарчим нь Sun-н SunOS™ загвартай ижил бөгөөд хэрэглэхэд тун хялбар.

Тэгэхээр, яагаад ийм олон хэлбэр байдаг юм бэ?

Энэ асуултанд хариулахын тулд хуучны, энгийн ажиллах зарчимтай төхөөрөмж хэрэглэж байсан бүүдгэр өнгөрсөн цаг уруу буцацгаая. Энэ энгийн төхөөрөмж нь энгийн жижигхэн систем дээр л ажиллахыг хүснэ. a.out нь (PDP-11) төрлийн иймэрхүү энгийн систем дээр бүгдийг нь хангаж байлаа. Хүмүүс UNIX® системийг ийм энгийн системээс үүсгэсэн болохоор хуучны загвар болох Motorola 68k, VAXen зэрэг системтэй зохицохын тулд a.out хэлбэрийг үлдээсэн юм.

Тэгтэл дараа нь нэг сүрхий инженер хөвүүн, зохиогдсон төхөөрөмжийн зарим ажиллах сул талыг нөхөж процессорыг илүү хурдан ажиллуулах хөнгөхөн програм бичжээ. Энэ програм нь шинэ төрлийн архитектурт (тэр үедээ RISC гэж нэрлэгддэг байсан архитектур) зориулан ажиллахаар бичигдсэн болохоор a.out хэлбэр нь энэ төхөөрөмжид тохиромжгүй болон хангахуйц сайн биш болж ирэв. Тийм болохоор энэ шинэ төхөөрөмжтэй илүү үр дүнтэй ажиллахын тулд илүү олон хэлбэрүүд шинэ загварт зориулж зохиогдож байсан бөгөөд хуучин төрөлд бол энгийн a.out төрлийг санал болгож болох юм. COFF, ECOFF мөн өөр илүү хэд хэдэн хэлбэр нь алдаануудаа нөхөн дэс дараалан үүсгэгдсээр ELF хүртэл хөгжжээ.

Мөн цаашлаад програмын хэмжээ хэдийгээр ихэссэн ч дискний (мөн санах ойн) хэмжээ харьцангуй бага байсан болохоор хуваалцаж болдог кодын сангийн "Shared-Libraries" санаа үүссэн юм. Мөн VM системүүд сайн хөгжиж эхлэв. Хэдийгээр эдгээр сайжруулалт болгон a.out хэлбэрийг хэрэглэж байсан ч, шинэ боломжууд үүсэх тоолонд энэ хэлбэрийг хэрэглэхгүй болж ирэх нь улам ихэссэн билээ. Мөн түүнчлэн, санах ойг хэмнэх үүднээс эхлэн ачаалсны дараа өөр тийшээ үсрэх юм уу эсвэл явцын дунд код нэмэгдэж болох загваруудыг хүмүүс сонирхож эхлэв. Програмын хэлүүд улам сайжирч хүмүүс програмын үндсэн хэсгийг автоматжуулсан код хүсэх болжээ. Энэ бүх боломжуудыг биелүүлэх гэж a.out хэлбэрийг маш их олон янзаар яргалсан бөгөөд хэсэгтээ л энэ нь ажилладаг байв. Нэг мэдэхэд a.out хэлбэр нь ихсэж буй бүх хүндрэлийг зохицуулж чадахааргүй бичлэгийн төвөгтэй болон хэрэглэхэд хэцүү байдалд хүрсэн байна. Хэдийгээр энэ хүндрэлүүдийг ELF хэлбэр нь давдаг боловч шилжих явц нь маш хүндрэлтэй байдаг. Тийм болохоор ELF хэлбэр руу шилжих төвөг нь a.out хэлбэрийг хэрэглэх төвгөөс их байвал ELF хэлбэр нь хүлээгдэхээс

өөр аргагүй болжээ.

Гэвч цаг хугацаа өнгөрсөөр, FreeBSD ба түүний уламжилж гарсан системийн хөрвүүлэх хэрэгсэл нь (ялангуяа ассемблер болон дуудагч буюу loader) хоёр замаар зэрэг хөгжсөөр байв. FreeBSD салаа нь кодын хуваалцдаг санг нэмж мөн зарим алдааг нь залруулсан байна. Үүнийг анх бичсэн GNU-н ард түмэн уг кодоо шинэчилж дахин бичээд янз бүрийн хэлбэрүүдийг нэмж болдог болгоод мөн хөрвүүлэгчээс хамааралгүй хөрвүүлэгддэг болгох зэрэг цааш нь хөгжүүлжээ. Хэдийгээр маш олон хүн FreeBSD дээр хөрвүүлэгчээс хамаарахгүй хөрвүүлэхийг хүссэн боловч FreeBSD-н as болон ld-д зориулсан хуучин кодноос болоод азгүйтжээ. GNU-н шинэ хэрэгслүүд нь (binutils) хөрвүүлэгчээс хамааралгүй, ELF, кодын хуваалцдаг сан, C++ өргөтгөл зэргүүдийг хөрвүүлж чаддаг болжээ. Мөн цаашлаад маш олон байгууллагууд ELF хэлбэртэй хоёртын програмуудыг гаргаж эхэлсэн тул тэдгээрийг хэрэглэхийн тулд FreeBSD уг хэлбэрийг дэмжих нь зөв гэж шийдсэн юм.

ELF хэлбэр нь a.out хэлбэрийг бодвол илүү өргөн хүрээтэй бөгөөд үндсэн системийг илүү өргөжүүлдэг. ELF хэрэгслүүд нь маш сайн зохион байгуулагдсан бөгөөд хөрвүүлэгчээс хамаардаггүй болохоор хүмүүсийн хүсэлд яг тохирдог. ELF нь a.out хэлбэрийг бодвол жаахан удаан байж болох боловч үүнийг хэмжиж тодорхойлно гэдэг нь хэцүү билээ. Мөн энэ хоёрыг санах ойд хуудас зохицуулах, эхлэн ажиллах зарчим зэргийг нь харьцуулсан маш олон шинж чанарууд байдаг. Тэдгээр шинж чанарууд нь тийм ч чухал биш бөгөөд энэ нь зөвхөн ялгаа нь билээ. Одоо бол a.out хэлбэр нь GENERIC цөмөөс хасагдсан бөгөөд a.out хэлбэрийг ажиллуулдаг байсан цөм нь хуучны цөмд тооцогдоно.

# 4.13. Нэмэлт мэдээлэл олж авах нь

## 4.13.1. Гарын авлага

Ихэнх дэлгэрэнгүй мэдээллүүд нь FreeBSD дээр гарын авлага хэлбэрээр оршиж байдаг. Систем дээр ажиллаж байгаа бараг бүх програмууд нь ажиллах болон авдаг шинж чанараа тодорхойлсон товч заавар буюу гарын авлагатай хамт ирдэг. Тийм гарын авлагыг `man` тушаалаар харна. `man` тушаалын хэрэглээ нь тун хялбар:

```
% man тушаал
```

`тушаал` нь судалж уншихыг хүссэн тушаалын нэр байх ёстой. Жишээлбэл `ls` тушаалын тухай мэдээлэл харахыг хүсвэл:

```
% man ls
```

Гарын авлага нь дотроо дараах хэсгүүдэд дугаарлагдаж хуваагддаг:

1. Хэрэглэгчийн тушаал.
2. Системийн дуудлага болон алдааны дугаар.
3. C програмын хэлний санд байрлах функц нь.
4. Төхөөрөмжийн драйвер.

5. Файлын хэлбэр.
6. Тоглоом болон бусад салбар.
7. Төрөл бүрийн бусад мэдээлэл.
8. Системээс санаа тавьж үйлдэх тушаал.
9. Цөм хөгжүүлэгч.

Зарим тохиолдолд гарын авлагын зарим бүлэг нь саяны хуваасан хэсэгт бүрд ижил байдаг. Жишээлбэл `chmod` тушаалыг хэрэглэгч нэг янзаар хэрэглэж байхад систем бас `chmod()` тушаалыг өөр зорилгоор хэрэглэдэг. Энэ тохиолдолд та системд аль сэдвээ сонгож байгаагаа ойлгуулахын тулд харгалзах дугаарыг нь өгөх ёстой:

```
% man 1 chmod
```

Энэ тохиолдолд `chmod` тушаалыг хэрэглэгч яаж дуудаж хэрэглэх тухай харуулна. Гарын авлагын тухайн хэсгийг нь харахын тулд голдуу хаалт дотор тухайн хэсгийн дугаарыг нь бичиж ханддаг. Тэгэхээр chmod(1) гэвэл хэрэглэгчид хамаатай хэсэг нь, chmod(2) гэвэл системд хамаатай хэсэг харуулагдана.

Хэрэв та тушаалынхаа нэрийг мэдэж байвал энэ аргаар маш амархан хэрэглэх зааврыг уншиж чадахаар боллоо. Гэтэл та тушаалынхаа нэрийг мэдэхгүй тохиолдолд яах вэ? Энэ үед та `man` тушаалд тухайн хэрэгтэй тушаалын зааварт хайх түлхүүр үгийг `-k` сонголт ашиглан зааж өгч болдог. :

```
% man -k mail
```

Энэ тохиолдолд, заавартаа "mail" гэдэг үг агуулсан тушаалуудыг жагсааж танд харуулна. Энэ арга нь үндсэндээ `apropos` тушаалын үүрэгтэй ижил болно.

За тэгэхээр, таны /usr/bin санд маш их олон тушаалууд байгааг та мэддэг мөртлөө ямар үйлдэл хийдгийг нь сайн мэдэхгүй тохиолдолд яах вэ? Хамгийн амархан арга бол:

```
% cd /usr/bin
% man -f *
```

эсвэл

```
% cd /usr/bin
% whatis *
```

энэ хоёр хоёулаа ижилхэн үйлдэл хийдэг.

### 4.13.2. GNU Info файлууд

FreeBSD нь Free Software Foundation (FSF) буюу Чөлөөт Програмын Сангаас бүтээсэн маш олон програмуудыг агуулж байдаг. Гарын авлага хуудаснаас гадна эдгээр програмууд нь мөн нэмэлт текст загвартай `info` файл агуулж байдаг бөгөөд уг төрлийн мэдээлэл нь `info` тушаалаар харуулагддаг. Хэрэв та emacs-г суулгасан бол emacs-н info горимд бас харж болно.

info(1) тушаалыг хэрэглэхийн тулд ердөө:

```
% info тушаал
```

Товч тайлбарыг нь харахын тулд `h` дарна. Тушаалын түргэн зааврыг харахын тулд `?` гэж дараарай.

# Chapter 5. Програм суулгах: Багцууд болон портууд

## 5.1. Ерөнхий агуулга

Маш олон системийн хэрэгслүүд FreeBSD-н үндсэн системтэй нь хамт суугддаг. Гэхдээ зарим хүмүүст ажлаа гүйцээхийн тулд маш олон гуравдагчдын бэлтгэсэн програмыг суулгах шаардлага гардаг билээ. Таны систем дээр гуравдагчийн бүтээсэн програмуудыг суулгах хоёр үндсэн маш сайн аргыг FreeBSD хангаж өгдөг нь: FreeBSD-н портын цуглуулга (эх бичлэгээс нь суулгахад зориулагдсан), ба багцууд (урьдчилан хөрвүүлсэн хоёртын хэлбэрийн файлаас суулгахад зориулагдсан) юм. Энэ хоёр арга нь хоёулаа таны суулгахыг хүссэн програмын хамгийн сүүлийн хувилбарыг сүлжээ юм уу дотоод төхөөрөмжөөс суулгахад хэрэглэгдэнэ.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Гуравдагчдын бүтээсэн програмын багцыг хэрхэн суулгах.
- Портын цуглуулга ашиглаж гуравдагчдын програмыг эх бичлэгээс нь хэрхэн бүтээх.
- Өмнө суугдсан багц юм уу эсвэл портуудыг хэрхэн устгах.
- Портын цуглуулгын хэрэглэдэг анхдагч утгуудыг хэрхэн дарж бичих.
- Шаардлагатай програмын багцаа хэрхэн хайж олох.
- Програмаа хэрхэн шинэчилсэн түвшинд авчрах.

## 5.2. Програм суулгацын ерөнхий ойлголт

Хэрэв та урьд нь UNIX® системийг хэрэглэж байсан бол, гуравдагчдын бүтээсэн програмыг суулгах явц дараах маягаар явагддаг билээ:

1. Програмыг татаж авах. Энэ нь голдуу эх бичлэг хэлбэрээрээ юм уу эсвэл хөрвүүлэгдсэн хоёрт файлын хэлбэрээр байдаг.
2. Авсан програмаа задлах (ихэнх нь compress(1), gzip(1), эсвэл bzip2(1) нараар шахагдсан байдаг).
3. Бичиг баримт нь хаана байгааг олоод (голдуу INSTALL юм уу README файлууд байдаг ба, эсвэл doc/ гэсэн дэд санд буй бичиг баримтууд) хэрхэн суулгах зааврыг нь унших.
4. Хэрэв програм нь эх бичлэг хэлбэрээрээ байгаа бол, түүнийг хөрвүүлэх. Ингэхийн тулд магадгүй Makefile файлыг засварлах, эсвэл `configure` скриптийг ажиллуулах зэрэг ажлууд байдаг.
5. Програмыг шалгаад, дараа нь суулгах.

Хэрэв бүх зүйл сайхан бүтсэн үед л ийм явц үйлдэгддэг. Хэрэв та тусад нь FreeBSD-д зориулж

порт болгоогүй програмыг суулгахыг хүсвэл, магадгүй та уг програмыг ажиллуулахын тулд тухайн програмын эх кодыг засварлах хэрэг гарна.

Хэрэв та хүсэж байгаа бол, иймэрхүү "уламжлалт" аргаар програмыг FreeBSD дээр суулгаж болно. Гэхдээ FreeBSD дээр таны нөр их хөдөлмөрийг хөнгөлөх хоёр арга байдаг нь: багцууд болон портууд юм. Энэ баримтжуулалтыг бэлтгэж байх үед 36000 гаран гуравдагчийн бэлтгэсэн програмууд ийм аргаар бэлэн болсон байгаа.

Ямар ч програм байсан, FreeBSD дээрх уг програмын багц гэж ердөө нэг л файлыг та татаж авах ёстой байдаг. Уг багц дотор тухайн програмын урьдчилан хөрвүүлсэн хувилбар, бичиг баримт болон тохируулга хийх скриптүүд зэрэг нь багтсан байгаа. Татаж авагдсан багц файлыг pkg_add(1), pkg_delete(1), pkg_info(1) зэрэг FreeBSD-н багц зохицуулагч програмаар өөрчилж болдог. Шинэ програм суулгах явцыг нэг л тушаалаар гүйцэтгэхийг эрмэлзсэн байдаг билээ.

Програмуудын FreeBSD порт гэдэг нь, тухайн програмуудыг эх бичлэгээс хөрвүүлж автоматаар суулгах зориулалттай бүлэглэгдсэн файлуудыг хэлдэг.

Програмыг эх бичлэгээс нь хөрвүүлж суулгахад хэд хэдэн үйлдэл (татаж авах, задлах, нөхөх, хөрвүүлэх, суулгах) хийдгийг та санаж байгаа байх. Портын агуулж буй файлууд нь яг энэ явцыг танд зориулж автоматжуулахад зориулагдсан мэдээлэл агуулсан байдаг. Та энгийн хэдэн тушаалыг гарнаас оруулахад л, уг програм татагдаж авагдаад, задлагдан, нөхөгдөж, хөрвүүлэгдээд суугдах болно.

Яг үнэн хэрэгтээ, портлох арга нь тухайн програмыг дараа нь `pkg_add` болон бусад багц зохицуулагч програмуудад хэрэглэгдэж болохоор багц үүсгэж хэрэглэгдэж бас болдог.

Багц болон портлох энэ хоёр арга нь хоёулаа, тухайн програмын *dependencies буюу хамаатан* програмуудыг мэдэж байдаг. Та нэг програм суулгах шаардлагатай байгаа бөгөөд уг програм нь бас нэг програмын сан файлыг суугдсан байхыг шаарддаг байна хэмээн авч үзье. Уг хоёр програм хоёулаа FreeBSD-н порт ба багц хэлбэрээр танд байгаа гэж бодъё. Хэрэв та `pkg_add` тушаалыг хэрэглэх юм уу эсвэл портын аргыг хэрэглэж уг програмыг суулганэ гэвэл, энэ хоёр арга нь хоёулаа уг програмыг суулгахад шаардагдах сан файл суугдаагүй байвал суугдаагүй байна гэж мэдээлээд, тэр сан файлыг эхлээд суулгадаг.

Энэ хоёр аргыг харах юм бол хоорондоо тун ижилхэн үйлдэл хийдгийг та ажиглаж магадгүй юм. Тэгээд яагаад FreeBSD гуай энэ хоёр аргатай хоёулантай нь зууралдаад байгаа юм бол гэж гайхаж мададгүй. Багц болон порт хоёр нь таны суулгах нөхцлөөс шалтгаалан өөрсдийн гэсэн хүчирхэг давуу талтай.

*Багцын давуу тал*

- Програмын шахсан багц файл нь уг програмын эх бичлэгийг агуулдаггүй болохоор хэмжээний хувьд бага байдаг.
- Багцууд нэмэлт хөрвүүлэх үйлдэл шаарддаггүй. Mozilla, KDE, эсвэл GNOME зэрэг том програмуудыг удаавтар систем дээр суулгахаар бол, энэ арга нь тун их давуу талтай юм.
- Багцууд нь FreeBSD дээрх ямар нэгэн хөрвүүлэх явцын тухай мэдээлэл шаарддаггүй.

*Портын давуу талууд*

- Багцыг аль болох олон систем дээр тохирогдон суулгах гэж хичээсэн болохоор, ийм багц

нь маш өргөн дэлгэр сонголттойгоор хөрвүүлэгдэж бэлтгэгдсэн байдаг. Порт хэрэглэн програм суулгахаар бол, (жишээ нь) Pentium 4 эсвэл Athlon процессордоо зориулж програмын тохируулгыг өөрчилж болдог.

- Зарим програмууд өөрийнхөө чадах болон чадахгүй зэрэг нөхцлөөсөө хамаараад хөрвүүлэгдэх явцад өөрчлөлт хийгдэх тохиолдлууд байдаг. Жишээ нь, Apache програм нь маш олон төрлийн өөртөө агуулсан тохиргоотой ирдэг. Ийм програмыг та портоос бүтээх үедээ анхдагч тохиргоонуудыг нь өөрчилж өөртөө тохируулж болно.

  Заримдаа, ижил төрлийн програмууд өөрсдийнхөө онцгой тохиргооноос хамаараад хэдэн хэдэн янзаар хөрвүүлэгдэж багцлагдсан байдаг. Жишээ нь, X11 сервер суусан эсэхээс хамаараад Ghostscript програм нь ghostscript гэсэн нэртэй багцлагдсан байдаг ба бас ghostscript-nox11 гэсэн нэрээр мөн багц хэлбэртэй байдаг. Багцлах аргад иймэрхүү явцуу тохируулгаас хамаарсан арга байдаг боловч, хэрэв хөрвүүлэх үеийн тохиргоо нь ихсэх болбол энэ арга нь тун явуургүй болдог билээ.

- Зарим програмын лицензэнд, уг програмыг хөрвүүлэгдсэн хэлбэрээр түгээхийг хорьсон байдаг. Тийм програмуудыг эх бичлэг хэлбэрээр нь түгээх ёстой болдог билээ.
- Зарим хүмүүс хөрвүүлэгдсэн хоёртын файлын түгээлтэнд дургүй байдаг. Ядаж эх бичлэг нь байвал, та (онолын хувьд) эх бичлэгийг уншиж болзошгүй аюулыг илрүүлж болох юм.
- Хэрэв танд нөхөлт бичлэг (засвар) байгаа бол, та уг нөхөлтийг зөвхөн эх бичлэг дээр л хийж чадна.
- Зарим хүмүүс эх бичлэгээр наадах дуртай байдаг. Тэд нар эх бичлэгийг уншаад залхуу нь хүрээд ирэхээрээ уг бичлэгийг өөрчилж эвдэх, эсвэл зээлдэж өөрчлөх ( мэдээж лицензийн дагуу ) гэх мэтээр оролддог.

Портын шинэчлэлийн тухай мэдээлэл авч байхыг хүсвэл FreeBSD портын захидлын жагсаалт болон FreeBSD портын алдааны захидлын жагсаалт зэрэг захианы жагсаалтад бүртгүүлэх хэрэгтэй.

Ямар нэгэн програм суулгахаасаа өмнө http://vuxml.freebsd.org/-с програмын нууцлалын тухай мэдээллийг харах нь зүйтэй.

Та мөн ports-mgmt/portaudit-г суулгаж болох бөгөөд энэ нь, програмыг суулгахын өмнө уг програмын хуучралт болон аюулгүй байдалтай хамаатай шаардлагатай шалгалтуудыг хийж өгдөг. Иймэрхүү шалгалтыг зарим багц суулгасны дараа та `portaudit -F -a` гэсэн тушаал хэрэглэн үйлдэж болно.

Энэ бүлгийн үлдсэн хэсэгт FreeBSD дээр порт юм уу багц ашиглан хэрхэн програм суулгах талаар өгүүлэх болно.

# 5.3. Програмаа олох нь

Ямар нэгэн програм суулгахаасаа өмнө та ямар програм суулгахыг хүсэж байгаа болон ямар нэртэй гэдгийг нь мэдсэн байх хэрэгтэй.

FreeBSD-н програмын жагсаалт нь цаг тутамд нэмэгдэж байдаг. Аз болж таны хүссэн

програмыг олох хэд хэдэн арга бий:

- FreeBSD-н вэб хуудас нь http://www.FreeBSD.org/ports/ хаяг дээр бүх байгаа програмын шинэчлэгдсэн хувилбаруудыг нь хайж болох хэлбэрээр арчилж байдаг. Портууд нь төрөлжиж хуваагдсан байдаг бөгөөд та хүссэн програмаа хайж олох (хэрэв та нэрийг нь мэдэж байвал) эсвэл тухайн төрөлд байгаа бүх програмуудын нэрсийн жагсаалтыг харж болно. *

  Dan Langille гэгч нь FreshPorts хуудсыг http://www.FreshPorts.org/ хаяг дээр арчлан эрхэлдэг. FreshPorts нь порт дээр гарсан өөрчлөлтүүдийг хянаж байдаг бөгөөд хэрэв таныг хүсвэл таны сонгосон хэд хэдэн портуудыг "ажиглаж" байгаад өөрчлөлт гарангуут танд захиагаар мэдэгддэг. *

  Хэрэв та програмынхаа нэрийг нь мэдэхгүй байгаа бол Freecode ( http://www.freecode.com/) хуудсан дээр хайж үзэх хэрэгтэй. Хэрэв уг хуудсан дээр шинэ програм олдсон бол буцаад FreeBSD хуудсанд очиж уг хуудсыг порт болсон эсэхийг магадлах нь зүйтэй.

- Хэрэв та портынхоо нэрийг яг мэддэг боловч ямар төрөлд багтдагийг нь мэдэх хэрэгтэй бол whereis(1) тушаалыг ашиглаарай. Ердөө *л* `whereis файлын_нэр` гэж бичих бөгөөд *файлын_нэр* нь суулгахыг хүссэн програмын нэр билээ. Хэрэв уг програм нь олдвол, танд хаана байгааг нь дараах маягаар харуулна:

  ```
  # whereis lsof
  lsof: /usr/ports/sysutils/lsof
  ```

  Энэ нь бидэнд `lsof` (системийн нэгэн хэрэгсэл) програмыг /usr/ports/sysutils/lsof санд буй гэж мэдэгдэж байна.

- Мөн портын модонд тухайн порт хаана байгааг олохын тулд ердийн echo(1) тушаалыг ашиглаж болно. Жишээ нь:

  ```
  # echo /usr/ports/*/*lsof*
  /usr/ports/sysutils/lsof
  ```

  Энэ нь /usr/ports/distfiles сан уруу татаж авагдсан таарсан файлуудыг харуулах болно гэдгийг анхаараарай.

- Хүссэн портоо олох бас нэг арга бол портын цуглуулга дотор нь байдаг өөрийнх нь хайгчийг нь ашиглах билээ. Уг хайгчийг нь хэрэглэхийн тулд та /usr/ports сан дотор байх шаардлагатай. Уг сан дотроос `make search name=програмын-нэр` гэж бичих бөгөөд *програмын-нэр* нь таны олохыг хүссэн програмын нэр байх ёстой. Жишээлбэл `lsof` програмыг олохын тулд:

  ```
  # cd /usr/ports
  # make search name=lsof
  Port:   lsof-4.56.4
  ```

```
Path:   /usr/ports/sysutils/lsof
Info:   Lists information about open files (similar to fstat(1))
Maint:  obrien@FreeBSD.org
Index:  sysutils
B-deps:
R-deps:
```

Уг хайлтын үр дүнд та "Path:" гэсэн мөрөнд анхаарлаа хандуулах хэрэгтэй бөгөөд энэ нь хаана байгааг илэрхийлж байдаг. Бусад харуулж байгаа мэдээллүүд нь програм суулгахад хэрэглэгддэггүй болохоор энд дэлгэн ярихгүй.

Мөн портын өөр нэг хайлт хийх арга нь `quicksearch` боломж юм. Энэ боломж нь `search`-н нэгэн адил параметрийг авдаг. Жишээ нь `lsof`-г хайхад дараах үр дүнг харуулна:

```
# cd /usr/ports
# make quicksearch name=lsof
Port:   lsof-4.87.a,7
Path:   /usr/ports/sysutils/lsof
Info:   Lists information about open files (similar to fstat(1))
```

Бүр нарийвчилан хайхыг хүсвэл `make search key=хайх-мөр` юм уу эсвэл `make quicksearch key=хайх-мөр` гэх бөгөөд *хайх-мөр* нь тухайн програмтай холбоотой мөр байдаг. Энэ нь портын нэр, тайлбар, тодорхойлолт болон хамааралтай програм зэргүүдэд хайлт хийдэг болохоор тухайн програмынхаа талаар дэлгэрэнгүй мэдээлэл байхгүй хүнд хайхад тун тохиромжтой.

Дээрх хоёр тохиолдолд (`search` ба `quicksearch`) хайх мөр тань жижиг болон том бичсэнээс хамаардаггүй. "LSOF" гэж хайх нь "lsof" гэж хайсантай ижилхэн үр дүнд хүргэнэ.

## 5.4. Багцалсан системийг хэрэглэх нь

FreeBSD дээр багцуудыг удирдах хэд хэдэн төрлийн хэрэгслүүд байдаг:

- Ажиллаж байгаа систем дээр боломжтой байгаа болон суулгагдсан багцуудыг харах, устгах, суулгахын тулд `sysinstall`-ийг ажиллуулж болно. Дэлгэрэнгүй мэдээллийг Програмын багц суулгах-с үзнэ үү.
- Багцын удирдлагын тушаалын мөрийн хэрэгслүүдийг энэ хэсгийн үлдсэн хэсэгт хэлэлцэх болно.

### 5.4.1. Багц суулгах

Та pkg_add(1) хэрэгслийг ашиглан өөртөө хадгалагдсан юм уу эсвэл сүлжээнд буй серверээс FreeBSD-н програмын багц суулгаж болдог.

*Жишээ 8. Багц татаж аваад суулгах явц*

```
# ftp -a ftp2.FreeBSD.org
```

```
Connected to ftp2.FreeBSD.org.
220 ftp2.FreeBSD.org FTP server (Version 6.00LS) ready.
331 Guest login ok, send your email address as password.
230-
230-     This machine is in Vienna, VA, USA, hosted by Verio.
230-         Questions? E-mail freebsd@vienna.verio.net.
230-
230-
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /pub/FreeBSD/ports/packages/sysutils/
250 CWD command successful.
ftp> get lsof-4.56.4.tgz
local: lsof-4.56.4.tgz remote: lsof-4.56.4.tgz
200 PORT command successful.
150 Opening BINARY mode data connection for 'lsof-4.56.4.tgz' (92375 bytes).
100% |***************************************************| 92375       00:00 ETA
226 Transfer complete.
92375 bytes received in 5.60 seconds (16.11 KB/s)
ftp> exit
# pkg_add lsof-4.56.4.tgz
```

Хэрэв танд багц суулгах дотоод эх үүсвэр ( FreeBSD CD-ROM гэх мэт) байхгүй бол pkg_add(1) хэрэгслийг -r сонголттой хамт хэрэглэх нь зүйтэй. Энэ нь тухайн програмыг төрөл болон хувилбар зэрэг мэдээллийг нь автоматаар таньж мэдээд FTP хуудаснаас татан авч суулгадаг.

```
# pkg_add -r lsof
```

Дээрх жишээн дээр уг хэрэгсэл нь хэрэглэгчийн оролцоогүйгээр татаж аваад суулгаж буйг харуулж байна. Хэрэв та татаж авах алс хаягийг нь адилхан толин тусгал болох өөр нэг FreeBSD багцын хаягаар солихыг хүсвэл PACKAGESITE орчны хувьсагчийн утгад анхных нь утгыг нь дарж өөр хаяг өгөх хэрэгтэй. pkg_add(1) нь fetch(3)-г ашиглаж файл татаж авдаг бөгөөд янз бүрийн орчны хувьсагчийн утга хэрэглэдэг. Тэд нар нь FTP_PASSIVE_MODE, FTP_PROXY, мөн FTP_PASSWORD гэх мэт хувьсагчид байдаг. Хэрэв та галт хананы цаана байгаа юм уу эсвэл FTP/HTTP прокси хэрэглэж байгаа бол уг хувьсагчдыг өөрчлөх ёстой. Хувьсагчдын бүрэн жагсаалтыг fetch(3) хуудаснаас харна уу. Дээрх жишээн дээр lsof нь lsof-4.56.4 оронд хэрэглэгдэв. Хэрэв та алсаас автоматаар татаж авах үйлдэл хийж байгаа бол багцны хувилбарын дугаарыг оруулалгүй бичих хэрэгтэй. pkg_add(1) нь тухайн програмын сүүлийн хувилбарыг автоматаар таньж суулгадаг.

Хэрэв та FreeBSD-CURRENT эсвэл FreeBSD-STABLE хувилбарыг хэрэглэж байгаа бол pkg_add(1) нь програмын хамгийн сүүлийн хувилбарыг татаж авч суулгадаг. Хэрэв та -RELEASE хувилбар хэрэглэж байгаа бол таны хэрэглэж байгаа тухайн хувилбарт тохирсон хувилбарыг нь татаж авч суулгадаг. Гэхдээ иймэрхүү үйлдлийг нь өөрчлөхийг хүсвэл PACKAGESITE-г өөрчлөөрэй. Жишээлбэл та FreeBSD 8.1-RELEASE систем дээр pkg_add(1)

> хэрэгслийг ажиллуулбал анхдагч хаяг нь ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-8.1-release/Latest/ байх болно. Хэрэв та pkg_add(1)-г FreeBSD 8-STABLE багц суулгахаар тохируулахыг хүсвэл PACKAGESITE хувьсагчийг ftp://ftp.freebsd.org/pub/FreeBSD/ports/i386/packages-8-stable/Latest/ гэж зааж өгөх хэрэгтэй.

Багц файлууд нь .tgz болон .tbz гэсэн хэлбэртэй байдаг. Та тэдгээрийг ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/packages/ хаягнаас олж болох бөгөөд эсвэл FreeBSD CD-ROM дээр бас буй. FreeBSD-н 4 ширхэг CD тус бүрийн /packages сан дотор багц байрлаж байгаа. Багц байрлуулалт нь /usr/ports-н загвартай ижил модлог байдаг. Төрөл болгон өөрийн гэсэн сантай ба багц болгон бас All буюу бүгд гэсэн сангаас олддог.

Багцны сангийн бүтэц нь портын бүтэцтэй таарч багц портын системийг бүрэлдүүлдэг юм.

### 5.4.2. Багц зохицуулалт

pkg_info(1) хэрэгсэл нь суугдсан багцтай холбоотой мэдээллүүдийг харуулдаг.

```
# pkg_info
colordiff-1.0.13    A tool to colorize diff output
docbook-1.2         Meta-port for the different versions of the DocBook DTD
...
```

pkg_version(1) хэрэгсэл нь суугдсан бүх багцны ерөнхий мэдээллийг харуулж байдаг. Энэ нь багцны хувилбарыг портын санд буй програмын хувилбартай жишиж харьцуулдаг.

```
# pkg_version
colordiff                   =
docbook                     =
...
```

Хоёр дахь баганад буй тэмдэгт нь порт доторх програмын хувилбартай харьцуулсан харьцуулалт юм.

| Тэмдэг | Утга |
|---|---|
| = | Суугдсан багцны хувилбар нь портын хувилбартай адилхан байна. |
| < | Суугдсан хувилбар нь портын хувилбараас хуучин байна. |
| > | Суугдсан хувилбар нь портын хувилбараас шинэ байна. (Дотоод порт тань хуучирсан байж болзошгүй) |

| Тэмдэг | Утга |
| --- | --- |
| ? | Суугдсан багц нь портын жагсаалтад олдсонгүй. (Энэ тохиолдож болох юм. Жишээлбэл суугдсан порт нь устгагдах юм уу нэр нь өөрчлөгдсөн байж болзошгүй.) |
| * | Багц олон хувилбартай байна. |
| ! | Суулгагдсан багц жагсаалтад байгаа боловч ямар нэг шалтгааны улмаас `pkg_version` тушаал нь суулгагдсан багцын хувилбарын дугаарыг жагсаалт дахь харгалзах оруулгатай харьцуулж чадахгүй байна. |

### 5.4.3. Багц устгах

Өмнө суугдсан байгаа багцыг устгахдаа pkg_delete(1) хэрэгслийг ашиглана.

```
# pkg_delete xchat-1.7.1
```

pkg_delete(1) нь багцын бүрэн нэр болон дугаарыг шаарддагийг тэмдэглэе; *xchat-1.7.1*-ийн оронд *xchat* өгөгдсөн бол дээр дурдсан тушаал ажиллахгүй. Гэхдээ суулгагдсан багцын хувилбарыг олохын тулд pkg_version(1)-ийг ашиглах хялбар байдаг. Ингэхийн оронд та бүгдийг орлуулах тэмдэгт ашиглаж болох юм:

```
# pkg_delete xchat\*
```

Энэ тохиолдолд `xchat`-аар эхэлсэн нэртэй бүх багцуудыг устгах болно.

### 5.4.4. Бусад мэдүүштэй зүйлс

Бүх багцтай холбоотой мэдээллүүд /var/db/pkg сан дотор хадгалагдаж байдаг. Суугдсан багцны жагсаалт болон холбогдох тодорхойлолтууд нь уг санд буй файл дотроос уншигдах боломжтой.

## 5.5. pkgng-г бинар багцыг удирдахад ашиглах нь

pkgng нь FreeBSD-н уламжлалт багц удирдах хэрэгсэл болох pkg_install-г сайжруулж сольсон хувилбар юм. Энэ нь бинар багцтай хурдан бөгөөд хялбараар ажиллах боломжийг олгодог олон боломжуудтай юм. pkgng-н эхний хувилбар нь 2012 оны 8 сард байсан.

pkgng нь ports-mgmt/portmaster эсвэл ports-mgmt/portupgrade гэх мэт порт удирдах хэрэгслүүдийг солихоор хийгдээгүй. ports-mgmt/portmaster ба ports-mgmt/portupgrade нь гуравдагч талын програм хангамжийг бинар багц болон портын цуглуулгаас суулгаж чаддаг бол pkgng нь зөвхөн бинар багцуудыг суулгадаг.

### 5.5.1. pkgng ашиглаж эхлэх

FreeBSD 9.1 ба түүнээс хойшхи хувилбарууд pkgng-д зориулсан "bootstrap" хэрэгсэлтэй байдаг. bootstrap хэрэгсэл нь pkgng-г татаж аваад суулгана.

Системийг эхлүүлэхийн тулд:

```
# /usr/sbin/pkg
```

FreeBSD-н өмнөх хувилбаруудын хувьд pkgng-г портын цуглуулга эсвэл бинар багц хэлбэрээр суулгах ёстой.

pkgng портыг суулгахын тулд дараахийг ажиллуулна:

```
# cd /usr/ports/ports-mgmt/pkg
# make
# make install clean
```

Бинар багцыг суулгахын тулд дараахийг ажиллуулна:

```
# pkg_add -r pkg
```

pkgng багц удирдах хэрэгсэл нь FreeBSD 7.*X* эсвэл FreeBSD 8.0 хувилбарууд дээр дэмжигдээгүй.

FreeBSD-г өмнөх суулгацын хувьд pkg_install багцын өгөгдлийн санг шинэ формат руу хувиргах шаардлагатай. Багцын өгөгдлийн санг хувиргахын тулд дараахийг ажиллуулна:

```
# pkg2ng
```

Энэ алхам нь гуравдагч талын програм хангамж суугаагүй шинэ суулгацуудын хувьд шаардлагагүй.

Энэ алхмыг буцааж болохгүй. Багцын өгөгдлийн санг pkgng формат руу хөрвүүлсний дараа pkg_install хэрэгслүүдийг ашиглах ёсгүй.

Багцын өгөгдлийн санг шинэ хувилбар руу хувиргалт хийх явцад алдаа гарч болох юм. Ерөнхийдөө эдгээр алдааг хаях нь аюулгүй боловч амжилттай хөрвөгдөөгүй гуравдагч талын програм хангамжийн жагсаалт `pkg2ng` ажиллаж дууссаны дараа харагддаг. Эдгээрийг гараар засах ёстой.

FreeBSD портын цуглуулга нь pkg_install биш pkgng ашиглан шинэ програмыг бүртгэж байгаа эсэхийг шалгахын тулд 10.*X* хувилбараас өмнөх FreeBSD хувилбарууд доорх мөрийг /etc/make.conf файлд байхыг шаарддаг:

```
WITH_PKGNG= yes
```

### 5.5.2. pkgng орчныг тохируулах

pkgng багц удирдах систем нь ихэнх үйлдэлдээ багцын репозиторыг ашигладаг. Багцын анхдагч репозиторын байрлал /usr/local/etc/pkg.conf юм уу эсвэл тохиргооны файл дахь утгыг дарж бичдэг `PACKAGESITE` орчны хувьсагчид тодорхойлогддог.

pkgng-н нэмэлт тохиргооны боломжуудыг pkg.conf(5)-д тайлбарласан байгаа.

### 5.5.3. pkgng-н үндсэн үйлдлүүд

pkgng-г хэрэглэх мэдээллийг pkg(8) гарын авлагын хуудаснаас эсвэл `pkg`-г нэмэлт өгөгдөлгүйгээр ажиллуулж харж болно.

pkgng тушаалын аргумент бүрийг тухайн тушаалын гарын авлагын хуудсанд тайлбарласан байдаг. Жишээ нь `pkg install`-н гарын авлагыг уншихын тулд дараахийн аль нэгийг ажиллуулж болно:

```
# pkg help install
```

```
# man pkg-install
```

#### 5.5.3.1. pkgng ашиглан суулгасан багцуудын талаарх мэдээллийг олж авах

Систем дээр суусан багцуудын мэдээллийг `pkg info` тушаал ашиглан харж болно. pkg_info(1) -н нэгэн адил багцын хувилбар болон бүх багцын тайлбарыг харуулах болно.

Тухайн багцын мэдээллийг ингэж харж болно:

```
# pkg info packagename
```

Жишээ нь систем дээр pkgng-н ямар хувилбар суулгасан байгааг харахын тулд дараахийг ажиллуулна:

```
# pkg info pkg
pkg-1.0.2          New generation package manager
```

#### 5.5.3.2. pkgng ашиглан багц суулгах ба устгах

Ерөнхийдөө FreeBSD-н ихэнх хэрэглэгчид бинар багцыг дараахийг ажиллуулж суулгана:

```
# pkg install packagename
```

`pkg install` нь pkgng орчныг тохируулах-д дурдсанаар репозиторын өгөгдлийг ашигладаг. Эсрэгээр pkg-add(8) нь репозиторын өгөгдөл эсвэл `PACKAGESITE`-д заасныг ашигладаггүй учир хамаарлуудыг зөв хянадаггүй бөгөөд дутуу байгаа хамаарлуудыг алсын эхээс татаж авдаггүй. Энэ хэсэг нь `pkg install`-г хэрхэн ашиглах талаар харуулна. `pkg add`-г хэрэглэх талаарх мэдээллийг pkg-add(8)-с үзнэ үү.

`pkg install` ашиглан нэмэлт бинар багцуудыг суулгаж болно. Жишээ нь curl-г суулгахын тулд:

```
# pkg install curl
Updating repository catalogue
Repository catalogue is up-to-date, no need to fetch fresh copy
The following packages will be installed:

    Installing ca_root_nss: 3.13.5
    Installing curl: 7.24.0

The installation will require 4 MB more space

1 MB to be downloaded

Proceed with installing packages [y/N]: y
ca_root_nss-3.13.5.txz      100%    255KB   255.1KB/s 255.1KB/s 00:00
curl-7.24.0.txz         100%    1108KB  1.1MB/s 1.1MB/s     00:00
Checking integrity... done
Installing ca_root_nss-3.13.5... done
Installing curl-7.24.0... done
```

Шинэ багц болон хамаарлууд хэлбэрээр суусан нэмэлт багцуудыг суулгасан багцуудын жагсаалтаас харж болно:

```
# pkg info
ca_root_nss-3.13.5  The root certificate bundle from the Mozilla Project
curl-7.24.0 Non-interactive tool to get files from FTP, GOPHER, HTTP(S) servers
pkg-1.0.2   New generation package manager
```

Хэрэгцээгүй болсон багцуудыг `pkg delete` тушаалаар устгаж болно. Жишээ нь curl хэрэггүй бол:

```
# pkg delete curl
The following packages will be deleted:

    curl-7.24.0_1

The deletion will free 3 MB

Proceed with deleting packages [y/N]: y
```

```
Deleting curl-7.24.0_1... done
```

#### 5.5.3.3. pkgng ашиглан суулгасан багцуудыг шинэчлэх

Хуучирсан багцуудыг `pkg version` тушаалаар олж болно. Хэрэв локал портын мод байхгүй бол pkg-version(8) нь алсад байрлах репозиторын каталогийг ашиглах бөгөөд хэрэв байгаа бол багцын хувилбарыг танихын тулд локал портын модыг ашиглагддаг.

Багцыг шинэ хувилбар руу pkgng ашиглан шинэчилж болно. curl-н шинэ хувилбар гарсан гэж бодъё. Локал багцыг шинэ хувилбар руу шинэчилж болно:

```
# pkg upgrade
Updating repository catalogue
repo.txz        100%    297KB 296.5KB/s 296.5KB/s   00:00
The following packages will be upgraded:

    Upgrading curl: 7.24.0 -> 7.24.0_1

1 MB to be downloaded

Proceed with upgrading packages [y/N]: y
curl-7.24.0_1.txz   100% 1108KB 1.1MB/s 1.1MB/s     00:00
Checking integrity... done
Upgrading curl from 7.24.0 to 7.24.0_1... done
```

#### 5.5.3.4. pkgng ашиглан суулгасан багцуудыг аудит хийх

Заримдаа портын цуглуулга дахь програм хангамжид цоорхой илэрч болно. pkgng нь өөртөө ports-mgmt/portaudit багцтай төстэй аудит хийх боломжийг агуулдаг. Систем дээр суусан програм хангамжийг аудит хийхийн тулд дараахийг ажиллуулна:

```
# pkg audit -F
```

### 5.5.4. pkgng-н дэвшилтэй үйлдлүүд

#### 5.5.4.1. pkgng ашиглан автоматаар салбар хамаарлуудыг арилгах

Багцыг устгаснаар дээрх жишээн дээрх security/ca_root_nss шиг хэрэггүй хамаарлуудыг үлдээж болох юм. Тийм багцууд нь суусан хэвээр байх боловч юу ч тэднээс хамааралгүй байдаг. Хамаарал болон суусан хэрэггүй багцуудыг автоматаар илрүүлж устгаж болно:

```
# pkg autoremove
Packages to be autoremoved:
    ca_root_nss-3.13.5

The autoremoval will free 723 kB
```

```
Proceed with autoremoval of packages [y/N]: y
Deinstalling ca_root_nss-3.13.5... done
```

#### 5.5.4.2. pkgng багцын өгөгдлийн санг нөөцлөх нь

pkg_install багц удирдах системээс ялгаатай нь pkgng өөрийн гэсэн өгөгдлийн санг нөөцлөх аргатай байдаг. Багцын өгөгдлийн сангийн агуулгыг гараар нөөцлөхийн тулд дараахийг ажиллуулна:

```
# pkg backup -d pkgng.db
```

*pkgng.db* файлын нэрийг тохирсон файлын нэрээр солих хэрэгтэй.

Мөн нэмэлтээр pkgng нь багцын өгөгдлийн санг өдөр тутам автоматаар нөөцөлж байх periodic(8) скрипттэй байдаг. Гэхдээ энэ нь periodic.conf(5) файлд `daily_backup_pkgng_enable` хувьсагчийг `YES` гэж тохируулж өгсөн тохиолдолд ажилладаг.

pkg_install-н давтамжтайгаар ажиллах скрипт багцын өгөгдлийн санг нөөцлөхөөс сэргийлэхийн тулд periodic.conf(5) файлд `daily_backup_pkgdb_enable` хувьсагчийг `NO` болгож тохируулна.

Өмнөх багцын өгөгдлийн сангийн нөөцийн агуулгыг сэргээхийн тулд дараахийг ажиллуулна:

```
# pkg backup -r /path/to/pkgng.db
```

#### 5.5.4.3. pkgng багцуудыг устгах нь

Анхдагчаар pkgng нь бинар багцуудыг pkg.conf(5)-н `PKG_CACHEDIR`-д заасан кэш санд хадгалдаг. `pkg upgrade` ашиглан багцуудыг шинэчлэх үед шинэчилсэн багцуудын хуучин хувилбарууд автоматаар устдаггүй.

Хуучирсан бинар багцуудыг устгахын тулд:

```
# pkg clean
```

#### 5.5.4.4. pkgng багцын мета өгөгдлийг өөрчлөх

FreeBSD-н портын цуглуулга дахь програм хангамжууд уламжлалаар бол хувилбарын гол өөрчлөлтөд өртөх магадлалтай байдаг. pkg_install-с ялгаатай нь pkgng багцын эхийг шинэчлэх тушаалтай байдаг. Жишээ нь хэрэв lang/php5 анхдагчаар `5.3` хувилбар дор байж байгаад `5.4`-г оруулах зорилгоор lang/php53 руу өөрчлөгдсөн бол pkg_install нь багцын өгөгдлийн санг шинэчлэхийн тулд аль портоос эхэлж суусныг мэдэх ports-mgmt/portmaster гэх мэт нэмэлт програм хэрэглэхийг шаарддаг.

ports-mgmt/portmaster ба ports-mgmt/portupgrade портуудаас ялгаатай нь шинэ болон хуучин хувилбарууд ямар дарааллаар жагсаанаас хамаарч өөр байдаг. pkgng-н хувьд синтакс нь:

```
# pkg set -o category/oldport:category/newport
```

Жишээ нь дээрх жишээний багцын эхийг өөрчлөхийн тулд дараахийг ажиллуулна:

```
# pkg set -o lang/php5:lang/php53
```

Бас нэг өөр жишээ нь lang/ruby18-г lang/ruby19 руу шинэчлэхийн тулд дараахийг ажиллуулна:

```
# pkg set -o lang/ruby18:lang/ruby19
```

Төгсгөлийн жишээ нь libglut хуваалцсан сангийн эхийг graphics/libglut-с graphics/freeglut руу өөрчлөхийн тулд дараахийг ажиллуулна:

```
# pkg set -o graphics/libglut:graphics/freeglut
```

Багцын эхийг солих үед ихэнх тохиолдолд өөрчлөгдсөн эхийн багцаас хамаарсан багцуудыг дахин суулгах нь чухал байдаг. Хамаарсан багцуудыг дахин суулгахын тулд дараахийг ажиллуулна:

```
# pkg install -Rf graphics/freeglut
```

## 5.6. Портын цуглуулгыг хэрэглэх нь

Дараах хэсэгт системдээ портын цуглуулгыг ашиглан хэрхэн програм суулгах болон устгах талаар өгүүлэх болно. `make` хөрвүүлэгчийн байршил болон орчны хувьсагчийн тухай нэмэлт тодорхой мэдээллийг ports(7) хуудаснаас харна уу.

2012 оны дундаас эхлээд FreeBSD портын төсөл хувилбар удирдах системийн CVS-с Subversion рүү шилжсэн. Ерөнхийдөө портыг ашиглах арга бол Portsnap-г ашиглах явдал юм. Портын локал өөрчлөлт шаардлагатай (нэмэлт локал нөхөөс арчилдаг) хэрэглэгчид магадгүй Subversion-г ашиглахыг илүүд үзэж болох юм. CVSup үйлчилгээ 2013 оны 2 сарын 28-с эхлээд ашиглагдахаа болих тул цаашид ашиглахыг зөвлөхгүй.

### 5.6.1. Портын цуглуулгыг суулгах нь

Портын цуглуулга нь /usr/ports сан дотор Makefiles, patches буюу нөхөгч файл, мөн тайлбар файлуудын цуглуулга юм. Эдгээр файлууд нь FreeBSD дээр програмуудыг бүтээж суулгахад

хэрэглэгддэг. Доор байгаа зааврууд нь FreeBSD суулгах явцад портын цуглуулга суугаагүй бол түүнийг авах хэд хэдэн аргуудыг харуулж байна.

**Procedure: Portsnap арга**

Portsnap нь портын цуглуулгыг татаж аван шинэчлэхэд зориулагдсан бөгөөд ихэнх хэрэглэгчдийн хувьд сонголт болсон хурдан, хэрэглэхэд хялбар хэрэгсэл юм. Portsnap-н нэмэлт тайлбарын талаар Portsnap-г хэрэглэх нь хэсгээс үзнэ үү.

1. Шахсан хэлбэртэй портын цуглуулгыг /var/db/portsnap сан дотор татаж авах хэрэгтэй. Хэрэв та хүсвэл энэ алхмын дараа интернэтээс салгаатай ажиллаж болдог.

```
# portsnap fetch
```

2. Хэрэв та Portsnap-г анх удаагаа ажиллуулж байгаа бол шахагдсан уг цуглуулгыг /usr/ports сан дотор задална:

```
# portsnap extract
```

   Portsnap-г эхний удаа дээрх маягаар ашиглаж эхэлсний дараа /usr/ports санг доорх тушаалаар шинэчилнэ:

```
# portsnap update
```

**Procedure: Subversion арга**

Хэрэв портын цуглуулгыг илүү хянах шаардлагатай бол (жишээ нь локал өөрчлөлтийг арчлах бол) Subversion-г портын цуглуулгыг татахдаа ашиглаж болно. Subversion-ий талаар дэлгэрэнгүйг the Subversion Primer хуудаснаас үзнэ үү.

1. Subversion-г портын модыг татахаасаа өмнө суулгасан байх шаардлагатай. Хэрэв портын мод аль хэдийн байгаа бол Subversion-г иймэрхүү маягаар суулгаарай:

```
# cd /usr/ports/devel/subversion
# make install clean
```

   Хэрэв портын мод байхгүй бол Subversion-г багц хэлбэрээр суулгаж болно:

```
# pkg_add -r subversion
```

   Хэрэв pkgng нь багцыг удирдахад хэрэглэгдэж байгаа бол Subversion-г ингэж суулгаж болно:

```
# pkg install subversion
```

2. Портын модыг татаж авна. Илүү ажиллагааг хурдан болгохын тулд доорх тушаалын *svn.FreeBSD.org* гэдгийн оронд танд газар зүйн байрлалын хувьд ойр байрлалыг Subversion толин тусгал хэсгээс харан сонгоорой. Итгэмжлэн нийлүүлэгчид зөв протокол сонгож ажиллахын тулд эхлээд Subversion Primer тусламжийг унших ёстой.

```
# svn checkout svn://svn.FreeBSD.org/ports/head /usr/ports
```

3. Subversion-ий эхний удаагийн таталтын дараа /usr/ports-г шинэчлэхийн тулд дараах тушаалыг ажиллуулна:

```
# svn update /usr/ports
```

**Procedure: Sysinstall арга**

Энэ арга нь суулгацын төхөөрөмжөөс sysinstall-г ашиглан портын цуглуулгыг суулгах арга юм. Гэхдээ тухайн төхөөрөмж дээр байгаа цуглуулга нь тухайн хувилбарыг гаргасан өдрийн хувилбартай байдгийг анзаарах хэрэгтэй. Хэрэв танд интернэт холболт байдаг бол дээр өгүүлсэн хоёр аргыг урьтал болгох нь зүйтэй.

1. `root` эрхээр ороод `sysinstall` гэсэн тушаал өгөх хэрэгтэй:

```
# sysinstall
```

2. Configure гэдгийг сонгоод `Enter` товч дарна.
3. Distributions цэсийг сонгоод `Enter` дээр дарна.
4. ports гэж сонгон `Space` товч дээр дарах хэрэгтэй.
5. Дээр буй Exit гэдгийг олж сонгон `Enter` дээр дарна.
6. CDROM юм уу FTP гэх мэт суулгах төхөөрөмжөө сонгоно.
7. Дээр буй Exit цэсийг сонгоод `Enter` дээр дарна.
8. `X` дээр дарж sysinstall-с гарах болно.

## 5.6.2. CVSup/csup-с portsnap рүү шилжих нь

2013 оны 2 сарын 28-аас эхлэн портын мод CVS рүү экспорт хийгдэхгүй учраас CVSup болон csup нь портын модны шинэчлэлийг хийхгүй.

**Procedure: Portsnap руу шилжүүлэх нь**

Шилжүүлэлт нь 1 GB орчим дискний хэмжээ /usr дээр байхыг шаардах бөгөөд дээрээс нь Portsnap нь /var санд 150 MB дискний хэмжээ шаардана.

1. cron(8) дотор чинь автоматаар CVSup эсвэл csup-г дуудсан ажлууд байвал хааж болиулах хэрэгтэй.
2. Байгаа портын модоо өөр түр зуурын байр руу зөөнө:

```
# mv /usr/ports /usr/ports.old
```

3. Шинэ портын модыг Portsnap-р татаж /usr/ports-д задална:

```
# portsnap fetch extract
```

4. distfile-ууд болон хадгалсан багцуудыг шинэ портын мод руугаа зөөнө:

```
# mv /usr/ports.old/distfiles /usr/ports
# mv /usr/ports.old/packages /usr/ports
```

5. Хуучин портын модыг устгана:

```
# rm -rf /usr/ports.old
```

6. Хэрэв CVSup өмнө нь ашиглагдаж байсан бол одоо устгаж болно:

```
# pkg_delete -r -v cvsup-without-gui-\*
```

   pkgng хэрэглэгчид дараах тушаалыг ашиглаж болно:

```
# pkg delete cvsup-without-gui
```

Portsnap ашиглан портын модыг шинэчлэх болон Portsnap-н талаар дэлгэрэнгүйг Portsnap хэрэглэх нь хэсгээс үзнэ үү.

## 5.6.3. Порт суулгах нь

Портын цуглуулгын талаар яриа хийхийн өмнө портын "skeleton буюу араг яс"ны талаар ойлголт авах хэрэгтэй. Хамгийн энгийн ойлголтоор бол энэ нь FreeBSD системд програмыг цэвэрхэн хөрвүүлээд суулгахыг зааварласан товч заавар байдаг. Порт болгоны араг яс дараах зүйлсийг агуулж байдаг:

- Makefile. Makefile нь янз бүрийн заавар бичлэгүүд агуулж байдаг бөгөөд энэ нь програм хэрхэн хөрвүүлэгдэх, бас хаана суугдах зэрэг үйлдлүүдийг тодорхойлсон байгаа.
- distinfo файл. Энэ файл нь хөрвүүлж бүтээгдэх ёстой файлуудын татаж авалт болон тэдгээрийн алдаагүй татагдсан эсэхийг (sha256(1) ашиглан) шалгах зэрэг мэдээллүүдийг агуулж байдаг.
- files нэртэй сан. Энэ санд таны FreeBSD систем дээр суугдах програмуудын хөрвүүлэгдэхэд хэрэглэгддэг patches буюу нөхөөс файлуудыг агуулна. Нөхөөс файлууд нь жижигхэн хэмжээний файл бөгөөд тодорхой зарим файлд гарсан өөрчлөлтүүдийг агуулдаг. Нөхөөсүүд текст хэлбэрээр оршдог ба голдуу "10 дугаар мөрийг устга" эсвэл "26 дугаар мөрийг үүгээр соль ..." гэсэн зааврууд байдаг. Нөхөөсүүдийг мөн "diffs буюу ялгааны" төрлийн файл гэж ярьцгаадаг. Ийм ялгааг нь илэрхийлсэн файлыг үүсгэхдээ diff(1) програмыг хэрэглэдэг юм.

  Энэ санд портыг бүтээхэд шаардагдах өөр төрлийн файлууд бас байж болох юм.
- pkg-descr файл. Энэ файл дотор тухайн програмын тухай нэлээн дэлгэрэнгүй тодорхойлолт агуулагдаж байдаг.
- pkg-plist файл. Энэ файл дотор порт суугдах явцад хуулагдаж суугдах файлуудын жагсаалт байдаг. Энэ нь мөн портыг устгах үед портын системд мэдэгдэх файлуудын жагсаалт билээ.

Зарим портууд pkg-message гэх мэтийн өөр файлууд агуулж байдаг. Портын систем нь онцгой тохиолдолд уг файлуудтай хандаж тухайн портод харгалзах үйлдлүүдийг хийх болно. Хэрэв та ийм файлуудын тухай дэлгэрэнгүй мэдээлэл мөн портын тухай үндсэн ойлголт авахыг хүсвэл FreeBSD порт бүтээгчийн гарын авлага хуудаснаас харна уу.

Порт дотор програмын эх бичлэгийг хэрхэн хөрвүүлж бүтээх тухай заавар байдаг болохоос уг програмын эх бичлэг нь байдаггүй. Та уг програмын эх бичлэгийг CD-ROM эсвэл интернэтээс уг зохиогчийнх нь гаргасан хэлбэрээр татаж авч болно. Голдуу эх бичлэгүүд нь tar болон gzip шахалтаар шахаж бэлдсэн байдаг боловч заримдаа өөр төрлийн хэрэгсэл ашиглан шахсан тохиолдол тулгарч магадгүй. Ямар ч хэлбэрээр програмын эх бичлэгийг авсан байг, түүнийг "distfile" гэж нэрийддэг. Доор FreeBSD порт суулгах хоёр аргыг танилцуулж байна.

Порт суулгахын тулд та `root` эрхэнд сэлгэсэн байх ёстой.

Ямар нэгэн портыг суулгахаасаа өмнө портынхоо цуглуулгыг та шинэчилсэн байх хэрэгтэй бөгөөд http://vuxml.freebsd.org/ хуудсанд тухайн порттой холбоотой аюул нууцлалын тухай сэдэв хөндөгдсөн эсэхийг шалгах хэрэгтэй.

Ямар ч порт суулгахаасаа өмнө аюулгүйн нууцлалын хувьд сул тал буй эсэхийг portaudit-р автоматаар шалгаж болно. Энэ хэрэгслийг портын цуглуулга дотор олох боломжтой (ports-mgmt/portaudit). Шинэ портыг суулгахаасаа өмнө `portaudit -F` гэж ажиллуулснаар аюулгүйн нууцлалд гарсан сул тал нүхнүүдийн тухай мэдээллийн өгөгдлийн баазаас мэдээллүүдийг авч нөхөлт хийдэг. Аюулгүйн нууцлалын мэдээллийн өгөгдлийн баазын шинэчлэх явц өдөр болгон давтагдаж хийгдэх болно.

> Нэмэлт дэлгэрэнгүй мэдээллийг portaudit(1) болон periodic(8) хуудаснаас харна уу.

Портын цуглуулга таныг интернэт холболттой гэж авч үздэг. Хэрэв танд интернэт холболт байхгүй бол distfile файлуудыг /usr/ports/distfiles санд хуулах хэрэгтэй.

Эхлэхийн өмнө суулгах гэж буй портын санд орох хэрэгтэй:

```
# cd /usr/ports/sysutils/lsof
```

Тэгээд lsof санд орсон хойноо уг сан дотор та тухайн портын араг ясыг харах болно. Дараагийн алхам бол портыг хөрвүүлэх буюу "бүтээх" билээ. Ингэхийн тулд тушаал бичих мөрөнд `make` гэж бичнэ. Ингэж гүйцэтгэсний дараа дараах маягийн явцын мэдээллийг та харах болно:

```
# make
>> lsof_4.57D.freebsd.tar.gz doesn't seem to exist in /usr/ports/distfiles/.
>> Attempting to fetch from ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/.
===>  Extracting for lsof-4.57
...
[extraction output snipped]
...
>> Checksum OK for lsof_4.57D.freebsd.tar.gz.
===>  Patching for lsof-4.57
===>  Applying FreeBSD patches for lsof-4.57
===>  Configuring for lsof-4.57
...
[configure output snipped]
...
===>  Building for lsof-4.57
...
[compilation output snipped]
...
#
```

Хөрвүүлэлт дуусаад та буцаад тушаал бичих мөрөнд ирэнгүүт хийх ёстой дараагийн алхам бол портыг суулгах билээ. Ингэхийн тулд таны хийх ёстой зүйл бол `make` тушаалыг өөр нэг үгтэй хамт бичих ёстой бөгөөд тэр үг нь `install` юм:

```
# make install
===>  Installing for lsof-4.57
...
[installation output snipped]
...
===>   Generating temporary packing list
===>   Compressing manual pages for lsof-4.57
===>   Registering installation for lsof-4.57
```

```
===>  SECURITY NOTE:
      This port has installed the following binaries which execute with
      increased privileges.
#
```

Ингээд тушаал бичих мөр боломжтой болонгуут суулгасан портоо ажиллуулах боломжтой болдог. Бид нарын жишээ авч суулгасан `lsof` програм нь нууцлалын давуу эрх шаарддаг болохоор аюулгүйн нууцлалын анхааруулга харуулагддаг. Порт суулгах үед хэрэв анхааруулга харуулагдваас түүнийг тун анхааралтай уншиж ойлгох хэрэгтэй.

Програмыг хөрвүүлж бүтээхэд хэрэглэгдсэн түр файлуудыг хадгалсан дэд сангуудыг устгах нь зүйтэй. Энэ нь дискний зайг хэмнэхээс гадна тухайн портыг шинэчлэх үед алдаа гаргуулахгүй маш сайн зуршил юм.

```
# make clean
===>  Cleaning for lsof-4.57
#
```

Та `make`, `make install` мөн `make clean` гэсэн дамжлагуудыг `make install clean` гэж товчилж бичиж болно.

Зөвхөн `make install` тушаал ашиглан порт суулгахад заримдаа хэрэглэгчээс сонголт оруулахыг хүлээж удах тохиолдлууд байж болдог. Учир нь зарим тохиолдолд анхдагчаар хэрэглэгчээс сонголт хийхийг асуухаар хийгдсэн байдаг. Иймээс олон хамааралтай үед заримдаа нэг портыг бүтээж эмхэтгэхэд төвөгтэй болгодог. Үүнээс гарахын тулд `make config-recursive` тушаал ашиглан тохиргоог нэг удаагаар хийх хэрэгтэй. Дараа нь `make install [clean]` тушаалыг ажиллуулах хэрэгтэй.

`config-recursive` гэдгийг ашиглахад тохиргоо хийгдэх портуудын жагсаалт `all-depends-list` гэсэн make(1)-н нэмэлт боломжоор нэгтгэгддэг. Хамааралтай бүх портуудын тохиргоонууд тодорхой болтол буюу dialog(1) дэлгэц гарч ирэхгүй болж бүх портуудын тохиргоо зөвөөр хийгдтэл `make config-recursive` тушаалыг ажиллуулахыг ихэвчлэн зөвлөдөг.

Зарим бүрхүүлийн орчнууд нь `PATH` орчны хувьсагч дотор буй сангууд доторх ачаалагдаж болдог програмуудын тушаалын хайлтыг түргэвчлэх зорилгоор түр хадгалагчид хадгалсан байдаг. Та хэрэв ийм төрлийн бүрхүүлийн орчин ашиглаж байгаа бол порт суулгасны дараа `rehash` тушаалыг хэрэглэн портын ачаалагдах файлыг бүрхүүлийн таних сан дотор нэмдэг. Энэ тушаал нь `tcsh` маягийн бүрхүүлд ажилладаг. `sh` төрлийн бүрхүүлд `hash -r` гэж ашигладаг. Тухайн бүрхүүлд хамаатай нэмэлт мэдээллийг нь харна уу.

FreeBSD Mall зэрэг зарим гуравдагчдын гаргасан DVD-ROM дээр distfiles файлууд агуулагдсан байдаг. Тэдгээрийг портын цуглуулганд хэрэглэж болно. DVD-ROM-г /cdrom санд таниулж холбоно. Хэрэв та өөр санд холбосон бол уг сангийн нэрийг `CD_MOUNTPTS` орчны хувьсагчид

зааж өгөөрэй. Шаардлагатай distfiles файлууд нь дискнээс автоматаар хэрэглэгддэг.

Зарим нэг цөөн тооны портуудын лицензийг нь анхаарах хэрэгтэй. Ийм төрлийн портуудын эх бичлэг нь CD-ROM дээр байдаггүй. Тэдгээрийг татаж авах юм уу цааш нь түгээхийн тулд ямар нэгэн гэрээ бөглөх илгээх ёстой болдог. Хэрэв таны суулгах порт CD-ROM дээр байхгүй байгаа бол интернэт рүү орон хэлсний дагуу гүйцэтгэх хэрэгтэй.

Портын систем нь fetch(1) хэрэгслийг ашиглан файлуудыг татаж авдаг. Уг хэрэгсэл нь `FTP_PASSIVE_MODE`, `FTP_PROXY`, мөн `FTP_PASSWORD` гэх мэтчилэн төрөл бүрийн орчны хувьсагчдыг ашиглаж ажилладаг. Хэрэв та галт хана юм уу FTP/HTTP прокси хэрэглэдэг бол эдгээр хувьсагчдад тохирох утгыг нь зааж өгөх хэрэгтэй. fetch(3) хуудаснаас хувьсагчдын бүрэн жагсаалтыг харна уу.

Интернэтэд байнга холбогддоггүй хэрэглэгчдэд зориулж `make fetch` тушаал байдаг. Уг тушаалыг портын дээд сан (/usr/ports) дотор гүйцэтгэхэд шаардагдах бүх файлуудыг татаж авдаг. Уг тушаалыг мөн /usr/ports/net гэх зэргийн дэд сан дотор гүйцэтгэж болно. Хэрэв тухайн порт өөр сан юм уу портоос хамаардаг бол fetch гүйцэтгэл нь тэдгээр хамаарлыг нь *нөхөж татдаггүй*г анхаараарай. Харин `fetch` гэдгийг `fetch-recursive` гэж орлуулснаар тухайн портын хамааралтай порт болон сангуудыг давхар татаж авдаг.

Та бүх портуудыг хөрвүүлж бүтээнэ гэвэл `make` тушаалыг `make fetch` тушаалыг тайлбарласан шиг дээд санд гүйцэтгэж болдог. Гэвч зарим порт байхгүй тохиолдолд ийм үйлдэл бол аюултай. Мөн зарим портууд хоёр өөр файлыг нэг нэрээр суулгах хүндрэл гаргаж болзошгүй.

Мөш цөөхөн тохиолдолд хэрэглэгчид `MASTER_SITES` (татаж авах файлуудын байршил) хаягнаас өөр байршил ашиглан эх файлуудыг татаж авах шаардлага гарч болох юм. Ийм үед `MASTER_SITES` хувьсагчийн утгыг дараах тушаалаар өөрчилдөг:

```
# cd /usr/ports/directory
# make MASTER_SITE_OVERRIDE= \
ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/distfiles/ fetch
```

Энэ жишээн дээр бид `MASTER_SITES`-н утгыг `ftp.FreeBSD.org/pub/FreeBSD/ports/distfiles/` гэж өөрчиллөө.

Зарим порт нь өөрийнхөө зарим хэсгийг шаардлагагүй гэж үзэх тохиолдолд, эсвэл аюулгүйн нууцлалын шалтгаанаар ч юм уу өөрчлөлт хийх боломж (эсвэл шаарддаг) өгдөг. www/firefox, security/gpgme, мөн mail/sylpheed-claws зэргүүдийг жишээ болгож болох юм. Иймэрхүү мэдэгдлүүд байвал тухайн үед танд харуулагдах болно.

#### 5.6.3.1. Портын анхдагч санг дарж бичих

Заримдаа портыг хөрвүүлэх болон суулгах өөр санг хэрэглэх нь хэрэгтэй (онц шаардлагатай) байдаг. `WRKDIRPREFIX` болон `PREFIX` хувьсагчдын утгыг өөрчилж анхдагч сангийн байршлыг

сольдог. Жишээ нь:

```
# make WRKDIRPREFIX=/usr/home/example/ports install
```

гэсэн тушаал нь портыг /usr/home/example/ports сан дотор хөрвүүлээд хөрвүүлэгдэж бүтээгдсэн файлуудыг /usr/local санд суулгадаг.

```
# make PREFIX=/usr/home/example/local install
```

гэсэн тушаал нь портыг /usr/ports санд хөрвүүлж бэлдээд /usr/home/example/local санд суулгана.

Мэдээж

```
# make WRKDIRPREFIX=../ports PREFIX=../local install
```

гэж хоёр хувьсагчийг хослуулан хэрэглэж бас болно (танд зөвхөн ерөнхий ойлголт өгөх зорилгоор урт мөрийг товчилж харуулсан билээ).

Мөн та уг хувьсагчдыг өөрийнхөө орчны хувьсагч болгож зарлаж болно. Тухайн бүрхүүлийнхээ орчны тухай зааврыг эхлээд уншиж танилцах нь зүйтэй.

#### 5.6.3.2. `imake`-г хэрэглэх

Зарим порт `imake`-г хэрэглэдэг ( X цонхот системийн хэсэг) бөгөөд `PREFIX` хувьсагч ашиглалгүй /usr/X11R6 санд суулгадаг. Үүнтэй ижил зарим Perl портууд `PREFIX` хувьсагч хэрэглэлгүй Perl-н санд суулгадаг. Эдгээр портуудыг `PREFIX` хувьсагчтай хамтран ажиллуулж энэ хувьсагчийг ойлгож хүндэтгэдэг болгоно гэдэг бол тун хэцүү бөгөөд бараг боломжгүй ажил юм.

#### 5.6.3.3. Портуудыг дахин тохируулах нь

Тухайн портуудыг бүтээж байхад чинь танд бүтээх тохируулгуудаас сонгож болох ncurses дээр тулгуурласан цэс харуулагддаг. Портыг нэгэнт бүтээсний дараа тэдгээр тохируулгуудыг нэмэх, хасах, эсвэл өөрчлөхийн тулд энэ цэсэнд дахин орохыг хүсэх нь хэрэглэгчдийн хувьд тийм ч нийтлэг биш юм. Үүнийг хийх олон аргууд байдаг. Нэг арга нь портыг агуулж байгаа сан уруу ороод урьдын адил тохируулгуудыг сонгосон хэвээр байгаа цэсийг дахин харуулах `make config` гэж тушаалыг ажиллуулах явдал юм. Өөр нэг арга нь портын хувьд бүх тохиргооны сонголтуудыг харуулах `make showconfig` тушаалыг ашиглах явдал юм. Бас нэг өөр арга нь бүх сонгогдсон тохируулгуудыг арилгаад танд дахин эхлэх боломж олгох `make rmconfig` тушаалыг ажиллуулах явдал юм. Эдгээр бүх аргууд болон бусдын талаар ports(7) гарын авлагын хуудсанд илүү дэлгэрэнгүй тайлбарласан байгаа.

### 5.6.4. Суугдсан портыг усттах нь

Та одоо портыг хэрхэн суулгах талаар мэдсэн юм чинь хэрэв буруу портоо суулгасан бол түүнийг хэрхэн утсгадаг талаар мэдэхийг хүсэж байгаа байх. Өмнөх жишээнд суулгасан

портоо бид устгацгаая (анхааралгүй уншсан нэгэнд нь `lsof` програм гэж санууллъя). Портууд багцтай ижилхэн pkg_delete(1) тушаалаар устгагдаж болно ( Багц конголт хэсэгт тайлбарласан буй):

```
# pkg_delete lsof-4.57
```

### 5.6.5. Порт шинэчлэх

Хамгийн түрүүнд pkg_version(1) тушаал ашиглан портын цуглуулгаас шинэ хувилбар агуулсан портуудыг жагсаах хэрэгтэй:

```
# pkg_version -v
```

#### 5.6.5.1. /usr/ports/UPDATING-г унших

Портыг шинэчлэхээсээ өмнө портын цуглуулгаа шинэчлээд /usr/ports/UPDATING файлыг шалгана. Энэ файл дотор портыг шинэчлэхэд шаардагдах үйлдлүүд болох өөрчлөгдсөн төрөл, тохируулгын өөрчлөлт, эсвэл өмнөх хувилбартайгаа зөрчилдөх хэсгүүд гэх мэтийн чухал мэдээллүүд агуулагдаж байдаг.

Хэрэв UPDATING файлд саяны таны уншсан зүйлүүдээс өөр юм агуулагдаж байвал уг файлд буй зааврыг дагах нь зүйтэй.

#### 5.6.5.2. Portupgrade ашиглан порт шинэчлэх нь

portupgrade хэрэгсэл нь портыг амархан шинэчлэхэд зориулагдсан. Үүнийг ports-mgmt/portupgrade портоос суулгаж болно. Бусад портын нэгэн адил порт суулгадаг журмаар `make install clean` гэсэн тушаалаар суулгадаг:

```
# cd /usr/ports/ports-mgmt/portupgrade
# make install clean
```

Суугдсан портын жагсаалтыг `pkgdb -F` тушаал ашиглан шалгаад үр дүнгийн мэдээлэлд байгаа зөрчлүүдийг нь залруулж засах хэрэгтэй. Шинэчлэл хийхээсээ өмнө ингэж зөрчлүүдийг арилгах нь тун сайн санаа билээ.

Хэрэв та `portupgrade -a` гэж ажиллуулбал таны системд суугдсан бүх хуучирсан портуудыг portupgrade шинэчилж эхэлдэг. Хэрэв та суугдах шинэчлэл болгонд лавлаж асуух үйлдэл хийлгэхийг хүсвэл `-i` конголт нэмээрэй.

```
# portupgrade -ai
```

Хэрэв та бүх шинэчлэгдэх порт биш харин зөвхөн тодорхой нэгэн портыг шинэчлэх хүсэлтэй бол `portupgrade багцны-нэр` гэж бичнэ. `-R` конголт нэмснээр тухайн програмд шаардагдах портуудыг portupgrade эхлээд нь шинэчлэдэг.

```
# portupgrade -R firefox
```

Портын оронд багц хэрэглэж суулгахыг хүсвэл `-P` сонголт хэрэглэнэ. Энэ сонголттой үед portupgrade нь `PKG_PATH` хувьсагчид зааж өгсөн жагсаалтад буй сан дотроос багцуудыг хайх ба хэрэв уг санд байхгүй байгаа бол сүлжээнээс татаж нөхдөг. Хэрэв дотоод сан болон сүлжээнээс багц олдоогүй тохиолдолд portupgrade нь портоор суулгахыг оролдох болно. Порт хэлбэрээр суулгах үйлдлийг зогсоохын тулд `-PP` сонголтыг нэмж өгдөг.

```
# portupgrade -PP gnome2
```

Хэрэв хөрвүүлж суулгалгүйгээр зөвхөн distfiles файлыг татаж авах (хэрэв `-P` сонголттой бол багц файлыг ) шаардлагатай бол `-F` сонголтыг ашиглаарай. Дэлгэрэнгүй мэдээллийг portupgrade(1) хэсэгт харна уу.

#### 5.6.5.3. Portmaster ашиглан портыг шинэчлэх нь

Portmaster нь "үндсэн" системд байдаг (бусад портуудаас хамаардаггүй) хэрэгслүүдийг ашиглахаар хийгдсэн бөгөөд аль портыг шинэчлэхээ тогтоохын тулд /var/db/pkg/ сан доторхийг ашигладаг. Тэрээр ports-mgmt/portmaster порт хэлбэрээр байдаг:

```
# cd /usr/ports/ports-mgmt/portmaster
# make install clean
```

Portmaster нь портуудыг дөрвөн төрөлд бүлэглэдэг:

- Root буюу язгуур портууд (хамаарсан портууд байхгүй, бусдаас хамаардаггүй)
- Trunk буюу гол портууд (хамаарсан портууд байхгүй, бусдууд хамаардаг)
- Салбар портууд (хамаарсан портуудтай, бусдууд хамаардаг)
- Навч портууд (хамаарсан портуудтай, бусдаас хамаардаггүй)

Та суулгагдсан бүх портуудыг жагсааж болох бөгөөд `-L` тохируулга ашиглан шинэчлэлт байгаа эсэхийг хайж болно:

```
# portmaster -L
===>>> Root ports (No dependencies, not depended on)
===>>> ispell-3.2.06_18
===>>> screen-4.0.3
        ===>>> New version available: screen-4.0.3_1
===>>> tcpflow-0.21_1
===>>> 7 root ports
...
===>>> Branch ports (Have dependencies, are depended on)
===>>> apache-2.2.3
        ===>>> New version available: apache-2.2.8
...
```

```
===>>> Leaf ports (Have dependencies, not depended on)
===>>> automake-1.9.6_2
===>>> bash-3.1.17
        ===>>> New version available: bash-3.2.33
...
===>>> 32 leaf ports

===>>> 137 total installed ports
        ===>>> 83 have new versions available
```

Бүх суулгагдсан портуудыг энэ хялбар тушаал ашиглан шинэчилж болно:

```
# portmaster -a
```

Анхдагчаар Portmaster нь байгаа портыг устгахаасаа өмнө нөөц багц үүсгэдэг. Хэрэв шинэ хувилбар суулгалт амжилттай болсон бол Portmaster нөөц багцыг устгана. `-b` сонголтыг ашиглаж Portmaster-т нөөцийг автоматаар устгахгүй байхыг тушаана. `-i` сонголтыг ашигласнаар порт бүрийг шинэчлэхээс өмнө асууж лавлан Portmaster-ийг интерактив горимд эхлүүлнэ.

Хэрэв шинэчлэх процессийн явцад алдаа гарвал та бүх портыг шинэчлэх буюу дахин бүтээхийн тулд `-f`-г ашиглаж болно:

```
# portmaster -af
```

Шинэ порт бүтээж суулгахаасаа өмнө бүх хамааралтай портуудыг шинэчилж Portmaster-ийг системд шинэ порт суулгахдаа та бас ашиглаж болно:

```
# portmaster shells/bash
```

Дэлгэрэнгүй мэдээллийг portmaster(8)-с үзнэ үү.

### 5.6.6. Порт ба дискний хэмжээ

Портын цуглуулга нь дискний хэмжээг байнга идэж байдаг. Портоос програмыг хөрвүүлж суулгасны дараа бэлдэж байсан work санг `make clean` тушаалаар цэвэрлэхээ үргэлж санах хэрэгтэй. Та портын цуглуулгыг бүхлээр нь дараах тушаалаар цэвэрлэж болно:

```
# portsclean -C
```

Маш олон эх файлууд distfiles санд явцын үр дүнд хадгалагдаж байдаг. Та тэдгээрийг гар аргаар устгаж болох ба дараах тушаал хэрэглэн ямар нэгэн порттой холбоогүй болсон бүх distfiles устгаж болно:

```
# portsclean -D
```

Эсвэл таны системд одоо суугдсан байгаа порттой хамаагүй бүх distfiles-уудыг дараах тушаалаар устгана:

```
# portsclean -DD
```

`portsclean` хэрэгсэл нь portupgrade програмын нэг бүрдэл хэсэг.

Суулгасан портоо хэрэглэхгүй болсон үедээ устгахаа мартаж болохгүй. Иймэрхүү автоматжуулалтыг гүйцэтгэдэг эвтэйхэн програм бол ports-mgmt/pkg_cutleaves порт билээ.

## 5.7. Порт суулгасны дараах үйлдлүүд

Ердийн програмыг суулгасны дараа тухайн програмтай холбоотой бичиг баримтыг унших, ямар нэгэн тохиргооны файл засварлах, эсвэл уг програм компьютер ачаалагдах үед эхлэх (хэрэв энэ нь далд чөтгөр бол) эсэхийг нь шалгах шаардлага гарч болзошгүй.

Суулгасан програм болгоныг тохируулах зарчим өөр өөр. Гэхдээ та шинэ програм суулгачихаад "Одоо яах вэ?" гэсэн асуулттай тулгарах үед дараах зүйлс тусалж магадгүй:

- pkg_info(1) тушаалыг ашиглан ямар файл хаана яаж суугдсаныг харна. Жишээ нь та саяхан FooPackage version 1.0.0, програмыг суулгасан бол

  ```
  # pkg_info -L foopackage-1.0.0 | less
  ```

  гэсэн тушаал нь уг програмыг суулгах явцад хуулсан бүх файлуудыг харуулдаг. man/ санд хуулсан файл байвал тун анхааралтай харах хэрэгтэй. Энэ нь гарын авлага файлууд байдаг. Мөн etc/ санд хуулсан файлууд нь тохируулгын файлууд байдаг бөгөөд doc/ санд бол дэлгэрэнгүй бичиг баримтуудыг хадгалсан байдаг.

  Хэрэв та програмынхаа хувилбарын талаар сайн мэдэхгүй байгаа бол

  ```
  # pkg_info | grep -i foopackage
  ```

  гэсэн тушаал нь бүх суугдсан програм дотор *foopackage* гэсэн нэртэй програм байвал харуулдаг. *foopackage*-н оронд хүссэн програмынхаа нэрийг бичээрэй.

- Програмын гарын авлага хаана суугдсаныг мэдсэн үедээ man(1)-г ашиглан харах хэрэгтэй. Мөн түүнчлэн ийм аргаар тохируулга болон нэмэлт мэдээллийн файлуудыг нь тухай бүрд нь харах хэрэгтэй.
- Хэрэв уг програм өөрийн гэсэн вэб хуудастай бол түүнд нь зорчин нэмэлт мэдээлэл авах, байнгын асуулт хариултыг нь үзэх гэх мэтчилэн оролдоорой. Уг програмын вэб хуудасны хаяг нь

```
# pkg_info foopackage-1.0.0
```

тушаалын гаралт дээр `WWW:` гэсэн хэсэгт байдаг.

- Эхлэн ачаалагдах үед эхлэх ёстой портууд (Интернэт сервер гэх мэт) голдуу /usr/local/etc/rc.d сан дотор эхлүүлэх файлаа хуулдаг. Та уг файлыг шаардлагатай бол засварлах юм уу өөрчилж болно. Үйлчилгээг эхлүүлэх хэсгээс дэлгэрэнгүй мэдээллийг харна уу.

## 5.8. Эвдрэлтэй портыг засах нь

Хэрэв та портыг ажиллуулж чадаагүй тохиолдолд хийж болох хэд хэдэн алхам бий:

1. Problem Report database хуудсанд уг портын засварлах заавар бий эсэхийг шалгах. Хэрэв байвал уг зааврын дагуу засварлах хэрэгтэй.
2. Уг портыг арчилж сайжруулдаг хүнээс тусламж авах. `make maintainer` тушаал ашиглаад юм уу эсвэл Makefile файл дотроос тухайн арчлагчийн захианы хаягийг нь олох хэрэгтэй. Захиандаа портынхоо нэр болон хувилбарыг оруулан (Makefile файлд буй `$FreeBSD:` мөрийг илгээх хэрэгтэй ) хэрэв боломжтой бол алдаа заасан явцын мэдээллүүдийг явуулах хэрэгтэй.

Зарим порт нь хувь хүнээр биш харин захианы жагсаалтаар арчлагдаж байдаг. Энэ жагсаалтад бүгд биш ч гэсэн ихэнх хүмүүс нь freebsd-listname@FreeBSD.org маягийн захианы хаягтай байдаг. Ийм хүмүүс рүү захиагаа явуулах хэрэгтэй.

Голдуу ports@FreeBSD.org-р арчлагдаж байгаа гэсэн портууд нэг тодорхой хүнээр арчлагдаагүй байдаг. Завсарлалт болон тусламж зэргүүд нь ихэнхдээ захианы жагсаалтад буй хүмүүсээс ирдэг. Туслан дэмжих хүмүүс бидэнд үргэлж хэрэгтэй байгаа!

Хэрэв та ямар нэгэн хариулт аваагүй бол send-pr(1)-г ашиглан алдааны мэдэгдэл ( FreeBSD-н алдаар мэдээлэх хэсгийг харна уу) хийж болно.

3. Өөрөө засаад үз! Порт хийгчдийн гарын авлага дотор "портын" ажиллах зарчим болон бүтэц заавар, засах дараалал, тэр байтугай өөрөө порт зохиох талаар дурдсан байгаа!
4. Ойрхон байгаа FTP хуудаснаас багцыг татаж авах хэрэгтэй. "Үндсэн" багцны цуглуулга `ftp.FreeBSD.org` хуудасны багцны санд байдаг бөгөөд толин тусгал сайтуудуудыг *эхлээд* турш! Ингэсэн нь эх бичлэгийг хөрвүүлэхээс түргэн бэлэн болсон багцыг татаж аван цаг хэмнэж байдаг. pkg_add(1) програмыг ашиглан өөртөө хадгалсан багцаа систем дээрээ суулгана.

# Chapter 6. X Цонхот систем

## 6.1. Ерөнхий агуулга

FreeBSD хэрэглэгчдэд зориулж график горимоор хангахын тулд X11-г ашигладаг. X11 бол X Цонхот системийн чөлөөтэй түгээгддэг хувилбар бөгөөд Xorg-д (болон энд дурдагдаагүй бусад програм хангамжийн багцуудад) хийгдсэн байдаг. X11-ийн Анхдагч бөгөөд албан ёсны хувилбар нь X.Org сангаас гаргасан Xorg бөгөөд түүний лиценз нь FreeBSD-ийн лицензтэй ижил төсөөтэй билээ.

X11-н дэмждэг дэлгэцтэй холбоотой төхөөрөмжийн талаар Xorg вэб хуудаснаас хараарай.

Энэ бүлгийг уншиж дууссаны дараа та дараах зүйлсийг мэдэх болно:

- X Цонхот системийн төрөл бүрийн бүрдлүүд болон тэд хэрхэн хоорондоо хамтарч ажилладаг тухай.
- X11-г хэрхэн суулгаж тохируулах.
- Төрөл бүрийн цонхот орчныг хэрэглэх.
- TrueType® төрлийн үсгийн маягийг X11 дээр хэрхэн хэрэглэх.
- Өөрийнхөө системийг график горимд хэрхэн нэвтрэлт хийлгэх тухай (XDM).

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- Гуравдагчдын бүтээсэн програмыг хэрхэн суулгах тухай мэддэг байх хэрэгтэй (Програм суулгах. Багцууд болон портууд).

## 6.2. X-г таньж мэдэх

Microsoft® Windows® болон Mac OS® зэрэг график горимыг урьд нь хэрэглэдэг байсан зарим хүмүүс анх удаагаа X-г хэрэглэх үед нэгэн төрлийн цочролд автаж магадгүй юм.

X-н төрөл бүрийн бүрдлүүд хоорондоо хэрхэн хамтарч ажилладаг талаар бүрэн мэдэх шаардлагагүй ч гэсэн түүний тухай үндсэн ойлголт авсан үед X-н хүчин чадлыг зөв ашиглаж болох юм.

### 6.2.1. Яагаад X гэж?

X нь UNIX®-д зориулж бүтээсэн цонхот системийн анхдагч нь биш ч хамгийн их түгсэн нь мөн билээ. X-н анхны хөгжүүлэгчид нь X-г бичихээсээ өмнө өөр цонхот систем дээр ажиллацгааж байсан юм. Тэр системийг нь "W" ("Window" гэсэн үг) гэж нэрлэдэг байжээ. X нь ердөө л Ром үсгийн дараагийн үсэг байв.

X нь заримдаа "X", "X Window System буюу X цонхот систем", "X11" гэх мэтчилэн янз бүрээр нэрлэгддэг. Магадгүй X11-г "X Цонх" гэж нэрлэсэн хүмүүс их байгааг анзаарч магадгүй юм. Дэлгэрэнгүй мэдээлэл X(7) дээр буй.

### 6.2.2. X үйлчлүүлэгч/сервер загвар

X нь анхнаасаа сүлжээтэй ажиллана гэж тооцож зохиогдсон бөгөөд "үйлчлүүлэгч-сервер" загвар хэрэглэдэг.

X загварт "X сервер" нь хулгана, гар, дэлгэцтэй ажилладаг. Энэ үед серверийн үүрэг бол дэлгэцийг удирдах, гар ба хулгана болон бусад оролт эсвэл гаралтын төхөөрөмжүүдээс (жишээ нь "tablet" нь оролтын төхөөрөмж маягаар ашиглагдаж видео проектор нь гаралтын төхөөрөмж байж болох юм) мэдээллийг хүлээж аван зохицуулах үйлдлүүдийг хариуцаж ажиллаж байдаг. X програм болгон (XTerm эсвэл Firefox гэх мэт) уг серверийн "үйлчлүүлэгч" нь юм. Үйлчлүүлэгч нь сервер уруу жишээлбэл "энэ байрлалд цонх зурна уу" гэх мэтийн хүсэлт илгээж болдог байхад серверээс үйлчлүүлэгч рүү жишээлбэл "Хэрэглэгч ОК товч дээр дарсан шүү" гэх мэтийн мэдэгдэл буцааж илгээж байдаг.

Гэртээ юм уу эсвэл жижигхэн албан байгууллагын орчинд X сервер болон X үйлчлүүлэгч нь нэг компьютер дээр ажиллаж байдаг. Гэхдээ X серверийг багахан чадалтай компьютер дээр ажиллуулаад харин X програмуудыг арай хүчтэй компьютер (үйлчлүүлэгч) дээр ажиллуулбал хамгийн тохирсон албаны хэрэглээ байж болох билээ. Ийм нөхцөлд X сервер болон үйлчлүүлэгчийн хоорондох холболт сүлжээгээр хийгдэнэ.

Ийм хэрэглээ нь X-г өөрөөр хэрэглэнэ гэж бодож байсан зарим хүмүүсийг гайхшруулж магадгүй юм. Тийм хүмүүс "X серверийг" маш хүчтэй мундаг эд байх ёстой ба бага чадалтай нь "X үйлчлүүлэгчийн" үүрэг гүйцэтгэж ширээн дээр байх ёстой гэж бодсон байдаг.

X сервер нь дэлгэц болон гар залгагдсан компьютер нь бөгөөд X үйлчлүүлэгч нь дэлгэц дээр цонх дүрсэлдэг програмууд шүү гэдгийг санах нь тун чухал.

Сервер болон үйлчлүүлэгч хоёрыг нэг төрлийн компьютер юм уу эсвэл нэг төрлийн үйлдлийн систем дээр ажиллах ёстой гэсэн ямар ч албадсан заавар байдаггүй. X серверийг Microsoft® Windows® эсвэл Apple-н Mac OS® дээр ажиллуулж болдог бөгөөд иймэрхүү үйлдлийг гүйцэтгэж байдаг маш олон үнэгүй юм уу худалдааны програмууд байдаг.

### 6.2.3. Цонх зохицуулагч

X-н загварын зарчим нь UNIX®-н "хэрэгсэл болохоос бодлого биш" гэсэн зарчимтай тун ижил байдаг. Энэ нь юу гэсэн үг вэ гэхээр, X нь үйлдлүүд хэрхэн биелэгдэх ёстой талаар ямар ч бичлэг хийдэггүй. Харин түүний оронд хэрэглэгчдэд зориулсан хэрэгслүүдээр хангагдсан байдаг бөгөөд тэдгээрийг яаж хэрэглэх нь хэрэглэгчийн үүрэг юм.

Энэ зарчмын дагуу X нь цонхнуудыг хэрхэн харагдах, хулгана хэрхэн хөдлөх, ямар товч хэрэглэж цонхнуудын хооронд дамжих (Microsoft® Windows® дээрх `Alt` + `Tab` товчлуурын хослол шиг), цонх болгоны дээрх гарчгийн самбар яаж харагдах, цонх болгонд хаах товч байх ёстой юу үгүй гэх зэргийн бүртгэл болон зааврууд агуулж байдаггүй.

Харин түүний оронд иймэрхүү хариуцлагыг X нь "Цонх зохицуулагч" гэж нэрлэгдэх програм руу дамжуулдаг. X-д зориулсан олон цонх зохицуулагчид байдаг. Эдгээр цонх зохицуулагч болгон тус тусдаа өөрсдийн өөрийн гэсэн төрөл бүрийн харуулах загвартай байдаг ба зарим нь өөртөө "virtual desktops буюу хийсвэр компьютерийн дэлгэцийг" дэмждэг бөгөөд тэд нарын зарим нь тийм хийсвэр дэлгэцэндээ сэлгэх товчлуурууд агуулж байхад зарим нь "Start буюу Эхлэх" товч юм уу түүнтэй ижил төхөөрөмж агуулж байдаг. Мөн зарим нь

"themeable буюу дэлгэцийн дурын өөрчлөлт" хийж харагдах загваруудын бүрдлийг агуулан сэлгэж хэрэглэх зэргээр хэрэглэгдэж байдаг билээ. Цонх зохицуулагчдыг портын цуглуулга доторх x11-wm төрөлд олж болно.

Мөн түүнчлэн KDE болон GNOME дэлгэцийн орчнууд нь өөрсдийн гэсэн цонх зохицуулагчтай байдаг ба тэр нь дэлгэцийн орчиндоо агуулагдчихсан ажиллаж байдаг.

Цонх зохицуулагч бүр өөр өөрсдийн тохируулах аргатай байдаг. Зарим нь тохируулгын файлыг гараараа бичих ёстой гэдэг байхад зарим нь ихэнх тохируулгын үйлдлүүдийг GUI буюу график орчны хэрэгслээр хийдэг; мэдэж байгаагаар, нэг цонх зохицуулагч (Sawfish) Lisp програмын хэллэгээр бичигдсэн тохиргооны файлтай байдаг.

*Сонгогдож идэвхжүүлэх бодлого*

Цонх зохицуулагчийн бас нэг үүрэг бол "сонгогдож идэвхжих бодлого " билээ. Цонх болгон ямар нэгэн байдлаар сонгогдсоноо илэрхийлж идэвхжин гарнаас оруулсан мэдээллийг хүлээж авах бөгөөд идэвхтэй болсноо нүдэнд харуулагдахаар дүрсэлж харуулах хэрэгтэй байдаг.

Түгээмэл болсон, сонгогдож идэвхжих бодлогыг "идэвхжүүлэхийн-тулд-дарах" гэж нэрлэдэг. Энэ загварыг Microsoft® Windows® хэрэглэдэг бөгөөд хулгана дарах үед заагдсан цонх нь идэвхитэй болно.

Х ямар нэгэн сонгож идэвхжүүлэх бодлогыг дэмждэггүй. Харин түүний оронд цонх зохицуулагч гуай тухайн үед ямар цонх идэвхжих вэ гэдгийг хянаж байдаг. Цонх зохицуулагч бүр өөр өөрсдийн сонгож идэвхжүүлэх арга хэрэглэдэг. Тэд бараг бүгдээрээ дарж идэвхжүүлэх бодлого баримталж байхад тун цөөхөн нь өөр зарчим баримталсан байдаг.

Хамгийн түгээмэл сонгож идэвхжүүлэх бодлогууд бол:

**хулганыг дагаж идэвхжүүл**

Хулганы заагч байгаа цонхыг сонгогдсон гэж авч үздэг. Цонхыг хамгийн өмнө нь байлгах шаардлагагүй бөгөөд нэмж товч даралгүй хулганы байрлалыг өөрчилснөөр цонхыг идэвхжүүлж болно.

**унтамхай-идэвхжүүлэлт**

Энэ бодлого бол хулгана-дагаж-идэвхжүүл бодлогын өргөжүүлсэн арга юм.хулгана-дагаж-идэвхжүүл бодлого дээр хэрэв хулгана эх цонхон дээр (эсвэл арын дэвсгэр) очвол ямар ч цонх идэвхтэй болдоггүй. Харин унтамхай-идэвхжүүлэлт дээр хэрэв хулгана өөрийнхөө цонхноос гараад өөр цонх уруу орсон үед тэр цонх идэвхтэй болдог.

**идэвхжүүлэхийн-тулд-дарах**

Хулганы товч дарахад л тухайн цонх идэвхжинэ. Уг цонх тэгээд "дэлгэгдэх" бөгөөд бүх цонхны өмнө харуулагддаг. Ингээд дарагдсан бүх товчнуудын өгөгдлүүд энэ цонх уруу илгээгдэж байдаг.

Маш олон цонх зохицуулагчид өөр бодлого дэмжих эсвэл жаахан өөрчилсөн бодлого хэрэглэдэг. Тухайн цонх зохицуулагчийн бичиг баримтаас нь

нэмэгдэл мэдээлэл аваарай.

### 6.2.4. Widgets буюу багажнууд

X-н хэрэгсэл болохоос бодлого биш зарчмыг өргөтгөн програм болгоныг төлөөлсөн багаж дэлгэц дээр харуулагддаг.

"Багажнууд" гэдэг нь дарагдах, эсвэл ямар нэгэн аргаар өөрчлөгдөж байдаг дэлгэц дээрх зүйлс бөгөөд товч, чагталдаг цонх, бөөрөнхий сонгодог товч, тэмдэгт зурагнууд, жагсаалтууд зэрэг юм. Microsoft® Windows® харин тэдгээрийг "controls буюу хянагчид" гэж нэрлэсэн байдаг.

Microsoft® Windows® болон Apple-н Mac OS® хоёр багажны маш тогтсон загвартай билээ. Програм хөгжүүлэгчид нь өөрсдийн програмуудаа нийтлэг харуулагдах загвараар аль болох хийх зарчмыг баримталж байдаг. X-ийн хувьд бол тодорхой нэг график загвар, аль эсвэл багажнуудын олонлогийг заавал баримтлах нь тийм ч ухаалаг бус санаа юм.

Тийм болохоор X програмуудыг хоорондоо адилхан харагдах нийтлэг загвартай байдаг гэж бодох хэрэггүй. Түгээмэл дэлгэрсэн багажны цуглуулгууд байдаг бөгөөд үүнд KDE-н хэрэглэдэг Qt, аль эсвэл GNOME-н хэрэглэдэг GTK+ зэрэг орчин үеийн багажнуудын олонлогийг дурдаж болно. Энэ нь UNIX®-н харагдах дэлгэцийн маягийн хувьд зарим нэг давхцалд хүргэх бөгөөд ингэснээр мэдээж юмсыг шинэ хэрэглэгчдийн хувьд илүү хялбар болгох юм.

## 6.3. X11-г суулгах нь

Xorg бол FreeBSD-н X11 гүйцэтгэл юм. Xorg нь X.Org сангаас гаргасан X цонхот системийн X сервер. Xorg нь XFree86™ 4.4RC2 болон X11R6.6 хоёрын эх бичлэг дээр үндэслэгдсэн билээ. FreeBSD-н портын цуглуулга дотор байгаа Xorg-н хувилбар нь 7.7 гэж буй.

Портын цуглуулгаас Xorg-г суулгахын тулд:

```
# cd /usr/ports/x11/xorg
# make install clean
```

Xorg-г бүхлээр нь хөрвүүлэхийн тулд хамгийн багадаа 4 ГБ хэмжээ сул байлгах хэрэгтэйг анхааараарай.

X11-г багц хэлбэрээр мөн суулгаж болно. Хоёрт хэлбэрийн файлыг нь pkg_add(1) хэрэгслээр X11-г суулгаж болдог. Алсад байгаа серверээс нөхөж суулгадаг боломжийг pkg_add(1) хэрэглэх үед багцнаас хувилбарынх нь дугаарыг устгах хэрэгтэй. Тэгээд pkg_add(1) нь автоматаар сүүлийн хувилбарыг нь нөхөж суулгадаг.

Xorg-н хамгийн сүүлийн хэлбэрийг нь автоматаар нөхөж суулгахын тулд ердөө л дараах тушаалыг өгнө:

```
# pkg_add -r xorg
```

Дээрх жишээгээр бол X11-г сервер, үйлчлүүлэгч, үсгийн маяг зэрэгтэй нь бүхлээр нь суулгана. X11-н багц болон портууд нь тусдаа мөн суугдаж болдог.

X11 түгээлтийг хамгийн бага хэлбэрээр суулгахын тулд x11/xorg-minimal-г та суулгаж болно.

Энэ бүлгийн үлдсэн хэсэгт X11-г суулгаж тохируулан хэрхэн эвтэйхэн компьютерийн дэлгэц бэлдэх талаар өгүүлэх болно.

# 6.4. X11-н тохируулга

## 6.4.1. Тохируулж эхлэхээс өмнө

Ихэнх тохиолдолд X11 нь өөрөө тохируулагддаг. Хуучин болон ердийн биш төхөөрөмжтэй хүмүүс тохируулахаасаа өмнө тоног төхөөрөмжийн талаар мэдээлэл цуглуулах нь хэрэгтэй байж магадгүй юм.

- Дэлгэцийн давтамжууд
- Видео картын схемийн мэдээлэл
- Видео картын санах ойн хэмжээ

Дэлгэцийн нягтрал ба зурах давтамж нь мониторын хэвтээ ба босоо давтамжуудаас хамаардаг. Бараг бүх мониторууд эдгээрийг электроноор автоматаар илрүүлэх дэмжлэгтэй байдаг. Цөөн мониторууд эдгээр утгуудгүй байдаг бөгөөд хэвлэсэн гарын авлага болон үйлдвэрлэгчийн вэб сайтаас үзүүлэлтүүдийг олж авах хэрэгтэй.

Видео картын бичих схемийг бас автоматаар илрүүлдэг бөгөөд энэ нь зөв видео драйвер сонгоход ашиглагддаг. Хүссэн үр дүнд автоматаар илрүүлэх явц хүргэхгүй бол хэрэглэгчийн хувьд ямар бичил схем суулгагдсан болохыг мэдэх нь хэрэг болдог.

Видео картын санах ой максимум нягтрал ба үзүүлж болох өнгөний гүнг тодорхойлдог.

## 6.4.2. X11-н тохиргоо

Гар болон хулганыг автоматаар танихын тулд Xorg нь HAL-ийг ашигладаг. sysutils/hal болон devel/dbus портууд нь x11/xorg-ийн хамаарал болон суудаг боловч тэдгээрийг /etc/rc.conf файл дахь дараах оруулгуудыг засварлан идэвхжүүлэх ёстой:

```
hald_enable="YES"
dbus_enable="YES"
```

Эдгээр үйлчилгээнүүдийг дараа дараачийн Xorg тохиргоог оролдох эсвэл ашиглахаасаа өмнө эхлүүлсэн байх шаардлагатай (гараар эсвэл дахин ачаалах замаар).

Xorg нь ямар нэг тохиргооны алхмуудыг цааш хийлгүйгээр дараах тушаалыг бичин ихэвчлэн ажилладаг:

```
% startx
```

Автомат тохиргоо нь зарим нэг тоног төхөөрөмжийн хувьд ажиллахгүй байх юм уу эсвэл зарим нэг зүйлсийг хүссэнээр тэр бүр тохируулж чаддаггүй. Энэ тохиолдлуудад гараар тохиргоог хийх шаардлагатай.

GNOME, KDE эсвэл Xfce зэрэг ширээний орчнууд нь хэрэглэгчид нягтрал зэрэг дэлгэцийн өгөгдлүүдийг хялбарханаар тохируулах боломж бүхий хэрэгслүүдтэй. Хэрэв анхдагч тохиргоог хүлээн авах боломжгүй бөгөөд та ширээний орчин суулгахаар шийдсэн бол ширээний орчны суулгалтыг үргэлжлүүлж тохирох дэлгэц тохируулах хэрэгслийг ашиглаарай.

X11-г тохируулах нь олон үе шаттай явц билээ. Хамгийн эхний алхам бол анхдагч тохируулгын файлыг бэлдэх билээ. Супер хэрэглэгчийн эрхээр ердөө л дараах тушаалыг өгнө:

```
# Xorg -configure
```

Ингэснээр /root сан дотор xorg.conf.new гэсэн нэртэй X11 ийн тохиргооны үндсэн файл үүсгэгддэг (та su(1) тушаалаар юм уу эсвэл анхнаасаа супер хэрэглэгчийн эрхээр орсон байсан ч супер хэрэглэгчийн $HOME буюу эх сан дотор үүсгэх болно). X11 гуай тухайн систем дээр байгаа график төхөөрөмжийг таних оролдлого хийж таниулах програмын мэдээллийг уг тохируулга руу бичдэг.

Дараагийн алхам бол уг тохируулга файлыг ашиглаад график төхөөрөмжтэй Xorg ажиллаж чадаж байгааг турших юм. Дараах тушаалыг оруулах хэрэгтэй:

```
# Xorg -config xorg.conf.new -retro
```

Хэрэв хар саарал дэвсгэр дээр X хэлбэртэй хулганы заагч харагдаж байвал амжилттай боллоо гэсэн үг. Туршилтаас гарахын тулд эхлүүлсэн виртуал консол руу `Ctrl` + `Alt` + `Fn` (эхний виртуал консолын хувьд `F1`) товчлууруудын хослолыг дарж шилжин `Ctrl` + `C` дарна.

`Ctrl` + `Alt` + `Backspace` товчлууруудын хослолыг ашиглан Xorg-с бас гарч болно. Үүнийг идэвхжүүлэхийн тулд дараах тушаалыг дурын X терминал эмуляторооc оруулж болно:

```
% setxkbmap -option terminate:ctrl_alt_bksp
```

Эсвэл hald-д зориулсан гарын тохиргооны файл x11-input.fdi-г /usr/local/etc/hal/fdi/policy санд үүсгэн хадгалж болно. Энэ файл нь дараах

мөрүүдтэй байх ёстой:

```
<?xml version="1.0" encoding="utf-8"?>
<deviceinfo version="0.2">
  <device>
    <match key="info.capabilities" contains="input.keyboard">
      <merge key="input.x11_options.XkbOptions"
type="string">terminate:ctrl_alt_bksp</merge>
    </match>
  </device>
</deviceinfo>
```

Та hald-гээр энэ файлыг уншуулахын тулд өөрийн машиныг дахин ачаалах хэрэгтэй болно.

Дараах мөрийг xorg.conf.new файлын `ServerLayout` эсвэл `ServerFlags` хэсэгт нэмж өгөх ёстой болно:

```
Option  "DontZap"   "off"
```

Хэрэв хулгана ажиллахгүй байвал өмнөх туршилтыг дахин хийхээсээ өмнө хулганаа тохируулах шаардлагатай. FreeBSD-г суулгах бүлэгт буй Хулганы тохиргоо хэсгээс харна уу. Сүүлийн үеийн Xorg-с эхлээд xorg.conf файл дахь `InputDevice` хэсгүүдээс автоматаар илрүүлсэн төхөөрөмжүүдийг илүүд үздэг. Хуучин байдалд нь эргүүлж оруулахын тулд дараах мөрийг энэ файлын `ServerLayout` эсвэл `ServerFlags` хэсэгт нэмэх хэрэгтэй:

```
Option "AutoAddDevices" "false"
```

Оролтын төхөөрөмжүүдийг дараа нь урьдны хувилбаруудад хийсний нэгэн адилаар хэрэгцээтэй бусад тохируулгуудын (жишээ нь гарын байрлалыг сэлгэх) хамтаар тохируулж болдог.

Өмнө тайлбарласны адил анхдагчаар hald демон таны гарыг автоматаар мэдэх болно. Таны гарын байрлал эсвэл загвар зөв биш байж болох боловч GNOME, KDE эсвэл Xfce зэрэг ширээний орчнууд гарыг тохируулах хэрэгслүүдтэй байдаг. Гэхдээ гарын тохиргоог setxkbmap(1) хэрэгсэл эсвэл hald-ийн тохиргооны дүрмийн тусламжтайгаар шууд тохируулах боломжтой байдаг.

Жишээ нь хэрэв Франц байрлалтай PC 102 товчлууртай гарыг хэн нэгэн ашиглахыг хүсэж байгаа бол hald-д зориулсан x11-input.fdi гэсэн гарын тохиргооны файлыг үүсгэж /usr/local/etc/hal/fdi/policy санд хадгалах хэрэгтэй. Энэ файл нь дараах мөрүүдийг агуулсан байх ёстой:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<deviceinfo version="0.2">
  <device>
    <match key="info.capabilities" contains="input.keyboard">
      <merge key="input.x11_options.XkbModel"
type="string">pc102</merge>
      <merge key="input.x11_options.XkbLayout" type="string">fr</merge>
    </match>
  </device>
</deviceinfo>
```

Хэрэв энэ файл байгаа бол хуулаад гарын тохиргооны мөрүүдийг өөрийн файлдаа нэмэх хэрэгтэй.

hald-ээр энэ файлыг уншуулахын тулд та өөрийн машиныг дахин ачаалах хэрэгтэй.

Адил тохиргоог X терминалаас эсвэл энэ тушаалын мөрөөс скриптээр хийх боломжтой:

```
% setxkbmap -model pc102 -layout fr
```

/usr/local/shared/X11/xkb/rules/base.lst файл нь төрөл бүрийн боломжит гар, байрлал, сонголтуудыг харуулдаг.

xorg.conf.new файлыг одоо өөрийнхөө хүсэлд тохируулан сайжруулах хэрэгтэй. Уг файлыг emacs(1) эсвэл ee(1) зэрэг засварлагчаар нээнэ. Хэрэв монитор хуучны эсвэл давтамжийг автоматаар мэдэрдэг ердийнх шиг биш бол xorg.conf.new файл дотор "Monitor" гэсэн хэсэгт оруулж өгөөрэй:

```
Section "Monitor"
        Identifier   "Monitor0"
        VendorName   "Monitor Vendor"
        ModelName    "Monitor Model"
        HorizSync    30-107
        VertRefresh  48-120
EndSection
```

Ихэнх мониторууд давтамжийг автоматаар мэдэрдэг бөгөөд гараар оруулах заавал шаардлагагүй. Автоматаар мэдэрдэг дэмжлэг байхгүй шинэ мониторуудын хувьд үйлдвэрлэгчээс зөвхөн зааж өгсөн утгуудыг оруулж эвдрэлээр сэргийлээрэй.

X нь DPMS (Тэжээл хэмнэх) чадвартай дэлгэцийг дэмждэг. xset(1) програм цаг дуусах нөхцлийг шалгаад дэлгэцийг standby, suspend, эсвэл off буюу нөөц, түр салгах, эсвэл унтраах зэрэг горимд шилжүүлдэг. Хэрэв та дэлгэцэндээ DPMS шинж чанарыг хэрэгжүүлнэ гэж бодож байвал дараах мөрийг monitor хэсэгт нэмэх хэрэгтэй:

```
        Option      "DPMS"
```

Та xorg.conf.new файлыг засварлагч дээр нээсэн хэвээр байгаа болохоор дэлгэцийн зуралтын хэмжээ болон өнгөний нягтыг оруулж өгч болно. Эдгээр утгыг "Screen" хэсэгт оруулдаг:

```
Section "Screen"
        Identifier "Screen0"
        Device     "Card0"
        Monitor    "Monitor0"
        DefaultDepth 24
        SubSection "Display"
                Viewport  0 0
                Depth     24
                Modes     "1024x768"
        EndSubSection
EndSection
```

DefaultDepth-д өгсөн утга нь анхдагч хэрэглэх өнгөний нягт юм. Хэрэв анхдагч утгыг нь ачаалах үед дарж өөрчлөж хэрэглэнэ гэвэл Xorg(1)-г ажиллуулах тушаалд -depth гэсэн сонголтыг хүссэн утгатай хамт оруулан хэрэглэж болно. Modes гэсэн түлхүүр үг нь заасан өнгөтэй үеийн дэлгэцийн зурах хэмжээг зааж байдаг. Тухайн график дүрслэгчийн үйлдвэрлэгчээс зааж өгсөн стандарт VESA горимууд л зөвшөөрөгдсөн гэдгийг анхаарна уу. Дээрх жишээн дээр дэлгэцийн анхдагч өнгөний нягт нь нэг цэгийг хорин дөрвөн битээр илэрхийлнэ гэж заасан байгаа бөгөөд ийм нягтшилтай үедээ өргөөшөө 1024, өндрөөшөө 768 цэгээр зурж харуул хэмээн зааж өгөв.

Эцэст нь тохируулгын файлаа хадгалаад өмнө заасны дагуу дахин турших хэрэгтэй.

Хүндрэл гарсан үед танд хэрэгтэй нэг хэрэгсэл бол X11-ийн log буюу бүртгэл тэмдэглэлийн файл юм. Энэ файлд X11-д залгагдсан төхөөрөмжийн тухай мэдээллүүд оршиж байдаг. Xorg-н бүртгэл тэмдэглэлийн нэрний загвар нь /var/log/Xorg.0.log хэлбэртэй байдаг. Энэ файлын жинхэнэ нэр нь Xorg.0.log -оос Xorg.8.log-н хооронд өөрчлөгдөж байдаг.

Хэрэв бүх зүйл сайн болж өнгөрсөн бол тохируулгын файлаа Xorg(1)-д олдох газарт байрлуулах хэрэгтэй. Үүнийг голдуу /etc/X11/xorg.conf эсвэл /usr/local/etc/X11/xorg.conf гэсэн байрлалд хуулдаг.

```
# cp xorg.conf.new /etc/X11/xorg.conf
```

X11-г тохируулах явц ингээд гүйцлээ. Xorg-г та одоо startx(1) хэрэгслээр эхлүүлж болно. X11 сервер нь мөн xdm(1)-р эхлүүлэгдэж болдог.

### 6.4.3. Нэмэгдэл тохиргоо

#### 6.4.3.1. Intel® `i810` Graphics Chipsets-г тохируулах

Intel® i810 integrated chipsets төрлийн график дүрслэгчдийг тохируулахын тулд agpgart AGP програмууд X11-д хэрэгтэй байдаг. agp(4)-н драйверийн гарын авлагаас нэмэгдэл мэдээллийг хараарай.

Энэ нь мөн бусад график хавтангуудын төхөөрөмжийг тохируулахад хэрэглэгддэг. Системийнхээ цөмд agp(4) драйверийг оруулж хөрвүүлээгүй бол kldload(8) тушаалаар драйверийг дуудсан үед ажиллахгүй гэдгийг анхаарна уу. Уг драйвер анхнаасаа цөмд оруулж бэлтгэгдсэн байх ёстой юм уу эсвэл ачаалах үед /boot/loader.conf-р цөмд нэмэгдсэн байх ёстой.

#### 6.4.3.2. Өргөн дэлгэцийн горим нэмэх

Энэ хэсэгт таныг тохируулгын жаахан илүү мэдлэгтэй гэж үзэх болно. Мөн энэ хэсэгт дээр өгүүлсэн ердийн тохиргоо хийхийг авч үзэхгүй. Бүртгэл мэдээллийн файлыг ашиглаж тохиргоог ямар нэгэн аргаар дуусгаж болно. Хамгийн багадаа текст засварлагч байхад л уг тохируулгыг хийж болох хангалттай.

16:10 болон 10:9 зэрэг харуулах харьцааг дэмждэг одоогийн өргөн дэлцгэцүүд (WSXGA, WSXGA+, WUXGA, WXGA, WXGA+, гэх мэт) жаахан хүндрэлтэй байж болзошгүй. Зарим төрлийн 16:10 харуулах харьцаануудыг дурдвал:

- 2560x1600
- 1920x1200
- 1680x1050
- 1440x900
- 1280x800

Зарим тохиолдолд `Section "Screen"` хэсгийн `Mode` хэсэгт эдгээр харьцаануудын нэгийг бичсэнээр тохиргоог амархан гүйцээж болдог:

```
Section "Screen"
Identifier "Screen0"
Device     "Card0"
Monitor    "Monitor0"
DefaultDepth 24
SubSection "Display"
    Viewport  0 0
    Depth     24
    Modes     "1680x1050"
EndSubSection
EndSection
```

Xorg нь тухайн өргөн дэлгэцээс I2C/DDC мэдээллийг нь авах ухаантай болохоор уг дэлгэцийн харгалзах зуралтын давтамжийг нь мэдэж чаддаг.

Хэрэв эдгээр `ModeLines`-ууд нь драйвер дотор байхгүй байвал Xorg танд жаахан тусламж өгч

чаддаг. /var/log/Xorg.0.log файлаас ажиллаж болох ModeLine-уудыг харж болно. Дараах шиг мөрийн хэсгүүдийг харах хэрэгтэй:

```
(II) MGA(0): Supported additional Video Mode:
(II) MGA(0): clock: 146.2 MHz   Image Size:  433 x 271 mm
(II) MGA(0): h_active: 1680  h_sync: 1784  h_sync_end 1960 h_blank_end 2240 h_border:
0
(II) MGA(0): v_active: 1050  v_sync: 1053  v_sync_end 1059 v_blanking: 1089 v_border:
0
(II) MGA(0): Ranges: V min: 48  V max: 85 Hz, H min: 30  H max: 94 kHz, PixClock max
170 MHz
```

Энэ мэдээллийг EDID мэдээлэл гэдэг. Үүгээр ModeLine мөрийг үүсгэхийн тулд ердөө л тэдгээрийг зөв дарааллаар нь оруулж бичихэд хангалттай:

```
ModeLine <name> <clock> <4 horiz. timings> <4 vert. timings>
```

Тэгэхээр дээр жишээний дагуу Section "Monitor" хэсэгт ModeLine мөрийг дараах маягаар оруулна:

```
Section "Monitor"
Identifier      "Monitor1"
VendorName      "Bigname"
ModelName       "BestModel"
ModeLine        "1680x1050" 146.2 1680 1784 1960 2240 1050 1053 1059 1089
Option          "DPMS"
EndSection
```

Ингэж энгийн засварлалт хийж дуусгаад X-ийг өргөн дэлгэцтэй нь шинээр ажиллуулж болно.

# 6.5. X11 дээр үсгийн маяг хэрэглэх нь

## 6.5.1. Type1 төрлийн үсгийн маяг

X11 тэй хамт ирдэг анхдагч үсгийн маягууд нь ширээний програмуудад тийм сайн тохиромжтой байж чаддаггүй. Том үсэгнүүд нь зарим нь арзайж харагдах, зарим getenv(3)-н жижиг үсэгнүүд тийм аятайхан харагддаггүй гэх мэт онцгүй тохиолдлууд тулгардаг. Гэхдээ X11 дээр хэрэглэгдэх боломжтой үнэгүй өндөр чанарын Type1 (PostScript®) төрлийн үсгийн маягууд байдаг. Жишээ нь URW үсгийн цуглуулганд (x11-fonts/urwfonts) өндөр чанарын type1 (Times Roman™, Helvetica™, Palatino™ гэх мэтчилэн) үсгийн маягууд орсон байдаг. Мөн Freefonts цуглуулганд (x11-fonts/freefonts) маш олон төрлийн үсгийн маяг байдаг бөгөөд ихэнх нь графиктай ажилладаг Gimp зэрэг програмд зориулагдсан болохоор дэлгэцийн харуулах зориулалтанд хэрэглэгддэггүй. Бас X11 дээр цөөхөн тохиргоо хийгээд TrueType® төрлийн үсгийн маяг хэрэглэдэг болгож болдог. X(7) хуудаснаас юм уу эсвэл TrueType®

үсгийн маягийн хэсгээс нэмэлт мэдээллүүдийг уншина уу.

Дээрх Type1 үсгийг портын цуглуулгаас суулгахын тулд дараах тушаалыг оруулна:

```
# cd /usr/ports/x11-fonts/urwfonts
# make install clean
```

Энэ үйлдлийг бусад цуглуулган дээр мөн адил хийнэ. X серверт эдгээр үсгээ таниулахын тулд тохируулгын файлд (/etc/X11/xorg.conf) дараах мөрийг нэмж өгдөг:

```
FontPath "/usr/local/lib/X11/fonts/URW/"
```

эсвэл X ажиллаж байх үед дараах тушаалыг өгч болно:

```
% xset fp+ /usr/local/lib/X11/fonts/URW
% xset fp rehash
```

Энэ тушаал нь X хэсгийг хаагдах хүртэл хүчинтэй бөгөөд ийм байдалд хүргэхгүй гэвэл ачаалах үед уншигддаг эхлэх файлд нэмж өгдөг (`startx`-н эхлэх үед уншдаг файл нь ~/.xinitrc, харин XDM зэргийн график нэвтрэлт хийдэг програмын эхлэн уншдаг файл нь ~/.xsession байдаг ). Гурав дахь арга бол шинэ /usr/local/etc/fonts/local.conf файл хэрэглэх юм: anti-aliasing хэсгээс уншина уу.

### 6.5.2. TrueType® үсгийн маяг

Xorg нь өөртөө бас TrueType® төрлийн үсгийн маягийг харуулах гүйцэтгэлтэй суусан байдаг. Үүнийг гүйцэтгэх хоёр төрлийн гүйцэтгэл бий. Энэ хэсэгт freetype модулийг харуулсан ба нөгөө үсгийн маяг харуулагчийг бодвол илүү тогтвортой билээ. freetype модулийг идэвхжүүлэхийн тулд дараах мөрийг /etc/X11/xorg.conf файлын `"Module"` хэсэгт нэмнэ.

```
Load  "freetype"
```

Одоо TrueType® үсгийн маягт зориулсан сан үүсгээд (жишээлбэл /usr/local/lib/X11/fonts/TrueType) бүх TrueType® үсгүүдээ тэр санд хуулна. Macintosh® машинаас TrueType® төрлийн үсгийн маягийг шууд хуулж болохгүй гэдгийг анхаарах хэрэгтэй. Эдгээр нь X11 дээр хэрэглэгдэхийн тулд UNIX®/MS-DOS®/Windows® хэлбэрийнх байх ёстой. Тэр санд хуулсныхаа дараа ttmkfdir-г ашиглаж fonts.dir файл үүсгэдэг бөгөөд ингэснээр X-н үсэг харуулагч нь шинэ файл суугдлаа гэдгийг таньдаг. `ttmkfdir` нь FreeBSD-н портын цуглуулганд x11-fonts/ttmkfdir гэж буй.

```
# cd /usr/local/lib/X11/fonts/TrueType
# ttmkfdir -o fonts.dir
```

Одоо TrueType® санг үсгийн маягийн таних замд нэмэх ёстой. Энэ нь дээр Type1 үсгийн

маягт дээр өгүүлсэнтэй ижилхэн

```
% xset fp+ /usr/local/lib/X11/fonts/TrueType
% xset fp rehash
```

гэж нэмэх юм уу эсвэл xorg.conf файл дотор `FontPath` гэж нэмнэ.

Ингээд болох нь тэр. Одоо getenv(3), Gimp, StarOffice™, гэх мэтчилэн бүх X програмууд суугдсан TrueType® үсгийн маягуудыг таних ёстой. Маш бага хэмжээтэй (өндөр нарийвчлалтай дэлгэц дээр үзүүлэгдэх вэбийн текстүүд) эсвэл маш том үсэгнүүд (StarOffice™ дээр) арай илүү харагдана.

### 6.5.3. Anti-Aliased үсгийн маяг

X11-ийн /usr/local/lib/X11/fonts/ ба ~/.fonts/ дотор байгаа бүх үсгийн маягууд автоматаар anti-aliasing гэж Xft-нийцтэй програмд хэрэглэгдэх боломжтой болсон. KDE, GNOME, Firefox зэрэг хамгийн сүүлийн үеийн Xft-нийцтэй програмууд байдаг.

Ямар үсгийн маягууд нь anti-aliased болохыг хянах эсвэл anti-aliasing шинж чанаруудыг тохируулахын тулд /usr/local/etc/fonts/local.conf файлыг үүсгэх (хэрэв байвал засварлах) хэрэгтэй. Xft үсгийн маягийн системийн хэд хэдэн шинж чанарууд энэ файлаар тохируулагддаг бөгөөд энэ хэсэгт зөвхөн энгийн хэдийг жишээ татах болно. Дэлгэрэнгүй мэдээллийг fonts-conf(5) хуудаснаас харна уу.

Энэ файл нь XML хэлбэрийн байх ёстой. Том жижиг үсгийн хэмжээнд нь анхааралтай хандах хэрэгтэй. Мөн нээгдсэн таглааг бас зөв хаасан байх ёстой. Энэ файл эхлэхдээ энгийн DOCTYPE тодорхойлолтоор эхэлдэг бөгөөд дараа нь `<fontconfig>` таглаа араас нь залгаж явдаг:

```
      <?xml version="1.0"?>
      <!DOCTYPE fontconfig SYSTEM "fonts.dtd">
      <fontconfig>
```

Өмнө хэлсэнчлэн /usr/local/lib/X11/fonts/ болон ~/.fonts/ санд байгаа үсгийн маягууд автоматаар Xft-нийцтэй програмд танигддаг. Та үүнээс өөр санд үсгийн маяг байрлуулсан бол /usr/local/etc/fonts/local.conf файлд доор дурдсантай ижил мөр оруулж өгөх хэрэгтэй:

```
<dir>/үсгийн/маягийн/байгаа/сангийн</зам>
```

Шинэ үсэг, ялангуяа шинэ сан нэмсний дараа үсгийн маягийн түр хадгалагдсан орон зайг дараах тушаалаар сэргээх ёстой:

```
# fc-cache -f
```

Anti-aliasing нь ирмэгийг жаахан уусгаж зөөлрүүлэн харагдуулдаг болохоор жижиг үсгүүдийг уншихад эвтэйхэн болгодог бөгөөд том үсгийн "шаталсан ирмэгийг" зөөллөж

гөлийлгөдөг. Иймэрхүү зөөллөсөн шинэ чанар нь ердийн текст дээр хэрэгжихээр нүдэнд ядаргаатай өвтгөх нөлөө үүсгэж магадгүй юм. Тэгэхээр 14-н хэмжээнээс бага үсгийн маяганд anti-aliasing шинж чанарыг хэрэглэхгүй гэж бодвол дараах мөрийг оруулаарай:

```
        <match target="font">
            <test name="size" compare="less">
                <double>14</double>
            </test>
            <edit name="antialias" mode="assign">
                <bool>false</bool>
            </edit>
        </match>
        <match target="font">
            <test name="pixelsize" compare="less" qual="any">
                <double>14</double>
            </test>
            <edit mode="assign" name="antialias">
                <bool>false</bool>
            </edit>
        </match>
```

Жигд зай авалттай үсгийн маяганд anti-aliasing шинж чанар зөв хэрэгжихгүй байж магадгүй. Энэ асуудал KDE дээр их тулгардаг. Үүний засах нэг арга бол тийм үсгүүдийн зай авалтыг 100 байх ёстой гэж зааж өгдөг. Дараах мөрийг нэмж үүнийг гүйцэтгэнэ:

```
       <match target="pattern" name="family">
           <test qual="any" name="family">
               <string>fixed</string>
           </test>
           <edit name="family" mode="assign">
               <string>mono</string>
           </edit>
       </match>
       <match target="pattern" name="family">
           <test qual="any" name="family">
               <string>console</string>
           </test>
           <edit name="family" mode="assign">
               <string>mono</string>
           </edit>
       </match>
```

(энэ нь бусад ердийн ижил зай авалттай үсгийн маягуудыг `"mono"` гэсэн нэрээр хандана гэж зааж өгч байна) тэгээд дараа нь:

```
        <match target="pattern" name="family">
            <test qual="any" name="family">
                <string>mono</string>
```

```
        </test>
        <edit name="spacing" mode="assign">
            <int>100</int>
        </edit>
    </match>
```

Helvetica зэрэг зарим үсгийн маягуудад anti-aliased хэрэглэх үед бага зэрэг хүндрэлүүд гарч болзошгүй. Энэ загвар нь тийм үсгийн маягуудын талыг нь хасах шинж илэрдэг. Хамгийн муугаар бодоход програмууд ажиллах үедээ нурдаг. Үүнээс сэргийлэхийн тулд local.conf файлд дараах мөрийг нэмнэ:

```
    <match target="pattern" name="family">
        <test qual="any" name="family">
            <string>Helvetica</string>
        </test>
        <edit name="family" mode="assign">
            <string>sans-serif</string>
        </edit>
    </match>
```

local.conf файлыг засварлаж дуусаад уг файлыг `</fontconfig>` таглаагаар дууссан эсэхийг шалгаарай. Ингээгүй бол уг файлын өөрчлөлтийг үл хэрэгсэж хэрэгжүүлдэггүй.

Эцэст нь хэрэглэгчид өөрсдийн гэсэн тохируулгаа хувийн сан дотор буй .fonts.conf файл дотор хийж болдог. Ингэхийн тулд хэрэглэгч бүр ~/.fonts.conf файл үүсгэх хэрэгтэй. Энэ файл нь мөн XML хэлбэртэй байх ёстой.

Сүүлчийн үг: LCD дэлгэцтэй үед sub-pixel харуулалт хэрэглэх хүсэлт гарч болзошгүй. Энэ нь үндсэндээ (хэвтээ чиглэлд нь салгасан) улаан, ногоон болон цэнхэр өнгийн бүрдлүүдийг тусад нь салгаж харуулснаар хэвтээ чиглэлийн дүрслэх чадварыг сайжруулдаг. Үр дүн нь мэдээж маш сайн харуулалт үүсдэг. Ингэж зөвшөөрүүлэхийн тулд local.conf файлд дараах мөрийг нэмнэ:

```
    <match target="font">
        <test qual="all" name="rgba">
            <const>unknown</const>
        </test>
        <edit name="rgba" mode="assign">
            <const>rgb</const>
        </edit>
    </match>
```

Дэлгэцийн төрлөөс хамаарч `rgb` нь `bgr`, `vrgb` эсвэл `vbgr` гэж өөрчлөгдөж болзошгүй тул янз янзаар нь туршиж аль нь илүү тохирч байгааг олоорой.

# 6.6. X Display Manager буюу харуулалт зохицуулагч

## 6.6.1. Ерөнхий ойлголт

X Display Manager (XDM) буюу X харуулалт зохицуулагч гэдэг нь X цонхот системд сонгогдон хэрэглэгдэж болох бөгөөд нэвтрэх үйлдэлд хэрэглэгддэг. Энэ нь бага хүчин чадалтай "X терминал", ширээний програмын орчин, мөн том сүлжээнд буй харуулалтын сервер зэрэг төрөл бүрийн нөхцөлд хэрэглэгддэг. Нэгэнт X цонхот систем нь сүлжээ болон бүртгэлийн бие даасан гүйцэтгэлтэй болохоор X үйлчлүүлэгч болон серверүүдийн хоорондох холбоог тохируулах маш олон арга бий. XDM нь холбогдож болох серверүүдийг график горимд дүрсэлж харуулдаг бөгөөд нэр болон нууц үг оруулах нөхцөл биелүүлж өгдөг.

Та XDM-г хэрэглэгчид зориулж getty(8) хэрэгслийн биелүүлдэг гүйцэтгэлтэй (Тохиргоо хэсгээс дэлгэрэнгүйг харна уу) ижил гэж бодох хэрэгтэй. Тэгэхээр энэ нь систем рүү нэвтрэх үйлдлийг гүйцэтгэдэг бөгөөд нэвтрүүлэхдээ тухайн хэрэглэгчийн орчинг ажиллуулж (голдуу X цонхны зохицуулагч байдаг ) хэрэглэгчийг гарах хүртэл нь хүлээж байдаг. Мөн өөр хэрэглэгч холбогдох үед холбогдох дэлгэцийг харуулан нэвтрэх үйлдэл хийлгэх боломжийг XDM биелүүлж байдаг.

## 6.6.2. XDM-г хэрэглэх нь

XDM-ийг ашиглаж эхлэхийн тулд x11/xdm портыг суулгана (энэ нь анхдагчаар Xorg-ийн сүүлийн үеийн хувилбаруудтай цуг суудаггүй). Дараа нь XDM далд чөтгөрийг /usr/local/bin/xdm гэсэн байрлалаас олж болно. Энэ нь `root` эрхтэй хүнээр ямар ч үед эхлүүлэгдэж болдог бөгөөд эхлэнгүүтээ уг машинд буй X цонхыг зохицуулах үүргийг гүйцэтгэж эхэлдэг. Хэрэв XDM-г машиныг шинээр ачаалагдах болгонд эхлүүлэх хүсэлтэй байгаа бол үүнийг биелүүлэх хамгийн эвтэйхэн зам бол /etc/ttys файлд оруулга хийж өгөх билээ. Энэ файлын зохион байгуулалтын талаар /etc/ttys уруу оруулгууд нэмэх нь хэсэгт хараарай. /etc/ttys файл дотор XDM-г виртуал терминал дээр далд чөтгөр хэлбэрээр ажиллуулна гэсэн дараах мөр байдаг:

```
ttyv8   "/usr/local/bin/xdm -nodaemon"  xterm   off secure
```

Анхдагч тохируулгаараа энэ нь хоригдсон байдаг бөгөөд идэвхжүүлэхийг хүсвэл тав дахь үг болох `off` гэснийг `on` болгож өөрчлөөд init(8)-г /etc/ttys-г init тушаалаар дахин уншуулах дээр заасны дагуу шинээр ачаалах хэрэгтэй. Эхний талбар нь уг програмын зохицуулах терминалын нэр бөгөөд `ttyv8` гэсэн буй. Энэ нь XDM есдүгээр виртуал терминал дээр ажиллаж эхэлнэ гэсэн үг юм.

## 6.6.3. XDM-г тохируулах нь

XDM-г тохируулах файл /usr/local/lib/X11/xdm санд байдаг. Энэ сан дотор XDM-н харагдцыг өөрчилж тохируулах хэд хэдэн файл бий. Голдуу дараах төрлийн файлууд байдаг:

| Файл | Тодорхойлолт |
| --- | --- |
| Xaccess | Хэрэглэгчийн эрхийн дүрэм. |

| Файл | Тодорхойлолт |
| --- | --- |
| Xresources | X-н resource буюу эх сурвалж анхдагч утгууд. |
| Xservers | Дотоод юм уу алсад буй холбогдож болох дэлгэц зохицуулагчдын жагсаалт. |
| Xsession | Нэвтрэх үед тухайн орчинд үйлдэгддэг анхдагч скрипт. |
| Xsetup_* | Нэвтрэх үйлдлээс өмнө нь програмуудыг ачаалагч скрипт. |
| xdm-config | Уг машин дээр ажиллаж байгаа бүх харуулагчдыг ерөнхийд нь тохируулагч файл. |
| xdm-errors | Сервер програмаас үүсгэгдсэн алдаанууд. |
| xdm-pid | Одоогийн ажиллаж байгаа XDM-н процесс ID дугаар. |

Мөн энэ сан дотор XDM-г ажиллаж байх үед уг график орчныг тохируулдаг скрипт болон програмууд байдаг. Саяны жагсаасан файлуудын зориулалтыг ерөнхийд нь дор дурдав. Уг файлуудыг хэрэглэх дэлгэрэнгүй заавар нь xdm(1) хуудсанд тодорхойлогдсон байгаа.

Анхныхаа тохируулгаар бол энгийн нэгэн нэвтрэх дөрвөлжин цонх дотор уг машины нэр харуулагдсан байдаг ба том үсгээр "Login:" гэж бичээд доод хэсэгт нь "Password:" гэж нууц үг оруулах талбар байдаг. Энэ цонхноос эхлэн XDM-н харагдцыг өөрчилж эхэлж болох юм.

#### 6.6.3.1. Xaccess

XDM-ээр хянагддаг харуулагчтай холбогдох протоколыг X Display Manager Connection Protocol (XDMCP) гэж нэрлэдэг. Энэ файл нь алсад буй компьютераас XDMCP холболт ямар дүрмээр хийгдэхийг заан хянаж байдаг. Энэ нь алсаас холбогдох холболтыг хүлээж авах тохиргоог xdm-config файлд хийх хүртэл хориотой байдаг. Анхдагч тохиргоогоороо бол ямар ч хэрэглэгчийн холболтыг зөвшөөрөхгүй гэж заагдсан байдаг.

#### 6.6.3.2. Xresources

Энэ нь харуулагчдыг сонгох нэвтрэх дэлгэцийн харуулалтыг өөрчилж болох анхдагч утгууд хадгалсан файл юм. Энэ файлаар нэвтрэх програмыг өөрчилж болно. Файлын зохион байгуулалт нь X11-н бичиг баримтад заасантай ижил загвартай байдаг.

#### 6.6.3.3. Xservers

Энэ нь сонгогдож болох харуулагчдын жагсаалт хадгалсан файл.

#### 6.6.3.4. Xsession

Энэ файл нь XDM-д зориулсан хэрэглээг холбогдсоны дараа гүйцэтгэгдэх скрипт файл юм. Ихэнхдээ хэрэглэгчид өөрсдийнхөө эхлэл санд буй ~/.xsession файл дотор өөрсдийн гэсэн ажиллуулах бичлэгээ бичиж энэ файлын гүйцэтгэлийг дардаг.

#### 6.6.3.5. Xsetup_*

Энэ файл нь нэвтрэх цонх болон харуулагчдыг сонгогдохоос өмнө автоматаар ажилладаг. Харуулагч болгонд зориулсан скрипт нь Xsetup_ нэрэн дээр харуулагчийн дугаар залгагдсан нэртэй байдаг ( жишээ нь дотоод харуулагч Xsetup_0 гэсэн нэртэй байдаг). Энэ файлд голдуу `xconsole` зэргийн нэг юм уу хоёр програмыг ар талд ажиллуулж байхаар бичсэн байдаг.

#### 6.6.3.6. xdm-config

Энэ файлд харуулагч болгонд хэрэглэгдэх програмын анхдагч утга хэлбэрийн тохируулгууд байдаг.

#### 6.6.3.7. xdm-errors

Энэ файл дотор XDM-н ажиллуулахыг оролдсон серверээс гарсан алдаануудыг бичсэн байдаг. Хэрэв XDM-н эхлүүлсэн дэлгэц ямар нэгэн замаар гацах юм бол юунаас болсон эсэхийг нь мэдэх хамгийн зөв газар бол энэ билээ. Эдгээр алдаанууд мөн хэрэглэгчийн тухайн орчноос хамаарч ~/.xsession-errors файл дотор бас бичигддэг.

### 6.6.4. Сүлжээний харуулагч серверийг ажиллуулах

Бусад хэрэглэгчдийг харуулагч сервер рүү холбохын тулд та хандах эрхийн дүрмийг засварлаж холболт хүлээж авагчийг зөвшөөрүүлэх хэрэгтэй. Анхдагч утгаараа үүнийг хорьсон байдаг. XDM-г холболт хүлээж авдаг болгохын тулд эхлээд xdm-config файл доторх мөрийг тайлбар мөр болгож хүчингүй болгох хэрэгтэй:

```
! SECURITY: do not listen for XDMCP or Chooser requests
! Comment out this line if you want to manage X terminals with xdm
DisplayManager.requestPort:     0
```

тэгээд XDM-ийг шинээр эхлүүлэх хэрэгтэй. Ийм файлд "#" тэмдгээр тайлбар мэт болгож хасдаггүй харин "!" тэмдэг хэрэглэх хэрэгтэй гэдгийг санах хэрэгтэй. Xaccess файл доторхоос жишээнүүдийг харах хэрэгтэй бөгөөд xdm(1) гарын авлагаас дэлгэрэнгүй унших нь зүйтэй.

### 6.6.5. XDM-н орлуулгууд

XDM-г орлох хэд хэдэн програм бий. Тэдний нэг болох KDM-г ( KDE-тэй цуг ирдэг) энэ бүлэгт тайлбарлах болно. Харуулалт зохицуулагч KDM нь маш олон төрлийн харуулах загварыг санал болгодог бөгөөд нэвтрэх үед цонх зохицуулагчдыг давхар сонгож болох боломж өгдөг.

## 6.7. Дэлгэцийн орчин

Энэ хэсэгт FreeBSD-н X-д зориулсан төрөл бүрийн дэлгэцийн орчнуудыг тайлбарлах болно. "Дэлгэцийн орчин" гэдэгт энгийн цонх зохицуулагчаас авахуулаад төрөл бүрийн ширээний програмыг агуулсан цогц програмууд болох KDE болон GNOME зэрэг ордог.

### 6.7.1. GNOME

#### 6.7.1.1. GNOME-н тухай

GNOME гэдэг нь компьютераа тохируулахад тань амарчилж хөнгөвчилсөн дэлгэцийн програмын орчин юм. GNOME дотор самбар (програм эхлүүлэх болон төлөв байдлыг нь харуулах зорилготой), дэлгэцийн орон зай (програмууд байрлахад зориулагдсан ), дэлгэцийн хэрэгслүүд болон програмуудын цуглуулга, мөн програмууд өөр хоорондоо зохицож ажиллахад зориулагдсан хэд хэдэн журам агуулагдаж байдаг. Өөр үйлдлийн систем юм уу эсвэл өөр орчинд ажиллаж сурсан хүмүүст GNOME-н санал болгож байгаа хүчтэй бөгөөд аятайхан график орчин нь дасахад амар санагддаг. FreeBSD дээрх GNOME-н талаар дэлгэрэнгүй мэдээллийг FreeBSD-н GNOME Төслийн вэб хуудаснаас харах хэрэгтэй. Уг вэб хуудсанд GNOME-г хэрхэн суулгаж тохируулан мөн зохицуулах талаар нэлээн өргөн хэмжээний асуулт хариултууд байдаг.

#### 6.7.1.2. GNOME-г суулгах

Энэ програм хангамжийг портын цуглуулга юм уу эсвэл багцаас хялбар аргаар суулгаж болно:

Сүлжээнээс GNOME-г багц хэлбэрээр нь суулгахыг хүсвэл дараах тушаалыг өгөхөд хангалттай:

```
# pkg_add -r gnome2
```

GNOME-г порт дотор эх бичлэгээс нь хөрвүүлж суулгахыг хүсвэл дараах тушаалаар суулгана:

```
# cd /usr/ports/x11/gnome2
# make install clean
```

Зөв ажиллагааг хангахын тулд GNOME нь /proc файлын системийг холбосон байхыг шаарддаг. Эхлүүлэх явцад procfs(5)-г автоматаар холбохын тулд доорхи

```
proc           /proc       procfs  rw  0   0
```

мөрийг /etc/fstab файлд нэмж өгөх хэрэгтэй.

GNOME суугдсаны дараа X серверт анхдагч цонх зохицуулагчийн оронд GNOME эхлэхийг зааж өгөх хэрэг гардаг.

GNOME-г эхлүүлэх амархан арга бол GNOME-н харуулалтыг зохицуулагч буюу GDM-г хэрэглэх билээ. GDM нь GNOME-той хамт суугддаг боловч анхдагч тохиргоогоороо бол идэвхжээгүй байдаг. Дараах мөрийг /etc/rc.conf файлд нэмж идэвхжүүлж болно:

```
gdm_enable="YES"
```

Дахин ачаалсны дараа GNOME автоматаар эхэлдэг.

GNOME-ийн бүх үйлчилгээг ихэвчлэн GDM-тэй цуг эхлүүлэх шаардлага байдаг. Ингэхийн тулд дараах мөрийг /etc/rc.conf файлд нэмнэ:

```
gnome_enable="YES"
```

GNOME-г мөн тушаал бичдэг мөрнөөс .xinitrc файлд зөв тохируулга хийснээр ажиллуулж бас болно. Хэрэв .xinitrc файл өмнө нь байж байвал уг файлд байгаа цонх зохицуулагч эхлэх мөрийг арилгаад оронд нь /usr/local/bin/gnome-session гэж оруулах хэрэгтэй. Хэрэв нэг их онц чухал мөр уг тохиргооны файл дотор байхгүй гэж та мэдэж байгаа бол дараах тушаалыг өгч бас болно:

```
% echo "/usr/local/bin/gnome-session" > ~/.xinitrc
```

Тэгээд `startx` гэсэн тушаалыг бичихэд GNOME дэлгэцийн орчин эхлэх болно.

Хэрэв XDM юм уу ямар нэгэн өөр дэлгэцийн орчин хэрэглэгдэж байсан бол .xsession гэсэн файл үүсгээд уг файл дотор өмнө ярьж байсан оруулгыг оруулах хэрэгтэй. Ингэхийн тулд уг файлыг засварлагчаар нээгээд дотор байгаа мөрийг /usr/local/bin/gnome-session мөрөөр дарж бичихэд болно:

```
% echo "#!/bin/sh" > ~/.xsession
% echo "/usr/local/bin/gnome-session" >> ~/.xsession
% chmod +x ~/.xsession
```

Өөр нэг сонголт бол цонх зохицуулагчид өөрчлөлт хийж нэвтрэлт хийгдэх үед дэлгэцийн орчныг сонгож болдгоор тохируулж болдог. KDE-н дэлгэрэнгүй хэсэгт KDE-н дэлгэцийн орчны зохицуулагч KDM дээр хэрхэн үүнийг хийдэг талаар тайлбарласан байгаа.

## 6.7.2. KDE

### 6.7.2.1. KDE-н тухай

KDE бол хэрэглэхэд тун амар орчин үеийн дэлгэцийн орчин юм. Хэрэглэгчид хэрэгтэй KDE-н санал болгодог зарим зүйлс бол:

- Орчин үеийн сайхан дэлгэцийн орчин
- Сүлжээгээр ажиллуулахад ямар ч хүндрэлгүй
- KDE дэлгэцийн орчин болон түүний програмуудад зориулж өөртөө агуулсан тусламжийн системтэй
- KDE-н бүх програмууддаа тохирсон загвар маягтай
- Стандартчлагдсан цэс болон багажит самбар, гарын товчлолууд, өнгөний хүснэгт гэх мэтчилэн.

- Internationalization буюу олон хэлийн дэмжлэг: KDE нь 55 гаран гадаад хэл дээр боломжтой
- Бүх дэлгэцийн орчны тохируулгыг төвлөрүүлсэн цонхот загвартай тохиргоо хийх боломж
- Маш олон KDE програмууд

KDE нь Konqueror нэртэй UNIX® ертөнцөд өрсөлдөөн ихтэй хөтлөгчүүдийн нэг болох вэб хөтлөгчтэй хамт ирдэг. KDE-н талаар дэлгэрэнгүй мэдээллийг KDE вэб хуудаснаас харж болно. KDE-тэй холбоотой эх үүсвэрүүд болон FreeBSD-тэй холбоотой тусгай мэдээллийг KDE/FreeBSD-ийн санаачлага вэб хуудаснаас лавлах хэрэгтэй.

FreeBSD дээр KDE-н хоёр хувилбар байдаг. Хувилбар 3 нь удаан хугацааны туршид байгаа бөгөөд портын цуглуулгад байдаг боловч цааш хөгжүүлэлт хийгдэхгүй бөгөөд ажиллахгүй байж болох асуудалтай. Хувилбар 4 нь шинэчлэгдсэн бөгөөд KDE хэрэглэгчийн хувьд анхдагч сонголт юм. Эдгээр нь бүр цуг зэрэгцэн суулгагдаж болдог.

#### 6.7.2.2. KDE-г суулгах нь

GNOME болон бусад дэлгэцийн орчны програмуудыг суулгадаг шиг энэхүү програм хангамжийг портын цуглуулга юм уу эсвэл багцаас хялбар аргаар суулгаж болно:

Сүлжээгээр KDE 3-г багц хэлбэрээр суулгана гэвэл дараах тушаалыг өгөхөд л болно:

```
# pkg_add -r kde
```

Сүлжээгээр KDE 4-г багц хэлбэрээр суулгана гэвэл дараах тушаалыг өгөхөд л болно:

```
# pkg_add -r kde4
```

pkg_add(1) автоматаар уг програмын сүүлийн хувилбарыг нь нөхөж суулгах болно.

KDE 3-г эх бичлэгээс нь хөрвүүлж суулгана гэвэл портын цуглуулгыг ашиглаарай:

```
# cd /usr/ports/x11/kde3
# make install clean
```

KDE 4-г эх бичлэгээс нь хөрвүүлж суулгана гэвэл портын цуглуулгыг ашиглаарай:

```
# cd /usr/ports/x11/kde4
# make install clean
```

KDE суугдсаны дараа X серверт анхдагч цонх зохицуулагчийн оронд KDE-г ажиллуулна гэж зааж өгөх хэрэгтэй. Үүний тулд .xinitrc файлыг дараах аргаар засварлаж мөн болно:

KDE 3-н хувьд:

```
% echo "exec startkde" > ~/.xinitrc
```

KDE 4-н хувьд:

```
% echo "exec /usr/local/kde4/bin/startkde" > ~/.xinitrc
```

Одоо ингээд `startx` тушаалыг ажиллуулбал KDE дэлгэцийн орчин эхлэх болно.

Хэрэв урьд нь XDM зэрэг өөр дэлгэцийн орчин хэрэглэгдэж байсан бол тохируулга арай өөр байх болно. Тэр үед .xsession файлыг засварлах хэрэгтэй. Энэ бүлгийн сүүл хэсэгт KDM-н талаар зааварласан байгаа.

### 6.7.3. KDE-н талаар дэлгэрэнгүй

Одоо KDE суугдсан байгаа болохоор ихэнх зүйлсийг та тусламжийн системийг нь ашиглан нээж олох юм уу эсвэл зүгээр л цэснүүд дээр дарж туршиж болох юм. Windows® эсвэл Mac® төрлийн хэрэглэгчдэд бол бараг гэртээ байгаа юм шиг л сэтгэгдэл төрөх байх.

KDE-н хамгийн сайн заавар бичиг баримтууд интернэт дээр бий. KDE өөртэйгөө хамт Konqueror хөтлөгч мөн маш олон програм болон өргөжүүлсэн бичиг баримтуудтай ирдэг. Бүлгийн үлдсэн хэсэгт санамсаргүй нээж олсон ч гэсэн сурахад хүндрэлтэй техникийн зарим асуудлуудыг зааварчлах болно.

#### 6.7.3.1. KDE Display Manager буюу KDE-н дэлгэцийн зохицуулагч

Олон хэрэглэгчтэй системийн администратор хэрэглэгчдэд график нэвтрэх горим хэрэглэхийг хүсдэг. Өмнө тайлбарласны дагуу XDM ийм зорилгоор бас хэрэглэгдэж болно. Гэхдээ KDE бас үүнтэй ижил үүрэг гүйцэтгэж чадах KDM нэртэй харагдац сайтай эвтэйхэн нэвтрэх хэсгийг санал болгодог. Мөн хэрэглэгч бүр уг зохицуулагчийн цэснээс ямар дэлгэцийн орчинд холбогдохоо (KDE, GNOME, эсвэл бусад өөр) нэвтрэх үедээ сонгож болдог.

KDM-г идэвхжүүлэхийн тулд KDE-ийн хувилбараас хамааран өөр өөр файлыг засварлах хэрэгтэй болдог.

KDE 3-н хувьд /etc/ttys файлд буй `ttyv8` гэсэн оруулгатай хэсгийг дараах маягаар өөрчлөх ёстой:

```
ttyv8 "/usr/local/bin/kdm -nodaemon" xterm on secure
```

KDE 4-н хувьд procfs(5)-г холбож /etc/rc.conf файлд дараах мөрийг нэмэх ёстой:

```
kdm4_enable="YES"
```

## 6.7.4. Xfce

### 6.7.4.1. Xfce-н тухай

Xfce нь GNOME дээр хэрэглэгддэг GTK+ хэрэгсэл дээр үндэслэгдсэн дэлгэцийн орчин бөгөөд хэрэглэхэд маш хөнгөн амархан тохируулгатай билээ. Өнгөц харахад UNIX® системүүд дээр байдаг үнэтэй зарагддаг CDE дэлгэцийн орчинтой төсөөтэй. Xfce-н зарим шинж чанарыг дурдвал:

- Дэлгэцийн орчинд ажиллахад тун амар хялбаршуулсан
- Хулганы чирж тавих үйлдэл зэрэгт бүрэн тохируулагдсан
- Цэс болон програм ачаалах үндсэн самбар нь CDE-тэй ижил
- Цонх зохицуулагч, файл зохицуулагч, дууны оролт гаралтыг зохицуулагч, GNOME-н дэмжлэгтэй програмуудыг ажиллуулах тохиромж зэрэг олон зүйлсийг багтаасан
- Орчноо өөрчилж болдог (GTK+ дэмжлэгтэй учраас)
- Түргэн, хөнгөн үр бүтээлтэй болохоор санах ойн хязгаарлалтай удаан машинуудад тун тохиромжтой.

Xfce-н дэлгэрэнгүй мэдээллийг Xfce вэб хуудаснаас хараарай.

### 6.7.4.2. Xfce-г суулгах нь

Xfce-н багц хувилбар (үүнийг бичиж байх үед) байдаг. Суулгахын тулд ердөө л:

```
# pkg_add -r xfce4
```

Мөн портын цуглуулга дотроос эх бичлэгээс нь хөрвүүлж суулгаж болно:

```
# cd /usr/ports/x11-wm/xfce4
# make install clean
```

Одоо X серверт X орчныг эхлүүлэх үед Xfce-г ажиллуул гэж зааж өгөхийн тулд:

```
% echo "/usr/local/bin/startxfce4" > ~/.xinitrc
```

Дараагийн удаа X эхлэх үед Xfce нь дэлгэцийн орчин болсон байна. Мөн өмнөх орчнуудтай ижил, хэрэв XDM зэрэг өөр орчин урьд нь хэрэглэгдэж байсан бол .xsession файлыг үүсгээд GNOME хэсэгт тайлбарласан шиг гэхдээ /usr/local/bin/startxfce4 гэсэн утгыг оруулах юм уу эсвэл нэвтрэх үед сонгогдохоор болгож тохируулахыг хүсвэл kdm хэсэгт тайлбарласны дагуу хийх хэрэгтэй.

# Part II: Нийтлэг асуудлууд

Үндсэн ойлголтуудыг нэгэнт тайлбарласан болохоор одоо FreeBSD-ийн гарын авлагын энэ хэсэгт FreeBSD-ийн зарим байнга ашиглагддаг боломжуудыг авч үзэх болно. Эдгээр бүлгүүд нь:

- Түгээмэл хэрэглэгддэг ширээний програмуудыг танд танилцуулах болно: хөтчүүд, бүтээмжийн багажууд, баримт үзэгчид гэх мэт.
- FreeBSD-н хэд хэдэн мультимедиа багажуудыг танд танилцуулах болно.
- Тусгайлан тохируулсан FreeBSD цөмийг бүтээх үйл явц болон, системд нэмэлт функцуудыг хэрхэн идэвхжүүлэхийг тайлбарлах болно.
- Ширээний болон сүлжээнд холбогдсон хэвлэгчийн тохируулгуудын хувьд хэвлэх системийг дэлгэрэнгүй тайлбарлах болно.
- FreeBSD систем дээр Линукс програмуудыг хэрхэн ажиллуулахыг танд үзүүлэх болно.

Эдгээр бүлгүүдийн заримд таныг зарим нэг ойлголтуудын талаар урьдаар уншихыг зөвлөх бөгөөд энэ нь бүлэг бүрийн эхэнд ерөнхий агуулгад дурдсан байгаа болно.

# Chapter 7. Ширээний програмууд

## 7.1. Ерөнхий агуулга

FreeBSD нь текст боловсруулагч програмууд болон хөтчүүд зэрэг ширээний төрөл бүрийн хэрэглээний програмуудыг ажиллуулж чаддаг. Эдгээр програмуудын ихэнх нь багц хэлбэрээр юм уу эсвэл Портын Цуглуулгаас автоматаар бүтээгдэж болно. Шинэ хэрэглэгчдийн олонхи нь эдгээр хэрэглээний програмуудыг өөрийн компьютер дээр суулгахыг хүсдэг. Энэ бүлэгт зарим нэг түгээмэл хэрэглэгддэг ширээний програмуудыг Портын цуглуулга болон тэдгээрийн багцуудаас хүндрэлгүйгээр суулгах талаар зааварлах болно.

Портуудаас програмуудыг суулгахад тэдгээрийг эхээс эмхэтгэдэг гэдгийг анхаарна уу. Энэ нь таны машины (машинуудын) процесс хийх хүч болон юу эмхэтгэж байгаагаас хамаарч их удаан ажиллаж болох юм. Эхээс бүтээх нь саад болмоор их хугацаа шаардаж байвал та Портын цуглуулгад байгаа ихэнх програмуудыг урьдчилан бүтээсэн багцуудаас суулгаж болно.

FreeBSD нь Линуксийн хоёртын хэлбэртэй нийцтэй ажиллах боломжийг хангадаг учраас Линуксд эхлэн хөгжүүлэгдсэн олон програмуудыг ашиглах боломжтой байдаг. Линуксийн ямар ч хэрэглээний програмыг суулгахаасаа өмнө Линуксийн Хоёртын Нийлэмж-ийг уншихыг зөвлөж байна. Линуксийн хоёртын хэлбэртэй нийцтэй байдлыг ашиглаж байгаа ихэнх портуудын нэр "linux-" гэж эхэлсэн байгаа. Жишээ нь whereis(1) тушаалын тусламжтайгаар ямар нэг портыг хайж байхдаа үүнийг санаарай. Нийтлэлийн энэ хэсгээс цааш ямар нэг Линуксийн хэрэглээний програмуудыг суулгахын өмнө таныг Линуксийн хоёртын хэлбэртэй нийцтэй байдлыг идэвхжүүлсэн байгаа гэж тооцох болно.

Энэ бүлэгт доорх хэсгүүдийн тухай ярих болно:

- Хөтчүүд (Firefox, Opera, Konqueror, Chromium зэрэг)
- Бүтээмж ( KOffice, AbiWord, The GIMP, OpenOffice.org, LibreOffice зэрэг)
- Бичиг баримт үзүүлэгчид (Acrobat Reader®, gv, Xpdf, GQview зэрэг)
- Санхүү ( GnuCash, Gnumeric, Abacus зэрэг)

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- Гуравдагч талуудын нэмэлт програмуудыг хэрхэн суулгах талаар мэдэх (Програм суулгах. Багцууд болон портууд).
- Линуксийн нэмэлт програм хангамжуудыг хэрхэн суулгах талаар мэдэх (Линуксийн Хоёртын Нийлэмж).

Мультимедиа орчны тохиргооны тухай мэдээллийг Мультимедиа-с уншина уу. Хэрэв та цахим захидал тохируулж ашиглахыг хүсвэл Цахим шуудан-т хандана уу.

## 7.2. Хөтчүүд

FreeBSD нь тодорхой хөтөч урьдчилан суулгагдсан байдаггүй. Харин Портын цуглуулгын www сан суулгахад бэлэн олон хөтчүүдийг агуулдаг. Танд бүгдийг эмхэтгэх хугацаа байхгүй бол (энэ нь зарим тохиолдолд маш их хугацаа шаардаж болно) тэдгээрийн олонхи нь багц хэлбэрээр байдаг.

KDE болон GNOME нь HTML хөтчүүдээр хангадаг. Эдгээр бүрэн гүйцэд ширээний програмуудыг хэрхэн суулгаж тохируулах талаар дэлгэрэнгүй мэдээллийг Дэлгэцийн орчин -с лавлана уу.

Хэрэв та хөнгөн чанарын хөтчүүдийг хайж байгаа бол Портын цуглуулгаас www/dillo2, www/links, эсвэл www/w3m зэргүүдийг судалж үзэх хэрэгтэй.

Энэ хэсэг эдгээр програмуудыг авч үзэх болно:

| **Програмын нэр** | **Шаардлагатай эх үүсвэрүүд** | **Портоос суулгах** | **Гол хамаарлууд** |
|---|---|---|---|
| Firefox | дунд зэрэг | хүнд | Gtk+ |
| Opera | бага | хөнгөн | FreeBSD болон Линуксийн хувилбарууд байдаг. Линуксийн хувилбар нь Линуксийн хоёртын нийцтэй байдал болон linux-openmotif-с хамаардаг. |
| Konqueror | дунд зэрэг | хүнд | KDE сангууд |
| Chromium | дунд зэрэг | дунд зэрэг | Gtk+ |

### 7.2.1. Firefox

Firefox нь FreeBSD рүү бүрэн порт хийгдсэн орчин үеийн, чөлөөтэй, нээлттэй эхийн тогтвортой хөтөч юм. Энэ нь стандартыг хангасан HTML үзүүлэх хэсэг, tab нээж ажиллах боломж, popup хаалт, өргөтгөлүүд, сайжруулсан аюулгүй байдал зэрэг олон боломжуудтай. Firefox нь Mozilla-ийн код дээр суурилжээ.

Firefox нь Mozilla-ийн код дээр суурилсан дараагийн үеийн хөтөч юм. Mozilla нь хөтөч, захидлын клиент, чалчих клиент зэрэг олон програмуудыг багтаасан бүрэн гүйцэд програмуудын цуглуулга юм. Firefox нь зөвхөн хөтөч бөгөөд энэ нь түүнийг илүү жижиг, хурдан болгодог байна.

Доор дурдсаныг бичин багцыг суулгана:

```
# pkg_add -r firefox
```

Энэ нь Firefox-н хамгийн сүүлийн хувилбарыг суулгах бөгөөд та хэрэв Firefox-н Extended Support Release (ESR) хувилбарыг ашиглахыг хүсэж байвал доорхийг ажиллуулаарай:

```
# pkg_add -r firefox-esr
```

Эх кодоос эмхэтгэхийг та илүүд үзэж байгаа бол Портын цуглуулгыг бас ашиглаж болох юм:

```
# cd /usr/ports/www/firefox
# make install clean
```

Firefox ESR-ийн хувьд өмнөх тушаал дээрх `firefox`-ийг `firefox-esr`-аар солиорой.

### 7.2.2. Firefox болон Java™ залгаас (plugin)

Энэ хэсэгт болон дараагийн хоёр хэсэгт бид таныг Firefox-г аль хэдийн суулгасан гэж тооцох болно.

OpenJDK 6-г Портын цуглуулгаас дараах тушаалаар суулгана:

```
# cd /usr/ports/java/openjdk6
# make install clean
```

Дараа нь java/icedtea-web портыг суулгана:

```
# cd /usr/ports/java/icedtea-web
# make install clean
```

Аль алиных нь хувьд анхдагч тохиргоог хадгалахаа мартуузай.

Хөтчөө эхлүүлээд байршил бичих хэсэгт `about:plugins` гэж бичээд `Enter` дарна. Суулгагдсан залгаасуудын жагсаалт бүхий хуудас дэлгэцэд гарна. Java™ залгаас тэр жагсаалтад одоо байх ёстой.

Хэрэв хөтөч залгаасыг олохгүй байгаа бол хэрэглэгч бүр дараах тушаалыг ажиллуулж хөтчийг дахин эхлүүлэх шаардлагатай:

```
% ln -s /usr/local/lib/IcedTeaPlugin.so \
$HOME/.mozilla/plugins/
```

### 7.2.3. Firefox болон Adobe® Flash™ залгаас (plugin)

Adobe® Flash™ залгаас нь FreeBSD-д байдаггүй. Гэхдээ залгаасын Линукс хувилбарыг ажиллуулах програм хангамжийн давхарга (гүйцэтгэл хялбаршуулагч) байдаг. Энэ хялбаршуулагч нь бас Adobe® Acrobat® залгаас, RealPlayer® залгаас болон өөр олныг

дэмждэг.

Таны ашиглаж байгаа FreeBSD-ийн хувилбараас хамааранд төрөл бүрийн алхмууд шаардлагатай:

1. FreeBSD 7.X-ийн хувьд

   www/nspluginwrapper порт суулгана. Уг порт нь emulators/linux_base-fc4 портыг шаардах бөгөөд энэ нь том порт юм.

   Дараагийн алхам нь www/linux-flashplugin9 портыг суулгах явдал юм. Энэ нь Flash™ 9.X-ийг суулгана. Энэ хувилбар нь FreeBSD 7.X дээр зөв ажилладаг.

2. FreeBSD 8.X болон түүнээс хойшхийн хувьд

   www/nspluginwrapper порт суулгана. Уг порт нь emulators/linux_base-f10 портыг шаардах бөгөөд энэ нь том порт юм.

   Дараагийн алхам нь Flash™ 11.X-г www/linux-f10-flashplugin11 портоос суулгах явдал юм.

   Энэ хувилбар нь дараах холбоосыг үүсгэхийг шаарддаг:

   ```
   # ln -s /usr/local/lib/npapi/linux-f10-flashplugin/libflashplayer.so \
     /usr/local/lib/browser_plugins/
   ```

   Хэрэв /usr/local/lib/browser_plugins сан систем дээр байхгүй бол гараар үүсгэж өгөх хэрэгтэй.

Таны ажиллуулж байгаа FreeBSD-ийн хувилбарт таарсан зөв Flash™ порт суулгагдсаны дараа залгаасыг хэрэглэгч бүрийн хувьд `nspluginwrapper` тушаал ашиглан суулгах шаардлагатай:

```
% nspluginwrapper -v -a -i
```

Тэгээд өөрийн хөтчийг эхлүүлж байрлал оруулдаг мөрөнд `about:plugins` гэж бичин `Enter` дарна. Байгаа бүх залгаасуудтай жагсаалт гарч ирэх ёстой.

### 7.2.4. Firefox болон Swfdec Flash™ залгаас

Swfdec нь Flash™ анимацыг задалж амилуулах сан юм. Swfdec-Mozilla нь SWF файлуудыг тоглуулахын тулд Swfdec санг ашигладаг бөгөөд Firefox хөтчид зориулагдсан залгаас юм. Энэ нь хүчтэй хөгжүүлэлтийн доор оршсон хэвээр байгаа болно.

Хэрэв та бүтээж чадахгүй эсвэл хүсэхгүй байгаа бол сүлжээнээс багцыг суулгаж болно:

```
# pkg_add -r swfdec-plugin
```

Хэрэв багц байхгүй бол та үүнийг Портын цуглуулгаас бүтээн суулгаж болно:

```
# cd /usr/ports/www/swfdec-plugin
# make install clean
```

Дараа нь энэ залгаасыг идэвхжүүлэхийн тулд өөрийн хөтчийг дахин ажиллуулаарай.

### 7.2.5. Opera

Opera нь өргөн боломжуудтай стандартад нийцсэн хөтөч юм. Энэ нь өөртөө бүтээгдсэн захидал болон мэдээ уншигч, IRC клиент, RSS/Atom уншигч болон өөр олон боломжуудыг агуулдаг. Эдгээрээс гадна Opera нь харьцангуй хөнгөн бөгөөд маш хурдан юм. Энэ нь хоёр хэлбэрээр байдаг нь: "эх" FreeBSD хувилбар болон Линуксийн эмуляц дээр ажилладаг хувилбар юм.

Opera-ийн FreeBSD хувилбараар вэб үзэхийн тулд багцыг суулгана:

```
# pkg_add -r opera
```

Зарим нэг FTP сайтуудад бүх багцууд байдаггүй боловч Портын цуглуулгыг ашиглан доор дурдсаныг бичин Opera-г авч болно:

```
# cd /usr/ports/www/opera
# make install clean
```

Opera-ийн Линукс хувилбарыг суулгахын тулд дээрх жишээний `opera` гэсний оронд `linux-opera` гэж солих хэрэгтэй.

Adobe® Flash™ залгаас нь FreeBSD дээр байдаггүй. Гэхдээ залгаасын Linux® хувилбар байдаг. Энэ хувилбарыг суулгахын тулд www/linux-f10-flashplugin11 порт суусан байх шаардлагатай бөгөөд дараа нь www/opera-linuxplugins порт суулгана:

```
# cd /usr/ports/www/linux-f10-flashplugin11
# make install clean
# cd /usr/ports/www/opera-linuxplugins
# make install clean
```

Залгаас суусан байгаа эсэхийг шалгахын тулд хөтчөө эхлүүлээд `opera:plugins` гэж байрлал бичдэг мөрөнд бичээд `Enter` дарна. Суусан байгаа залгаасуудын жагсаалт гарч ирэх ёстой.

Java™ залгаасыг суулгахын тулд Firefox-д зориулсан зааврыг дагаарай.

### 7.2.6. Konqueror

Konqueror нь KDE-ийн нэг хэсэг боловч үүнийг KDE-ийн гадна талд x11/kdebase3-ийг суулган бас ашиглаж болох юм. Konqueror ердийн нэг хөтчөөс илүү бөгөөд файлын менежер болон мультимедиа үзүүлэгч болж чаддаг.

Konqueror нь бас залгаасуудын олонлогтой ирдэг бөгөөд эдгээрийг misc/konq-plugins сангаас суулгах боломжтой байдаг.

Konqueror нь WebKit болон өөрийн KHTML-г дэмждэг. WebKit нь Chromium зэрэг орчин үеийн олон хөтчүүдэд ашиглагддаг. WebKit-г FreeBSD дээр Konqueror-тай ашиглахын тулд:

```
# cd /usr/ports/www/kwebkitpart
# make install clean
```

Дараа нь Konqueror дээрээ "Settings", "Configure Konqueror" дарж "Change KHTML to WebKit" гэдгийг сонгоно.

Konqueror бас Flash™-г дэмждэг бөгөөд Konqueror дээр Flash™-ийн дэмжлэгтэй болох талаар "Хэрхэн Хийх" заавар нь http://freebsd.kde.org/howtos/konqueror-flash.php хаягт байдаг.

### 7.2.7. Chromium

Chromium нь нээлттэй эхийн аюулгүй, хурдан илүү тогтвортой вэб үзэх боломж олгодог вэб хөтчийн төсөл юм. Chromium нь таб хөтчийн боломж, цонх блоклох, өргөтгөлүүд болон илүү олон боломжийг агуулдаг. Chromium нь нээлттэй эхийн төсөл бөгөөд Google Chrome вэб хөтөч түүн дээр тулгуурласан байдаг.

Chromium-г багц хэлбэрээр ингэж суулгаж болно:

```
# pkg_add -r chromium
```

Мөн Chromium-г портын цуглуулгаас эх кодыг ашиглан бүтээж болно:

```
# cd /usr/ports/www/chromium
# make install clean
```

Chromium нь /usr/local/bin/chromium биш харин /usr/local/bin/chrome гэж суудаг.

### 7.2.8. Chromium ба Java™ залгаас

Энэ хэсэг Chromium аль хэдийн суусан гэж үздэг.

OpenJDK 6-г портын цуглуулгаас суулгана:

```
# cd /usr/ports/java/openjdk6
#  make install clean
```

Дараа нь java/icedtea-web-г портын цуглуулгаас суулгана:

```
# cd /usr/ports/java/icedtea-web
#  make install clean
```

Chromium-г эхлүүлж хаягийн цонхон дээр `about:plugins` гэж бичнэ. IcedTea-Web нь суулгасан залгаасууд дотор харагдах ёстой.

Хэрэв Chromium нь IcedTea-Web залгаасыг харуулахгүй байгаа бол дараах тушаалыг ажиллуулж вэб хөтчийг дахин эхлүүлнэ:

```
# mkdir -p /usr/local/shared/chromium/plugins
# ln -s /usr/local/lib/IcedTeaPlugin.so \
  /usr/local/shared/chromium/plugins/
```

### 7.2.9. Chromium ба Adobe® Flash™ залгаас

Энэ хэсэг Chromium аль хэдийн суусан гэж үздэг.

Chromium ба Adobe® Flash™-г тохируулах нь Firefox-н заавартай төстэй. Adobe® Flash™-г FreeBSD дээр суулгах талаар илүү дэлгэрэнгүйг тэр хэсгээс үзээрэй. Chromium нь бусад хөтчийн зарим залгаасуудыг ашиглаж чаддаг болохоор илүү нэмэлт тохиргоо шаардлагагүй.

## 7.3. Бүтээмж

Бүтээмжтэй холбоотой болоод ирэхээрээ, шинэ хэрэглэгчид ихэвчлэн сайн оффисийн цуглуулга эсвэл нөхөрсөг текст боловсруулагчийг хайдаг. KDE зэрэг зарим ширээний орчнууд нь хэдийгээр оффисийн цуглуулгатай байдаг боловч анхдагч бүтээмжийн багц байдаггүй билээ. FreeBSD нь таны ширээний орчноос хамааралгүйгээр хэрэгтэй бүгдийг хангадаг.

Энэ хэсэг нь эдгээр програмуудыг авч үзэх болно:

| Програмын нэр | Шаардлагатай эх үүсвэрүүд | Портоос суулгах | Гол хамаарлууд |
|---|---|---|---|
| KOffice | бага | хүнд | KDE |
| AbiWord | бага | хөнгөн | Gtk+ эсвэл GNOME |
| Gimp | бага | хүнд | Gtk+ |
| OpenOffice.org | их | асар их | JDK™, Mozilla |

| Програмын нэр | Шаардлагатай эх үүсвэрүүд | Портоос суулгах | Гол хамаарлууд |
|---|---|---|---|
| LibreOffice | хүнддүү | асар их | Gtk+, эсвэл KDE/ GNOME, эсвэл JDK™ |

### 7.3.1. KOffice

KDE-ийн хүрээнийхэн KDE-ээс гадна ашиглагдаж болох оффисийн цуглуулгатай өөрийн ширээний орчинг өгдөг. Энэ нь бусад оффисийн цуглуулгад байдаг дөрвөн стандарт бүрэлдэхүүнийг агуулдаг. KWord нь текст боловсруулагч, KSpread нь хүснэгттэй ажилладаг програм, KPresenter нь танилцуулгуудыг удирддаг бөгөөд Kontour нь график баримтуудыг зурахыг зөвшөөрнө.

Хамгийн сүүлчийн KOffice-г суулгахаасаа өмнө та KDE-ийн хамгийн шинэчлэгдсэн хувилбартай эсэхээ шалгаарай.

KDE4-т зориулсан KOffice-ийг багц хэлбэрээр суулгахын тулд дараах тушаалыг ажиллуулна:

```
# pkg_add -r koffice-kde4
```

Хэрэв багц байхгүй бол та Портын цуглуулгыг ашиглаж болно. Жишээ нь KDE4-т зориулж KOffice-г суулгахын тулд доор дурдсаныг ажиллуулна:

```
# cd /usr/ports/editors/koffice-kde4
# make install clean
```

### 7.3.2. AbiWord

AbiWord нь Microsoft® Word-той төстэй харагдаж, санагддаг, чөлөөтэй, текст боловсруулагч програм юм. Энэ нь бичиг, захидал, тайлан, сануулга гэх зэргүүдийг бичихэд тохиромжтой юм. Энэ нь маш хурднаас гадна олон боломжуудыг агуулдаг бөгөөд хэрэглэгчид маш хялбар байдаг.

AbiWord нь Microsoft®-ийн .doc зэрэг арилжааныхыг оруулаад олон файлын хэлбэршилтүүдийг импорт болон экспорт хийж чаддаг.

AbiWord нь багц хэлбэрээр байдаг. Та дараах тушаалыг ажиллуулан үүнийг суулгаж болно:

```
# pkg_add -r abiword
```

Хэрэв багц байхгүй бол үүнийг Портын цуглуулгаас эмхэтгэж болно. Портын цуглуулга шинэчлэгдсэн байх ёстой. Үүнийг ингэж хийж болно:

```
# cd /usr/ports/editors/abiword
```

```
# make install clean
```

### 7.3.3. GIMP

Зургийг зохиох эсвэл дүрс тодруулах тохиолдолд GIMP нь маш нарийн, зураггай ажиллах програм юм. Энэ нь энгийн будах програм эсвэл чанартай, фото зураг тодруулах цуглуулга болгон ашиглагдаж болох юм. Энэ нь олон тооны нэмэлт залгаасыг (plug-ins) дэмждэг бөгөөд скрипт хийх интерфэйсийн боломжуудтай байдаг. GIMP нь төрөл бүрийн файлын хэлбэршилтүүдийг уншиж, бичиж чаддаг. Энэ нь сканнер болон зөөврийн цахим төхөөрөмжтэй (tablet) холбогдох боломжийг дэмждэг.

Та энэ тушаалыг ажиллуулан багцыг суулгаж болно:

```
# pkg_add -r gimp
```

Хэрэв таны FTP сайт энэ багцгүй байгаа бол та Портын цуглуулгыг ашиглаж болно. Портын цуглуулгын graphics сан бас Gimp Manual програмыг агуулдаг. Энд тэдгээрийг хэрхэн суулгах талаар үзүүлэв:

```
# cd /usr/ports/graphics/gimp
# make install clean
# cd /usr/ports/graphics/gimp-manual-pdf
# make install clean
```

Портын цуглуулгын graphics сан GIMP-ийн хөгжүүлэлтийн хувилбарыг graphics/gimp-devel санд агуулдаг. Gimp Manual-ийн HTML хувилбар нь graphics/gimp-manual-html санд байдаг.

### 7.3.4. OpenOffice.org

OpenOffice.org нь бүх шаардлагатай програмуудыг бүрэн гүйцэд оффисийн бүтээмжийн цуглуулгадаа агуулсан байдаг: энд текст боловсруулагч, хүснэгттэй ажиллах програм, танилцуулгын менежер, болон зургийн програмыг дурдаж болно. Уг програмын хэрэглэгчийн интерфэйс нь бусад оффисийн цуглуулгуудтай их төстэй бөгөөд төрөл бүрийн файлын хэлбэршилтүүдийг импорт, экспорт хийж чаддаг байна. Энэ нь интерфэйсүүд, алдаа шалгагчид болон тольнуудыг оролцуулаад хэд хэдэн өөр хэлнүүд дээр байдаг.

OpenOffice.org-ийн текст боловсруулагч нь эх XML файлын хэлбэршилтийг зөөвөрлөгдөх боломж ба уян хатан чанарыг сайжруулах зорилгоор ашигладаг. Хүснэгттэй ажиллах програм нь макро хэлний боломжтой бөгөөд гаднах мэдээллийн баазуудтай холбогдох боломжтой байдаг. OpenOffice.org нь аль хэдийн тогтвортой болсон бөгөөд Windows®, Solaris™, Линукс, FreeBSD, болон Mac OS® X дээр ажилладаг. OpenOffice.org-ийн тухай илүү мэдээллийг OpenOffice.org вэб сайтаас олж болно. FreeBSD-тэй холбоотой мэдээлэл болон багцуудыг шууд татаж авахын тулд FreeBSD OpenOffice.org Хөрвүүлэх Багийн вэб сайтыг ашиглана уу.

OpenOffice.org-ийг суулгахын тулд доор дурдсаныг ажиллуулна:

```
# pkg_add -r openoffice.org
```

FreeBSD-ийн -RELEASE хувилбарыг ажиллуулж байхад энэ нь ажиллах ёстой. Үгүй бол та тохирох багцыг татан авч pkg_add(1) ашиглан суулгахын тулд FreeBSD-ийн OpenOffice.org Хөрвүүлэх Багийн вэб хаягт хандах хэрэгтэй. Одоогийн хувилбар болон хөгжүүлэлтийн хувилбар нь энд татаж авагдахаар байрладаг.

Багц суулгагдсаны дараа OpenOffice.org-ийг ажиллуулахын тулд та ердөө л дараах тушаалыг бичих хэрэгтэй:

```
% openoffice.org
```

Эхний удаа ажиллуулахад танаас зарим нэг асуултууд асуух бөгөөд .openoffice.org сан таны гэрийн санд үүсгэгдэх болно.

Хэрэв OpenOffice.org багцууд байхгүй бол портыг эмхэтгэх сонголт танд бас байдаг. Гэхдээ энэ нь дискний асар их зай болон эмхэтгэх хугацаа ихийг шаардана гэдгийг та санаж байх ёстой шүү.

```
# cd /usr/ports/editors/openoffice.org-3
# make install clean
```

Хэрэв та локалчлагдсан хувилбарыг бүтээхийг хүсэж байгаа бол түрүүчийн тушаалын мөрийг дараах мөрөөр солих хэрэгтэй:

```
# make LOCALIZED_LANG=your_language install clean
```

Та *your_language* мөрийг зөв хэлний ISO кодоор солих хэрэгтэй. Дэмжигдсэн хэлний кодуудын жагсаалт портын санд байх files/Makefile.localized файлд байдаг.

Үүнийг хийсний дараа OpenOffice.org-г дараах тушаалаар ажиллуулж болно:

```
% openoffice.org
```

### 7.3.5. LibreOffice

LibreOffice нь Бичиг баримтын сангаас хөгжүүлсэн чөлөөтэй оффисын програм хангамж бөгөөд бусад гол гол оффисын програм хангамжуудтай нийцтэй, төрөл бүрийн тавцан дээр

ажилладаг юм. Энэ нь оффисын бүрэн хэмжээний програм хангамжуудад байдаг бичиг баримт боловсруулагч, хүснэгтийн програм, танилцуулга бэлдэгч, зургийн програм, өгөгдлийн сан удирдах програм, болон математикийн томъёонуудыг үүсгэж засварлах зориулалттай хэрэгсэл зэрэг гол гол програмуудыг агуулсан OpenOffice.org-н нэгэн хувилбар юм. Энэ хэд хэдэн төрлийн хэл дээр байдаг бөгөөд өөр хэл рүү хөрвүүлэх ажил интерфэйс, үг үсгийн алдаа шалгагч болон толь бичгүүдэд хүрсэн байна.

LibreOffice-н бичиг баримттай ажиллах програм нь хөрвөх боломж, уян хатан чанарыг хангахын тулд эх XML файлын форматыг ашигладаг. Хүснэгттэй ажилладаг програм нь макро хэлтэй бөгөөд гадаад өгөгдлийн сангуудтай холбогдож болдог. LibreOffice нь аль хэдийн тогтвортой болсон бөгөөд Windows®, Linux, FreeBSD, ба Mac OS® X дээр ажилладаг байна. LibreOffice -н талаар дэлгэрэнгүйг LibreOffice-н вэб хуудаснаас олж болно.

LibreOffice-г багц хэлбэрээр суулгахын тулд:

```
# pkg_add -r libreoffice
```

FreeBSD-н -RELEASE хувилбарыг ажиллуулж байгаа үед энэ нь ажиллах ёстой.

Багц суусны дараа LibreOffice-г ажиллуулахын тулд та дараах тушаалыг өгөх ёстой:

```
% libreoffice
```

Эхний удаагийн ажиллах явцад танаас хэдэн асуулт асуух бөгөөд .libreoffice хавтас таны гэрийн санд үүсэх болно.

Хэрэв LibreOffice багцууд байхгүй бол та портыг бүтээх боломж бас бий. Гэхдээ энэ нь маш их дискний хэмжээ болон бүтээхэд нэлээн удаан хугацаа шаардах болно гэдгийг санаарай.

```
# cd /usr/ports/editors/libreoffice
# make install clean
```

Хэрэв танд локалчлагдсан хувилбар хэрэгтэй бол өмнөх тушаалыг дараахаар солиорой:

```
# make LOCALIZED_LANG=your_language install clean
```

Та *your_language* гэдгийг ISO-код бүхий зөв хэлээрээ солих хэрэгтэй. Дэмжигддэг хэлний кодын жагсаалт Makefile-ийн `pre-fetch` хэсэгт байдаг.

Үүнийг хийсний дараа LibreOffice-г дараах тушаалаар ажиллуулж болно:

```
% libreoffice
```

# 7.4. Бичиг баримт үзүүлэгчид

Баримтын зарим нэгэн шинэ хэлбэршилтүүд UNIX®-ийг бий болсон цагаас эхлэн түгээмэл болсон билээ. Тэдгээрийн шаарддаг стандарт үзүүлэгчид үндсэн системд байхгүй байж болох юм. Бид тэдгээр үзүүлэгчдийг хэрхэн суулгахыг энэ хэсэгт үзэх болно.

Энэ хэсэг нь эдгээр програмуудыг авч үзэх болно:

| Програмын нэр | Шаардлагатай эх үүсвэрүүд | Портоос суулгах | Гол хамаарлууд |
|---|---|---|---|
| Acrobat Reader® | бага | хөнгөн | Линуксийн хоёртын хэлбэртэй нийцтэй байдал |
| gv | бага | хөнгөн | Xaw3d |
| Xpdf | бага | хөнгөн | FreeType |
| GQview | бага | хөнгөн | Gtk+ эсвэл GNOME |

## 7.4.1. Acrobat Reader®

Олон баримтууд нь одоо PDF файл хэлбэрээр тараагддаг бөгөөд энэ нь "Portable Document Format" буюу зөөврийн баримтын хэлбэршилт гэгддэг. Эдгээр файлуудын төрлүүдэд зориулсан, зөвлөдөг үзүүлэгчдийн нэг нь Adobe-оос гаргасан Линуксд зориулсан Acrobat Reader® юм. FreeBSD нь Линуксийн хоёртын хэлбэрийн програмуудыг ажиллуулж чаддаг болохоор энэ нь FreeBSD-д бас байдаг юм.

Портын цуглуулгаас Acrobat Reader® 8-г суулгахын тулд доор дурдсаныг хийнэ:

```
# cd /usr/ports/print/acroread8
# make install clean
```

Лицензийн хязгаарлалтуудаас болоод багц байдаггүй.

## 7.4.2. gv

gv нь PostScript® болон PDF үзүүлэгч юм. Энэ нь эхлээд ghostview дээр суурилсан боловч Xaw3d сангийн тусламжтай илүү сайхан харагдах болсон. Энэ нь хурдан бөгөөд интерфэйс нь цэвэрхэн байдаг. gv нь байрлуулалт, цаасны хэмжээ, масштаб эсвэл antialias зэрэг олон боломжуудтай байдаг. Бараг бүх үйлдлүүд нь гараас эсвэл хулганаас хийгдэж болно.

gv-г багц хэлбэрээр суулгахын тулд доор дурдсаныг хийнэ:

```
# pkg_add -r gv
```

Хэрэв та багцыг авч чадахгүй байгаа бол Портын цуглуулгыг ашиглаж болно:

```
# cd /usr/ports/print/gv
# make install clean
```

### 7.4.3. Xpdf

Хэрэв та жижиг FreeBSD PDF үзүүлэгчийг хүсэж байгаа бол Xpdf нь хөнгөн бөгөөд үр ашигтай үзүүлэгч юм. Энэ нь маш цөөн эх үүсвэрүүдийг шаарддаг бөгөөд маш тогтвортой ажилладаг. Стандарт X фонтуудыг ашигладаг бөгөөд Motif® юм уу эсвэл бусад дурын X хэрэгслүүдийг шаарддаггүй.

Xpdf багцыг суулгахын тулд энэ тушаалыг ажиллуулна:

```
# pkg_add -r xpdf
```

Багц байхгүй юм уу эсвэл та Портын цуглуулга ашиглахыг илүүд үзэж байгаа бол доор дурдсаныг хийнэ:

```
# cd /usr/ports/graphics/xpdf
# make install clean
```

Суулгац дууссаны дараа та Xpdf-г ажиллуулж хулганы баруун товчийг ашиглаж цэсийг идэвхжүүлж болно.

### 7.4.4. GQview

GQview нь зургийн менежер юм. Та файлыг ганц даралтаар үзэх, гадаад засварлагчийг ажиллуулах, зургийг жижгээр урьдчилан үзүүлэх зэрэг олон зүйлсийг хийж болох юм. Энэ нь бас slideshow буюу цомог үзүүлэх горим болон файлын зарим нэг үндсэн үйлдлүүдтэй. Та зургийн цуглуулгуудыг удирдаж давтагдаж байгааг нь хялбархан олж болно. GQview нь бүрэн дэлгэцээр үзүүлэх болон интернационалчлалыг дэмждэг.

Хэрэв та GQview багцыг суулгахыг хүсэж байгаа бол доор дурдсаныг хийнэ:

```
# pkg_add -r gqview
```

Багц байхгүй юм уу эсвэл та Портын цуглуулга ашиглахыг илүүд үзэж байгаа бол доор дурдсаныг хийнэ:

```
# cd /usr/ports/graphics/gqview
```

```
# make install clean
```

# 7.5. Санхүү

Хэрэв ямар нэгэн шалтгаанаар та өөрийн хувийн санхүүг FreeBSD ширээний компьютер дээрээ удирдахыг хүсэж байгаа бол хүчирхэг, ашиглахад хялбар, суулгахад бэлэн зарим нэг програмууд байдаг. Тэдгээрийн зарим нэг нь Quicken эсвэл Excel баримтуудад хэрэглэгддэгтэй адил төрөл бүрийн файлын хэлбэршилтүүдтэй нийцтэй байдаг.

Энэ хэсэг нь эдгээр програмуудыг авч үзэх болно:

| **Програмын нэр** | **Шаардлагатай эх үүсвэрүүд** | **Портоос суулгах** | **Гол хамаарлууд** |
|---|---|---|---|
| GnuCash | бага | хүнд | GNOME |
| Gnumeric | бага | хүнд | GNOME |
| Abacus | бага | хөнгөн | Tcl/Tk |
| KMyMoney | бага | хүнд | KDE |

## 7.5.1. GnuCash

GnuCash нь эцсийн хэрэглэгчдэд хэрэглэхэд хялбар, бас хүчирхэг програмуудын боломжийг олгох GNOME-ийн чармайлтын нэг хэсэг юм. GnuCash-ийн тусламжтай та өөрийн орлого болон зарлагууд, банкны данснууд болон хувьцаануудаа хянаж чадах юм. Энэ нь хялбар интерфэйстэйгээс гадна маш мэргэжлийн програм юм.

GnuCash нь ухаалаг регистр, системийн шаталсан бүртгэлүүд, гарын олон хурдасгагчид болон автоматаар гүйцээх аргуудтай байдаг. Энэ нь ганц гүйлгээг хэд хэдэн, илүү дэлгэрэнгүй хэсгүүдэд хувааж чаддаг. GnuCash нь Quicken QIF файлуудыг импорт хийн нийлүүлж чаддаг. Мөн олон улсын огноо болон мөнгөн тэмдэгтийн хэлбэрүүдтэй ажиллаж бас чаддаг байна.

Өөрийн систем дээрээ GnuCash-ийг суулгахын тулд доор дурдсаныг хийнэ:

```
# pkg_add -r gnucash
```

Хэрэв багц байхгүй байгаа бол та Портын цуглуулгыг ашиглаж болно:

```
# cd /usr/ports/finance/gnucash
# make install clean
```

## 7.5.2. Gnumeric

Gnumeric нь хүснэгттэй ажилладаг програм бөгөөд GNOME ширээний орчны хэсэг юм. Энэ нь хэрэглэхэд хялбар, олон дарааллын хувьд автоматаар дүүргэх систем бүхий үүрний

хэлбэрийн дагуу хэрэглэгчийн оруулж байгаа зүйлийг автоматаар "таах" боломжтой байдаг. Бас Excel, Lotus 1-2-3, эсвэл Quattro Pro зэрэг хэд хэдэн түгээмэл хэлбэршилтүүдээс файлуудыг импорт хийж чаддаг. Gnumeric нь зураг зурах math/guppi програмын тусламжтай зураг зуралтыг дэмждэг. Бас маш олон тооны цуг бүтээгдсэн функцуудтай бөгөөд тоо, мөнгөн тэмдэгт, огноо, цаг зэрэг олон, бүх энгийн үүрний хэлбэрүүдийг зөвшөөрдөг.

Gnumeric-ийг багцаас суулгахын тулд доор дурдсаныг хийнэ:

```
# pkg_add -r gnumeric
```

Хэрэв багц байхгүй бол та доор дурдсаныг хийн Портын цуглуулгыг ашиглаж болно:

```
# cd /usr/ports/math/gnumeric
# make install clean
```

### 7.5.3. Abacus

Abacus нь жижиг бөгөөд ашиглахад хялбар хүснэгттэй ажиллах програм юм. Энэ нь статистик, санхүү болон математик зэрэг хэд хэдэн салбарт хэрэгтэй, өөрт нь цуг бүтээгдсэн функцуудтай байдаг. Энэ нь Excel файлын хэлбэршилтийг импорт, экспорт хийж чаддаг. Мөн Abacus нь PostScript® гаралтаар гаргаж чаддаг байна.

Abacus-ийг багц хэлбэрээр нь суулгахын тулд доор дурдсаныг ажиллуулна:

```
# pkg_add -r abacus
```

Хэрэв багц байхгүй бол та доор дурдсаныг хийн Портын цуглуулгыг ашиглаж болно:

```
# cd /usr/ports/deskutils/abacus
# make install clean
```

### 7.5.4. KMyMoney

KMyMoney нь KDE-д зориулсан хувийн санхүүгийн менежер юм. KMyMoney нь арилжааны зорилгоор хийгдсэн хувийн санхүүгийн менежер програмуудад байдаг бүх л чухал боломжуудыг өөртөө агуулсан байдаг. Бас ашиглахад хялбар, дахин оруулах зөв бүртгэлийн систем зэрэг боломжуудыг нь дурдаж болох юм. KMyMoney нь стандарт Quicken Interchange Format (QIF) бүхий файлуудаас импорт хийж чаддагаас гадна хөрөнгө оруулалтыг хянах, олон төрлийн мөнгөн тэмдэгтүүдтэй ажиллаж чаддаг бөгөөд баялаг тайлангуудыг гаргаж чаддаг. OFX импорт хийх боломжууд нь бас тусдаа залгаасын тусламжтай хийгдэж болдог байна.

KMyMoney-г багц хэлбэрээс суулгахын тулд дараахийг хийнэ:

```
# pkg_add -r kmymoney2
```

Хэрэв багц байхгүй бол Портын цуглуулгаас суулгаж болно:

```
# cd /usr/ports/finance/kmymoney2
# make install clean
```

## 7.6. Дүгнэлт

FreeBSD нь ISP-уудын дунд өөрийн ажиллагаа болон найдвартай байдлаараа алдартай байдаг ч гэсэн түүнийг өдөр тутам ширээний компьютер болгон ашиглахад бараг л бэлэн болжээ. Багцууд эсвэл портууд хэлбэрээр хэдэн мянган програмуудтай болохоор өөрийн чинь бүх хэрэгцээг хангасан төгс ширээний компьютерийг та бүтээж болох юм.

Энэ бүлэгт авч үзсэн ширээний бүх програмуудын тоймыг энд дурдав:

| Програмын нэр | Багцын нэр | Портын нэр |
|---|---|---|
| Opera | opera | www/opera |
| Firefox | firefox | www/firefox |
| Chromium | chromium | www/chromium |
| KOffice | koffice-kde4 | editors/koffice-kde4 |
| AbiWord | abiword | editors/abiword |
| GIMP | gimp | graphics/gimp |
| Apache OpenOffice.org | openoffice | editors/openoffice-3 |
| LibreOffice | libreoffice | editors/libreoffice |
| Acrobat Reader® | acroread | print/acroread8 |
| gv | gv | print/gv |
| Xpdf | xpdf | graphics/xpdf |
| GQview | gqview | graphics/gqview |
| GnuCash | gnucash | finance/gnucash |
| Gnumeric | gnumeric | math/gnumeric |
| Abacus | abacus | deskutils/abacus |
| KMyMoney | kmymoney2 | finance/kmymoney2 |

# Chapter 8. Мультимедиа

## 8.1. Ерөнхий агуулга

FreeBSD нь танд өөрийн компьютераас гарах өндөр чанарын гаралтад сэтгэл ханамжтай байх боломжийг олгож төрөл бүрийн дууны картуудыг дэмждэг. Энэ нь аудиог MPEG Аудио Давхарга 3 (MP3), WAV, болон Ogg Vorbis зэрэг бусад олон хэлбэршилтүүдээр бичлэг хийх болон тоглуулах чадвартай байдаг. FreeBSD-ийн портын цуглуулга нь таны бичигдсэн аудиог засварлах, дууны нөлөөллүүдийг нэмэх болон залгагдсан MIDI төхөөрөмжүүд хянахыг зөвшөөрөх програмуудыг бас агуулдаг.

Зарим нэг туршилтын тусламжтайгаар FreeBSD нь видео файлууд болон DVD-үүдийг тоглуулахыг дэмжиж чаддаг. Төрөл бүрийн видео зөөвөрлөгчийг кодчилох, хөрвүүлэх, болон тоглуулах програмуудын тоо дууны програмуудын тоотой харьцуулах юм бол хязгаарлагдмал байдаг. Жишээ нь, үүнийг бичиж байх үед, audio/sox-той адил хэлбэршилтүүдийн хооронд хөрвүүлэхэд хэрэглэгдэх дахин кодчилох сайн програм FreeBSD-ийн портын цуглуулгад байхгүй байгаа билээ. Гэхдээ энэ талбар дахь програм хангамжийн байдал хурдацтай өөрчлөгдөж байна.

Энэ бүлэг нь таны дууны картыг тохируулахад шаардлагатай алхмуудыг тайлбарлах болно. X11-ийн (X Цонхот систем) тохиргоо болон суулгац нь таны видео тоног төхөөрөмжийн асуудлуудыг аль хэдийн зохицуулсан байдаг боловч илүү сайн тоглуулахын тулд зарим нэгэн нарийн тохируулгууд хийх хэрэгтэй байж болох юм.

Энэ бүлгийг уншиж дууссаны дараа та дараах зүйлсийг мэдэх болно:

- Таны дууны карт чинь танигдсан байхаар өөрийн системээ хэрхэн тохируулах талаар.
- Таны карт ажиллаж байгааг шалгах аргууд.
- Өөрийн дууны тохиргооны алдааг хэрхэн олж засварлах талаар.
- MP3-ууд болон бусад аудиог хэрхэн тоглуулах ба кодчилох талаар.
- X серверээр видео хэрхэн дэмжигдсэн талаар.
- Сайн үр дүн өгдөг зарим нэгэн тоглуулагч/кодчилогч портууд.
- DVD-үүд, .mpg болон .avi файлуудыг хэрхэн тоглуулах талаар.
- CD болон DVD-ээс файлууд уруу агуулгыг хэрхэн авах талаар.
- TV карт хэрхэн тохируулах талаар.
- Дүрс скан хийгчийг хэрхэн тохируулах талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- Шинэ цөмийг хэрхэн тохируулж суулгах талаар мэдэх (FreeBSD цөмийг тохируулах нь).

mount(8) тушаалаар аудио CD-үүдийг холбохыг оролдох нь хамгийн гайгүйдээ алдаанд, хамгийн муудаа *цөмийн сүйрэл*д хүргэж болох юм. Эдгээр зөөвөрлөгчид нь ердийн ISO файлын системүүдээс ялгаатай тусгай

> кодчилолуудтай байдаг.

# 8.2. Дууны картыг тохируулах

## 8.2.1. Системийг тохируулах

Эхлэхээсээ өмнө та өөртөө байгаа картынхаа загвар, ашиглаж байгаа бичил схем болон PCI эсвэл ISA картны алин болохыг мэдэх шаардлагатай. FreeBSD нь төрөл бүрийн PCI эсвэл ISA картуудыг дэмждэг. Тоног төхөөрөмжийн тэмдэглэгээний дэмжигдсэн аудио төхөөрөмжүүдийн жагсаалтыг шалгаж өөрийн картыг дэмжигдсэн эсэхийг үзээрэй. Тоног төхөөрөмжийн тэмдэглэгээ нь таны картыг ямар драйвер дэмждэгийг бас хэлэх болно.

Өөрийн дууны төхөөрөмжийг ашиглахын тулд тохирох төхөөрөмжийн драйверийг та ачаалах хэрэгтэй болно. Үүнийг хоёр аргын аль нэгээр хийж болно. Хамгийн амархан арга бол kldload(8) тушаалаар өөрийн картныхаа цөмийн модулийг тушаалын мөрөөс:

```
# kldload snd_emu10k1
```

гэж ачаалах юм уу эсвэл /boot/loader.conf файлд тохирох мөрийг ингэж нэмэн дуудаж болно:

```
snd_emu10k1_load="YES"
```

Эдгээр жишээнүүд нь Creative SoundBlaster® Live! дууны картанд зориулагдсан юм. Бусад байгаа дуудагдаж болох дууны модулиуд /boot/defaults/loader.conf-д жагсаагдсан байдаг. Хэрэв та аль драйверийг ашиглахаа мэдэхгүй эргэлзэж байвал snd_driver модулийг дуудаж үзэж болох юм:

```
# kldload snd_driver
```

Энэ нь хамгийн түгээмэл драйверуудыг нэг удаа ачаалдаг метадрайвер юм. Энэ нь зөв драйверийг хайхыг хурдасгадаг. Дууны бүх драйверуудыг /boot/loader.conf файлаар ачаалах бас боломжтой байдаг.

snd_driver метадрайверийг дуудсаны дараа та өөрийн дууны картдаа ямар драйвер сонгогдсоныг олохыг хүсвэл /dev/sndstat файлыг `cat /dev/sndstat` тушаалын тусламжтай шалгаж болох юм.

Хоёр дахь арга нь өөрийн дууны картын дэмжлэгийг өөрийн цөмдөө эмхэтгэх явдал юм. Доорх хэсэг нь энэ аргаар өөрийн тоног төхөөрөмжид зориулан дэмжлэг нэмэхэд шаардлагатай мэдээллийг өгнө. Өөрийн цөмийг дахин эмхэтгэх тухай дэлгэрэнгүй мэдээллийг FreeBSD цөмийг тохируулах нь-с үзнэ үү.

### 8.2.1.1. Дууны дэмжлэгтэй цөмийг тохируулах

Эхлээд хийх ёстой зүйл нь аудио тогтолцооны драйвер sound(4)-г цөм уруу нэмэх явдал юм.

Ингэхийн тулд та дараах мөрийг цөмийн тохиргооны файлдаа нэмэх хэрэгтэй болно:

```
device sound
```

Дараа нь та өөрийн дууны картдаа зориулж дэмжлэг нэмэх хэрэгтэй. Тийм болохоор картыг ямар драйвер дэмждэгийг мэдэх хэрэгтэй. Өөрийн дууны картын зөв драйверийг тодорхойлохын тулд Тоног төхөөрөмжийн тэмдэглэгээний дэмжигдсэн аудио төхөөрөмжүүдийн жагсаалтыг шалгах хэрэгтэй. Жишээ нь Creative SoundBlaster® Live! дууны карт нь snd_emu10k1(4) драйвераар дэмжигдсэн байдаг. Энэ картанд зориулж дэмжлэг нэмэхийн тулд доор дурдсаныг ашиглана:

```
device snd_emu10k1
```

Хэрхэн ашиглах талаар драйверийн гарын авлагын хуудаснаас уншихаа мартуузай. Дэмжигдсэн дууны драйверуудын цөмийн тохиргооны зөв бичиглэлийг /usr/src/sys/conf/NOTES файлаас бас олж болно.

PnP биш ISA дууны картууд нь цөмийг картын тохируулгуудын (IRQ, I/O порт, гэх мэт) тухай мэдээлэлтэй нь танаас шаардаж болох юм. Энэ нь бүх PnP биш ISA дууны картуудын хувьд үнэн байдаг. Үүнийг /boot/device.hints файлын тусламжтай хийдэг. Системийг ачаалах үед loader(8) нь энэ файлыг уншиж тохируулгуудыг цөм уруу дамжуулдаг. Жишээ нь хуучин Creative SoundBlaster® 16 ISA PnP биш карт snd_sbc(4) драйверийг `snd_sb16`-тай цуг ашиглах болно. Энэ картны хувьд дараах мөрүүдийг цөмийн тохиргооны файлд нэмэх хэрэгтэй:

```
device snd_sbc
device snd_sb16
```

бөгөөд бас доор дурдсаныг /boot/device.hints-д нэмэх хэрэгтэй:

```
hint.sbc.0.at="isa"
hint.sbc.0.port="0x220"
hint.sbc.0.irq="5"
hint.sbc.0.drq="1"
hint.sbc.0.flags="0x15"
```

Энэ тохиолдолд карт нь `0x220` I/O порт болон IRQ `5`-ыг ашиглаж байна.

/boot/device.hints файлд ашигласан зөв бичиглэлийн талаар sound(4) драйверийн гарын авлагын хуудас болон ашиглах гэж байгаа драйверийн гарын авлагын хуудсанд өгүүлсэн байдаг.

Дээр үзүүлсэн тохируулгууд нь анхдагчууд юм. Зарим тохиолдолд та IRQ юм уу эсвэл бусад тохируулгуудыг өөрийн картдаа тааруулан өөрчлөх хэрэг гарч болох юм. Энэ картны талаар дэлгэрэнгүй мэдээллийг snd_sbc(4) гарын авлагын хуудаснаас үзнэ үү.

### 8.2.2. Дууны картыг тест хийх

Өөрчилсөн цөмийг дахин ачаалсны дараа эсвэл шаардлагатай модулийг дуудсаны дараа дууны карт нь иймэрхүүгээр таны системийн мэдэгдлийн буферт (dmesg(8)) гарч ирэх ёстой:

```
pcm0: <Intel ICH3 (82801CA)> port 0xdc80-0xdcbf,0xd800-0xd8ff irq 5 at device 31.5 on
pci0
pcm0: [GIANT-LOCKED]
pcm0: <Cirrus Logic CS4205 AC97 Codec>
```

Дууны картын төлөвийг /dev/sndstat файлын тусламжтай шалгаж болно:

```
# cat /dev/sndstat
FreeBSD Audio Driver (newpcm)
Installed devices:
pcm0: <Intel ICH3 (82801CA)> at io 0xd800, 0xdc80 irq 5 bufsz 16384
kld snd_ich (1p/2r/0v channels duplex default)
```

Таны системээс гарах гаралт өөр өөр байж болно. Хэрэв ямар ч pcm төхөөрөмжүүд гарч ирэхгүй байвал буцаад урьд нь юу хийснийгээ дахин шалгах хэрэгтэй. Өөрийн цөмийн тохиргооны файлыг дахин шалгаж зөв төхөөрөмжийн драйвер сонгогдсон эсэхийг шалгаарай. Нийтлэг асуудлууд Нийтлэг асуудлууд хэсэгт жагсаагдсан байгаа.

Хэрэв бүгд зүгээр болвол одоо та ажиллагаатай дууны карттай байх ёстой. Хэрэв таны CD-ROM эсвэл DVD-ROM хөтчийн дууны гаралтын холбогч таны дууны карттай зөв холбогдсон бол та хөтөчдөө CD хийж түүнийг cdcontrol(1)-оор тоглуулж болно:

```
% cdcontrol -f /dev/acd0 play 1
```

audio/workman зэрэг төрөл бүрийн програмууд нь илүү нөхөрсөг интерфэйсийн боломжтой байдаг. Та MP3 аудио файлууд сонсохын тулд audio/mpg123 зэрэг програмыг суулгахыг хүсэж болох юм.

Картыг хурдан тест хийх өөр нэг арга бол /dev/dsp уруу өгөгдөл иймэрхүү маягаар илгээх явдал юм:

```
% cat filename > /dev/dsp
```

Энд байгаа filename нь ямар ч файл байж болно. Энэ тушаалын мөр нь шуугиан үүсгэх ёстой бөгөөд энэ нь уг дууны карт ажиллаж байгааг баталж байгаа юм.

Төхөөрөмжийн цэгүүд болох /dev/dsp* шаардлагатай тохиолдолд автоматаар үүсдэг. Хэрэв тэдгээр нь ашиглагдаагүй бол байхгүй байх бөгөөд ls(1) тушаалын гаралтад харагдахгүй байна.

Дууны картын холигчийн түвшингүүдийг mixer(8) тушаалаар өөрчилж болно. Илүү дэлгэрэнгүйг mixer(8) гарын авлагын хуудаснаас олж болно.

#### 8.2.2.1. Нийтлэг асуудлууд

| Алдаа | Шийдэл |
|---|---|
| `sb_dspwr(XX) timed out` | I/O порт зөв тохируулагдаагүй. |
| `bad irq XX` | IRQ буруу тохируулагдсан. Тохируулагдсан IRQ болон дууны IRQ адил байгаа эсэхийг шалгана. |
| `xxx: gus pcm not attached, out of memory` | Төхөөрөмжийн ашиглах хангалттай санах ой байхгүй байна. |
| `xxx: can't open /dev/dsp!` | Өөр програм төхөөрөмжийг онгорхой барьж байгаа эсэхийг `fstat \| grep dsp` тушаалаар шалгана. Дурдахад илүүдэхгүй хэргийн эзэд бол esound болон KDE-ийн дууны дэмжлэгүүд юм. |

Өөр нэг асуудал бол орчин үеийн график картууд ихэвчлэн HDMI төст зүйлсүүдтэй ажиллахын тулд өөрсдийн дууны драйвертай ирдэг. Энэ дууны төхөөрөмж нь жинхэнэ дууны картаас өмнө танигдах тохиолдол байдаг бөгөөд ингэснээр дууны карт нь анхдагч тоглуулагч төхөөрөмж хэлбэрээр ашиглагдахгүй болдог. Үүнийг шалгахын тулд dmesg ажиллуулж `pcm` гэж байгаа эсэхийг шалгаарай. Гаралт нэг иймэрхүү байна:

```
...
hdac0: HDA Driver Revision: 20100226_0142
hdac1: HDA Driver Revision: 20100226_0142
hdac0: HDA Codec #0: NVidia (Unknown)
hdac0: HDA Codec #1: NVidia (Unknown)
hdac0: HDA Codec #2: NVidia (Unknown)
hdac0: HDA Codec #3: NVidia (Unknown)
pcm0: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 0 nid 1 on hdac0
pcm1: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 1 nid 1 on hdac0
pcm2: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 2 nid 1 on hdac0
pcm3: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 3 nid 1 on hdac0
hdac1: HDA Codec #2: Realtek ALC889
pcm4: <HDA Realtek ALC889 PCM #0 Analog> at cad 2 nid 1 on hdac1
pcm5: <HDA Realtek ALC889 PCM #1 Analog> at cad 2 nid 1 on hdac1
pcm6: <HDA Realtek ALC889 PCM #2 Digital> at cad 2 nid 1 on hdac1
pcm7: <HDA Realtek ALC889 PCM #3 Digital> at cad 2 nid 1 on hdac1
...
```

Энд график карт (`NVidia`) нь дууны картаас (`Realtek ALC889`) өмнө танигдсан байна. Дууны картыг анхдагч тоглуулагч төхөөрөмж хэлбэрээр ашиглахын тулд `hw.snd.default_unit`-г тоглуулахад ашиглах хэрэгслээр солих хэрэгтэй, өөрөөр хэлбэл:

```
# sysctl hw.snd.default_unit=n
```

Энд `n` нь ашиглах дууны төхөөрөмжийн тоо бөгөөд жишээн дээр `4` байна. Дараах мөрийг /etc/sysctl.conf файлд нэмж өгөн энэ өөрчлөлтийг байнгын болгож болно:

```
hw.snd.default_unit=4
```

### 8.2.3. Дууны олон эхүүдийг хэрэглэх нь

esound эсвэл artsd нь зарим нэг програмуудтай дууны төхөөрөмжийг хуваалцахыг дэмждэггүй бөгөөд энэ тохиолдолд заримдаа зэрэг тоглуулах чадвартай дууны олон эхүүдтэй байх туйлын сонирхол байдаг.

FreeBSD нь sysctl(8) хэрэгслийн тусламжтай идэвхжүүлж болох *Виртуал Дууны Сувгуудын* тусламжтай үүнийг хийх боломжийг танд олгодог. Виртуал сувгууд нь цөмд дууг холих замаар өөрийн дууны картын тоглуулалтыг олон болгохыг танд зөвшөөрдөг.

Виртуал сувгуудын тоог тохируулах sysctl-ийн гурван хувьсагч байдаг бөгөөд хэрэв та `root` хэрэглэгч бол иймэрхүү маягаар үүнийг тохируулж болно:

```
# sysctl dev.pcm.0.play.vchans=4
# sysctl dev.pcm.0.rec.vchans=4
# sysctl hw.snd.maxautovchans=4
```

Дээрх жишээ нь дөрвөн виртуал сувгийг хуваарилж байгаа бөгөөд энэ тоо нь өдөр тутмын хэрэглээнд зориулагдсан практик тоо юм. `dev.pcm.0.play.vchans=4` болон `dev.pcm.0.rec.vchans=4` нь pcm0-ийн тоглуулах болон бичих виртуал сувгуудын тоо бөгөөд төхөөрөмжийг залгасны дараа тохируулах боломжтой болдог. `hw.snd.maxautovchans` нь kldload(8) ашиглан залгагдах үед шинэ аудио төхөөрөмжид өгөгдөх виртуал сувгуудын тоо юм. pcm төхөөрөмж нь тоног төхөөрөмжийн драйверуудаас тусдаа дуудагдаж болох учраас `hw.snd.maxautovchans` нь дараа нь залгагдах төхөөрөмжүүдэд өгөгдөх виртуал сувгуудын тоог хадгалдаг. Дэлгэрэнгүй мэдээллийг pcm(4) гарын авлагын хуудаснаас үзнэ үү.

Төхөөрөмжийг ашиглаж байхад та виртуал сувгуудын тоог өөрчилж чадахгүй. Эхлээд хөгжим тоглуулагчид юм уу эсвэл дууны дэмонууд зэрэг төхөөрөмжийг ашиглаж байгаа програмуудыг хаах хэрэгтэй.

Зөв pcm төхөөрөмж /dev/dsp0-ийг хүсэх програмд мэдэгдэлгүйгээр автоматаар хуваарилагдах болно.

### 8.2.4. Холигчийн сувгуудын анхдагч утгуудыг тохируулах

Өөр өөр холигчийн сувгуудын анхдагч утгууд нь pcm(4) драйверийн эх кодонд бичигдэж хийгдсэн байдаг. Холигчийн ажиллуулах явцад санагдсан утгуудыг тохируулах боломжийг олгодог олон төрлийн програмууд болон дэмонууд байдаг боловч энэ нь цэвэр шийдэл биш

юм. Холигчийн анхдагч утгуудыг драйверийн түвшинд тохируулж болдог. Үүнийг /boot/device.hints файлд тохирох утгуудыг тодорхойлон хийдэг. Өөрөө хэлбэл:

```
hint.pcm.0.vol="50"
```

Энэ нь pcm(4) модуль ачаалагдах үед дууны сувгийн чангыг анхдагч утга 50 гэж тохируулах болно.

# 8.3. MP3 аудио

MP3 (MPEG Давхарга 3 Аудио) нь CD-тэй ойролцоо дууны чанарт хүрдэг бөгөөд энэ боломжийг таны FreeBSD ажлын станц ашиглахгүйгээр үлдэх ямар нэг шалтгаан байх ёсгүй юм.

## 8.3.1. MP3 тоглуулагчид

Хамгийн алдартай X11 MP3 тоглуулагч бол XMMS (X Multimedia System) юм. Nullsoft-ийн Winamp-тай GUI нь бараг л адилхан болохоор Winamp-ийн арьснуудыг XMMS-д ашиглаж болдог. XMMS нь өөрийн гэсэн залгаасын дэмжлэгтэй бас байдаг.

XMMS нь multimedia/xmms портоос эсвэл багцаас суулгагдаж болдог.

XMMS-ийн интерфэйс нь хялбар бөгөөд тоглуулах жагсаалт, графикийн тэнцүүлэгч болон өөр олон боломжуудтай байдаг. Winamp-ийг мэддэг хүмүүс XMMS-ийг ашиглахад хялбарыг олж мэдэх болно.

audio/mpg123 нь өөр нэг, тушаалын мөрийн MP3 тоглуулагч юм.

mpg123-ийг тушаалын мөрөөс дууны төхөөрөмж болон MP3 файлыг зааж өгөн ажиллуулж болно. Таны дууны төхөөрөмж /dev/dsp1.0 бөгөөд MP3 файл *Foobar-GreatestHits.mp3*-ийг тоглуулахыг хүсэж байгаа гэж үзвэл дараах тушаалыг ажиллуулна:

```
# mpg123 -a /dev/dsp1.0 Foobar-GreatestHits.mp3
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2 and 3.
Version 0.59r (1999/Jun/15). Written and copyrights by Michael Hipp.
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!

Playing MPEG stream from Foobar-GreatestHits.mp3 ...
MPEG 1.0 layer III, 128 kbit/s, 44100 Hz joint-stereo
```

## 8.3.2. CD Аудио замуудыг авах нь

CD эсвэл CD замыг MP3 уруу кодчилохоосоо өмнө CD дээрх аудио өгөгдлийг хатуу хөтөч урууга авах ёстой. Түүхий CDDA (CD Digital Audio) өгөгдлийг WAV файлууд уруу хуулан үүнийг хийдэг.

sysutils/cdrtools цуглуулгын cdda2wav хэрэгсэл нь CD-үүдээс аудио мэдээлэл болон тэдгээртэй холбоотой мэдээллийг авахад хэрэглэгддэг.

Хөтөчид аудио CD байхад дараах тушаалыг (root-ээр) ажиллуулж бүх CD-г тус тусдаа (зам тус бүрийг) WAV файлуудад авч болно:

```
# cdda2wav -D 0,1,0 -B
```

cdda2wav нь ATAPI (IDE) CDROM хөтчүүдийг дэмжих болно. IDE хөтчөөс авахын тулд SCSI нэгжийн дугааруудын оронд төхөөрөмжийн нэрийг заах хэрэгтэй. Жишээ нь 7-р замыг IDE хөтчөөс авахын тулд:

```
# cdda2wav -D /dev/acd0 -t 7
```

-D 0,1,0 тохируулга нь cdrecord -scanbus тушаалын гаралттай таарч байгаа SCSI төхөөрөмж 0,1,0-ийг заана.

Замуудыг авахын тулд -t тохируулгыг доор үзүүлсэн шиг ашиглана:

```
# cdda2wav -D 0,1,0 -t 7
```

Энэ жишээ нь аудио CDROM-ийн долдугаар замыг авч байна. Хэсэг замуудыг авахын тулд, жишээ нь, нэгээс долоо хүртэлх замыг авахын тулд хүрээг зааж өгнө:

```
# cdda2wav -D 0,1,0 -t 1+7
```

dd(1) хэрэгсэл ATAPI хөтчүүд дээрх аудио замуудыг гаргаж авахад бас ашиглагдаж болно. Энэ боломжийн талаар дэлгэрэнгүйг Аудио CD-үүдийг хувилах-с уншина уу.

### 8.3.3. MP3-уудыг кодчилох

Өнөө үед сонгож байгаа mp3 кодчилогч бол Lame юм. Lame нь audio/lame портын модноос олдож болно.

Авсан WAV файлуудыг ашиглан дараах тушаал audio01.wav-г audio01.mp3 болгон хөрвүүлнэ:

```
# lame -h -b 128 \
--tt "Foo Song Title" \
--ta "FooBar Artist" \
--tl "FooBar Album" \
--ty "2001" \
--tc "Ripped and encoded by Foo" \
--tg "Genre" \
audio01.wav audio01.mp3
```

128 кбит нь ашиглагдаж байгаа стандарт MP3 битийн хурд юм шиг байгаа юм. Олон хүмүүс илүү өндөр чанарын 160 эсвэл 192 гэсэн хурдуудад сэтгэл ханамжтай байдаг. Битийн хурд өндөр болох тусам MP3 илүү их дискний хэмжээг ашиглах боловч чанар өндөр болдог. `-h` тохируулга нь "илүү өндөр чанартай боловч бага зэрэг удаан" горимыг идэвхжүүлдэг. `--t`-ээр эхэлсэн тохируулгууд ID3 мэдээллүүдийг заадаг бөгөөд энэ нь MP3 файл дотор оруулах дууны мэдээллийг ихэвчлэн агуулдаг. Нэмэлт кодчилох тохируулгуудыг lame-ийн гарын авлагын хуудаснаас лавлан олж болно.

### 8.3.4. MP3-уудыг декод хийх

Аудио CD-г MP3-уудаас шарахын тул тэдгээр нь шахагдаагүй WAV хэлбэршилт уруу хөрвүүлэгдэх ёстой. XMMS болон mpg123 нь MP3-ийн гаралтыг шахаагүй файлын хэлбэршилт уруу гаргахыг дэмждэг.

XMMS-ээр диск уруу бичихдээ:

1. XMMS-ийг ажиллуулна.
2. Цонхон дээр баруун товчлуурыг дарж XMMS-ийн цэсийг гаргана.
3. `Options`-оос `Preference`-ийг сонгоно.
4. Гаралтын Залгаасыг (Output Plugin) "Disk Writer Plugin" болгон өөрчилнө.
5. `Configure`-г дарна.
6. Шахагдаагүй файлуудыг бичих санг оруулна (эсвэл browse-ийг сонгоно).
7. MP3 файлыг XMMS уруу, дууны чангыг 100% болгон EQ тохиргоонуудыг хааж ердийнхөөрөө дуудна.
8. `Play`-г дарна - XMMS нь MP3 тоглуулж байгаа юм шигээр гарах бөгөөд гэхдээ хөгжим сонсогдохгүй. Энэ нь үнэндээ MP3-ийг файл уруу тоглуулж байгаа юм.
9. MP3-уудыг дахин сонсохын тулд анхдагч Гаралтын Залгаасыг буцааж өмнө нь байсан шигээр тохируулахаа мартуузай.

mpg123-аар stdout уруу бичих:

1. `mpg123 -s audio01.mp3 > audio01.pcm` тушаалыг ажиллуулна

XMMS нь файлыг WAV хэлбэршилтээр бичдэг бол mpg123 нь MP3-ийг түүхий PCM аудио өгөгдөл болгон хөрвүүлдэг. Эдгээр хэлбэршилтүүдийг cdrecord-д аудио CD-үүд хийхийн тулд ашиглаж болно. Та түүхий PCM-ийг burncd(8)-д ашиглах ёстой. Хэрэв та WAV файлуудыг ашиглавал зам болгоны эхэнд богино тик гэсэн чимээг мэдрэх бөгөөд энэ чимээ нь WAV файлын толгой юм. Та WAV файлын толгойг SoX хэрэгсэл ашиглан арилгаж болно (үүнийг audio/sox портоос эсвэл багцаас суулгаж болно):

```
% sox -t wav -r 44100 -s -w -c 2 track.wav track.raw
```

FreeBSD дээр CD шарагчийг ашиглах тухай илүү дэлгэрэнгүй мэдээллийг Оптик зөөвөрлөгчийг (CD-үүд) үүсгэж ашиглах нь-с уншина уу.

# 8.4. Видео тоглуулах

Видео тоглуулах нь маш шинэ, хурдацтай хөгжиж байгаа хэрэглээний талбар юм. Хүлээцтэй байгаарай. Дууны хувьд хийгдсэн шиг бүгд асуудалгүй ажиллахгүй байж болох юм.

Эхлэхээсээ өмнө өөрийн видео картын загвар болон ашиглаж байгаа бичил схемийг та мэдэх шаардлагатай. Xorg нь төрөл бүрийн видео картуудыг дэмждэг боловч цөөхөн нь сайн тоглуулах ажиллагаатай байдаг. Өөрийн картыг ашиглан X серверийн дэмжсэн өргөтгөлүүдийн жагсаалтыг авахын тулд X11-ийг ажиллаж байхад нь xdpyinfo(1) тушаалыг ашиглах хэрэгтэй.

Ер нь богино хэмжээний MPEG файлтай байх нь зүйтэй бөгөөд төрөл бүрийн тоглуулагч болон тохируулгуудыг шалгахад тест файл маягаар ашиглагдаж болох юм. Зарим нэгэн DVD тоглуулагчид анхдагчаар /dev/dvd-д DVD зөөвөрлөгчийг хайх юм уу эсвэл энэ төхөөрөмжийн нэрийг тэдгээрт бичигдсэн байх ёстой гэж үздэг болохоор та симболын холбоосыг тохирох төхөөрөмжүүд уруу хийж өгөх нь ашигтай байж болох юм:

```
# ln -sf /dev/acd0 /dev/dvd
# ln -sf /dev/acd0 /dev/rdvd
```

devfs(5)-ийн хийгдсэн чанараас хамааралд эдгээр шиг гараар хийгдсэн холбоосууд нь системийг дахин ачаалахад үлддэггүй. Өөрийн системийг ачаалахад автоматаар симболын холбоосууд үүсгэхийн тулд дараах мөрүүдийг /etc/devfs.conf файлд нэмнэ:

```
link acd0 dvd
link acd0 rdvd
```

Мөн тусгай DVD-ROM функцуудыг ажиллуулахыг шаарддаг DVD шифрлэлтийг буцаах үйлдэл нь DVD төхөөрөмж дээр бичих зөвшөөрлийг шаарддаг.

Хуваалцсан санах ойн X11 интерфэйсийг өргөтгөж сайжруулахын тулд зарим sysctl(8) хувьсагчдын утгуудыг ихэсгэх шаардлагатай байдаг:

```
kern.ipc.shmmax=67108864
kern.ipc.shmall=32768
```

## 8.4.1. Видео боломжуудыг тодорхойлох нь

X11 дээр видеог үзүүлэх хэд хэдэн боломжит аргууд байдаг. Яг юу ажиллах нь голчлон тоног төхөөрөмжөөс хамаарна. Доор тайлбарласан арга бүр тоног төхөөрөмжөөс хамааралд чанарын хувьд янз бүр байна. Хоёрдугаарт X11 дээр видеог амь оруулах нь сүүлийн үед маш их анхаарал татаж байгаа сэдэв бөгөөд Xorg-ийн хувилбар бүрт чухал сайжруулалт хийгдсэн байж болох юм.

Нийтлэг видео интерфэйсүүдийг дурдвал:

1. X11: хуваалцсан санах ойг ашиглах ердийн X11 гаралт.
2. XVideo: Дурын X11-ийн дүрслэгдэх боломжтой зүйл дээр видеог дэмждэг X11 интерфэйсийн өргөтгөл.
3. SDL: Simple Directmedia Layer буюу энгийн Directmedia давхарга.
4. DGA: Direct Graphics Access буюу шууд графикийн хандалт.
5. SVGAlib: доод түвшний консолын график давхарга.

#### 8.4.1.1. XVideo

Xorg нь видеог тусгай хурдасгууpаар дүрслэгдэхүйц обьектууд дээр шууд үзүүлэхийг зөвшөөрдөг *XVideo* (бас Xvideo, Xv, xv гэгддэг) гэгддэг өргөтгөлтэй байдаг. Энэ өргөтгөл нь доод түвшний чанар муутай машин дээр ч гэсэн видеог сайн чанартай тоглуулах боломжийг олгодог.

Өргөтгөл ажиллаж байгаа эсэхийг шалгахын тулд `xvinfo` тушаалыг ашиглана:

```
% xvinfo
```

Хэрэв үр дүн иймэрхүү харагдвал XVideo нь таны картны хувьд дэмжигдсэн байна:

```
X-Video Extension version 2.2
screen #0
  Adaptor #0: "Savage Streams Engine"
    number of ports: 1
    port base: 43
    operations supported: PutImage
    supported visuals:
      depth 16, visualID 0x22
      depth 16, visualID 0x23
    number of attributes: 5
      "XV_COLORKEY" (range 0 to 16777215)
              client settable attribute
              client gettable attribute (current value is 2110)
      "XV_BRIGHTNESS" (range -128 to 127)
              client settable attribute
              client gettable attribute (current value is 0)
      "XV_CONTRAST" (range 0 to 255)
              client settable attribute
              client gettable attribute (current value is 128)
      "XV_SATURATION" (range 0 to 255)
              client settable attribute
              client gettable attribute (current value is 128)
      "XV_HUE" (range -180 to 180)
              client settable attribute
              client gettable attribute (current value is 0)
```

```
    maximum XvImage size: 1024 x 1024
    Number of image formats: 7
      id: 0x32595559 (YUY2)
        guid: 59555932-0000-0010-8000-00aa00389b71
        bits per pixel: 16
        number of planes: 1
        type: YUV (packed)
      id: 0x32315659 (YV12)
        guid: 59563132-0000-0010-8000-00aa00389b71
        bits per pixel: 12
        number of planes: 3
        type: YUV (planar)
      id: 0x30323449 (I420)
        guid: 49343230-0000-0010-8000-00aa00389b71
        bits per pixel: 12
        number of planes: 3
        type: YUV (planar)
      id: 0x36315652 (RV16)
        guid: 52563135-0000-0000-0000-000000000000
        bits per pixel: 16
        number of planes: 1
        type: RGB (packed)
        depth: 0
        red, green, blue masks: 0x1f, 0x3e0, 0x7c00
      id: 0x35315652 (RV15)
        guid: 52563136-0000-0000-0000-000000000000
        bits per pixel: 16
        number of planes: 1
        type: RGB (packed)
        depth: 0
        red, green, blue masks: 0x1f, 0x7e0, 0xf800
      id: 0x31313259 (Y211)
        guid: 59323131-0000-0010-8000-00aa00389b71
        bits per pixel: 6
        number of planes: 3
        type: YUV (packed)
      id: 0x0
        guid: 00000000-0000-0000-0000-000000000000
        bits per pixel: 0
        number of planes: 0
        type: RGB (packed)
        depth: 1
        red, green, blue masks: 0x0, 0x0, 0x0
```

Жагсаагдсан хэлбэршилтүүд (YUV2, YUV12, гэх мэт) нь XVideo-ийн шийдэл бүрт байдаггүй бөгөөд тэдгээрийн энэ байхгүй явдал нь зарим нэг тоглуулагчдад саад болж болохыг санаарай.

Хэрэв үр дүн иймэрхүү харагдаж байвал:

```
X-Video Extension version 2.2
screen #0
no adaptors present
```

XVideo таны картын хувьд дэмжигдээгүй байх магадлалтай.

Хэрэв XVideo нь таны картын хувьд дэмжигдээгүй байвал амилуулж байгаа видео тооцоололтын шаардлагуудыг таны дэлгэц хангах нь улам илүү хэцүү болно гэсэн үг юм. Таны видео карт болон процессороос хамааранд магадгүй та сэтгэл ханахуйц үр дүнд хүрч болох юм. Ажиллагааг сайжруулах аргуудын талаар та Нэмэлт унших зүйлс хэсгээс нэмэлт уншлага хийх шаардлагатай болж болох юм.

#### 8.4.1.2. Simple Directmedia Layer буюу Энгийн Directmedia давхарга

Simple Directmedia Layer буюу SDL нь Microsoft® Windows®, BeOS, болон UNIX®-ийн хооронд хөрвүүлэх давхарга байхаар зориулагдсан бөгөөд ингэснээр дуу болон графикийн үр ашигтай хэрэглээг бий болгож тавцан хооронд ажиллах програмуудыг хөгжүүлэх боломжтой болсон юм. SDL давхарга нь тоног төхөөрөмжийг доод түвшинд хийсвэрлэх боломжийг олгодог бөгөөд энэ нь заримдаа X11 интерфэйсээс илүү үр дүнтэй байдаг.

SDL нь devel/sdl12-с олдож болно.

#### 8.4.1.3. Direct Graphics Access буюу Шууд Графикийн Хандалт

Direct Graphics Access буюу Шууд Графикийн Хандалт нь X11 өргөтгөл бөгөөд програмд X серверийг орхин өнгөрч framebuffer-ийг шууд өөрчлөх боломжийг олгодог. Энэ нь энэ хуваалцалд нөлөөлөх доод түвшний санах ойн тааруулалт дээр тулгуурладаг болохоор үүнийг ашиглаж байгаа програмууд `root`-ээр ажиллах ёстой байдаг.

DGA өргөтгөл нь dga(1)-аар тест хийгдэж хурд, ажиллагааны хувьд шалгагдаж болно. `dga` ажиллаж байхад товчлуур дарах болгонд дэлгэцийн өнгөнүүдийг өөрчилдөг. Гарахын тулд `q` -г дарна.

### 8.4.2. Видеотой холбоотой портууд болон багцууд

Энэ хэсэг нь FreeBSD-ийн портын цуглуулга дахь видео тоглуулахад ашиглаж болох програм хангамжуудыг хэлэлцэх болно. Видео тоглуулах нь програм хангамжийн хөгжүүлэлтийн маш идэвхтэй талбар бөгөөд төрөл бүрийн програмуудын боломжууд нь энд өгсөн тайлбараас зарим талаараа зөрж болох юм.

Эцэст нь FreeBSD дээр ажилладаг олон видео програмууд нь Линуксийн програмууд маягаар хөгжүүлэгдсэн байдгийг мэдэх нь чухал юм. Эдгээр програмуудын олонхи нь бета чанарых хэвээр байгаа билээ. FreeBSD дээр видео багцуудад учирч болох асуудлуудын заримыг дурдвал:

1. Ямар нэг програмын үүсгэсэн файлыг өөр нэг нь тоглуулж чадахгүй байх.
2. Өөрийн үүсгэсэн файлыг тоглуулж чадахгүй байх.
3. Тухайн машин бүр дээр бүтээгдсэн, хоёр өөр машин дээрх нэг програм ижил файлыг

өөрөөр тоглуулах.

4. Дүрсний хэмжээг өөрчлөх зэрэг маш хялбар шүүлт нь алдаатай хэмжээ өөрчлөх дэг журмаас болж маш муу үзэгдлүүдэд хүргэдэг.
5. Програм байнга core үүсгэдэг.
6. Баримт порттой цуг суугддаггүй бөгөөд түүнийг нэг бол вэб дээрээс юм уу эсвэл портын work сангаас олж болдог.

Эдгээр програмуудын олонхи нь "Линукс-изм"ийг харуулдаг. Линукс тархацууд дахь зарим стандарт сангуудын хийгдсэн арга эсвэл програм зохиогчдын тооцсон Линуксийн цөмийн зарим боломжуудаас болоод асуудлууд гарч болно гэсэн үг юм. Эдгээр асуудлууд нь заримдаа анзаарагддаггүй бөгөөд порт арчлагчид эдгээрийг тойрон гарсан шийдлүүдийг хийсэн байдаг. Гэхдээ энэ нь доорх асуудлуудад хүргэж болзошгүй юм:

1. Процессорын шинж чанаруудыг илрүүлэх /proc/cpuinfo-ийн хэрэглээ.
2. Дуусахдаа програмыг бүрмөсөн төгсгөхийн оронд гацаадаг урсгалуудын (threads) буруу хэрэглээ.
3. Програмтай цуг ихэвчлэн ашиглагддаг програм хангамж FreeBSD-ийн портын цуглуулгад байхгүй байх.

Одоогоор эдгээр програмуудын хөгжүүлэгчид нь порт хийхэд шаардагдах тойрон гарах арга замуудыг багасгахын тулд порт арчлагчидтай хамтран ажиллаж байгаа билээ.

#### 8.4.2.1. MPlayer

MPlayer нь саяхан хөгжүүлэгдсэн бөгөөд хурдацтай хөгжүүлэгдэж байгаа видео тоглуулагч юм. MPlayer багийн зорилго нь хурд болон Линукс ба бусад Юниксууд дээрх уян хатан чанар юм. Энэ төсөл нь багийг үндэслэгч тэр үед байсан тоглуулагчуудын муу тоглуулах ажиллагааанаас залхсан үед үүссэн байна. Зарим нь дизайныг сайжруулж график интерфэйс хийгдсэн гэж хэлж болох юм. Гэхдээ та тушаалын мөрийн тохируулгууд болон товчлууруудын хяналтад дасаад ирэхэд энэ нь харин маш сайн ажилладаг.

##### 8.4.2.1.1. MPlayer-ийг бүтээх

MPlayer нь multimedia/mplayer санд байрладаг. MPlayer нь бүтээх явцад төрөл бүрийн тоног төхөөрөмжийн шалгалтуудыг гүйцэтгэж нэг системээс нөгөөд хөрвүүлэгдэхгүй хоёртын файлыг гаргадаг. Тиймээс хоёртын багцыг ашиглалгүйгээр портоос бүтээх нь чухал юм. Мөн Makefile-д тайлбарласнаар бүтээлтийн эхэнд `make` тушаалд хэд хэдэн тохируулгуудыг зааж өгч болдог:

```
# cd /usr/ports/multimedia/mplayer
# make
N - O - T - E

Take a careful look into the Makefile in order
to learn how to tune mplayer towards you personal preferences!
For example,
make WITH_GTK1
```

```
builds MPlayer with GTK1-GUI support.
If you want to use the GUI, you can either install
/usr/ports/multimedia/mplayer-skins
or download official skin collections from
http://www.mplayerhq.hu/homepage/dload.html
```

Ихэнх хэрэглэгчдэд анхдагч портын тохируулгууд хангалттай байх ёстой. Гэхдээ хэрэв танд XvID кодек хэрэгтэй бол тушаалын мөрөнд `WITH_XVID` тохируулгыг зааж өгөх хэрэгтэй. Анхдагч DVD төхөөрөмжийг бас `WITH_DVD_DEVICE` тохируулгаар тодорхойлж болох бөгөөд анхдагчаар /dev/acd0 ашиглагдах болно.

Үүнийг бичиж байх үед MPlayer порт нь өөрийн HTML баримт болон хоёр ажиллах файл `mplayer`, болон `mencoder` тушаалуудаа бүтээдэг байсан бөгөөд сүүлийн тушаал нь видеог дахин кодчилох хэрэгсэл юм.

MPlayer-ийн HTML баримт нь маш мэдээлэл сайтай байдаг. Хэрэв уншигч энэ бүлэгт дурдсан видео тоног төхөөрөмж болон интерфэйсүүдийн тухай мэдээлэл дутуу байгааг мэдвэл MPlayer-ийн баримт нь харин маш бүрэн гүйцэд нэмэгдэл болох юм. Хэрэв та UNIX® дээрх видео дэмжлэгийн талаар мэдээлэл хайж байгаа бол MPlayer-ийн баримтыг цаг зав гарган заавал ч үгүй унших хэрэгтэй.

#### 8.4.2.1.2. MPlayer ашиглах

MPlayer-ийн ямар ч хэрэглэгч өөрийн гэрийн сандаа .mplayer дэд санг үүсгэх ёстой. Энэ шаардлагатай дэд санг үүсгэхийн тулд та дараах тушаалыг бичиж болно:

```
% cd /usr/ports/multimedia/mplayer
% make install-user
```

`mplayer`-ийн тушаалын тохируулгууд гарын авлагын хуудсанд дурдсан байдаг. Бүр илүү дэлгэрэнгүйг HTML баримтаас үзэж болно. Энэ хэсэгт бид цөөн хэдэн нийтлэг хэрэглээг тайлбарлах болно.

testfile.avi зэрэг файлыг тоглуулахын тулд `-vo` тохируулгаар төрөл бүрийн видео интерфэйсүүдийн аль нэгийг оруулж өгнө:

```
% mplayer -vo xv testfile.avi
```

```
% mplayer -vo sdl testfile.avi
```

```
% mplayer -vo x11 testfile.avi
```

```
# mplayer -vo dga testfile.avi
```

```
# mplayer -vo 'sdl:dga' testfile.avi
```

Эдгээр бүх тохируулгуудыг бүгдийг туршиж үзэх нь зүйтэй. Эдгээрийн харьцангуй ажиллагаа нь олон хүчин зүйлүүдээс шалтгаалдаг бөгөөд тоног төхөөрөмжөөс ихээхэн хамаарч өөр өөр байна.

DVD-ээс тоглуулахын тулд testfile.avi-г `dvd://N -dvd-device DEVICE` мөрөөр солих хэрэгтэй. Энд байгаа *N* нь тоглуулах гарчгийн дугаар бөгөөд DEVICE нь DVD-ROM-д зориулсан төхөөрөмжийн цэг юм. Жишээ нь /dev/dvd-ийн 3-р гарчгийг тоглуулахын тулд доор дурдсаныг ажиллуулна:

```
# mplayer -vo xv dvd://3 -dvd-device /dev/dvd
```

Анхдагч DVD төхөөрөмжийг MPlayer портыг `WITH_DVD_DEVICE` тохируулгаар бүтээж байх үед тодорхойлж болдог. Анхдагчаар энэ төхөөрөмж нь /dev/acd0 юм. Илүү дэлгэрэнгүй мэдээллийг портын Makefile файлаас олж болно.

Зогсоох, пауз авах, урагшлуулах гэх мэтийг хийхийн тулд товчлууруудын талаар үзэх хэрэгтэй бөгөөд ингэхийн тулд `mplayer -h` тушаалыг ажиллуулах юм уу эсвэл гарын авлагын хуудсыг унших нь зүйтэй юм.

Тоглуулахад хэрэгтэй чухал нэмэлт тохируулгууд нь: бүрэн дэлгэцийн горимд шилжүүлдэг `-fs -zoom` болон ажиллагаанд тусалдаг `-framedrop` тохируулгууд юм.

mplayer-ийн тушаалын мөрийг их урт болгохгүйн тулд .mplayer/config файл үүсгээд анхдагч тохируулгуудыг тэнд тохируулах хэрэгтэй:

```
vo=xv
fs=yes
zoom=yes
```

Төгсгөлд нь `mplayer` нь DVD гарчгийг .vob файл уруу гаргахад хэрэглэгдэж болдог. DVD-ээс хоёр дахь гарчгийг гаргаж авахын тулд үүнийг бичнэ:

```
# mplayer -dumpstream -dumpfile out.vob dvd://2 -dvd-device /dev/dvd
```

Гаралтын файл out.vob нь MPEG байх бөгөөд энэ хэсэгт тайлбарлагдсан бусад багцуудаар удирдагдаж болох юм.

#### 8.4.2.1.3. mencoder

`mencoder`-ийг ашиглахаасаа өмнө HTML баримтаас тохируулгуудтай танилцах нь зүйтэй юм. Гарын авлагын хуудас байдаг боловч HTML баримтгүйгээр энэ нь тийм ч тустай биш юм. Чанар, битийн бага хурдыг сайжруулах болон хэлбэршилтүүдийг өөрчлөх тоолж баршгүй аргууд байдаг бөгөөд эдгээр аргуудын зарим нь сайн, муу ажиллагааны хооронд ялгаа

гаргаж болох юм. Танд туслах хэд хэдэн жишээ энд байна. Эхлээд энгийн хуулбар:

```
% mencoder input.avi -oac copy -ovc copy -o output.avi
```

Тушаалын мөрийн буруу хослолууд нь `mplayer`-ээр ч тоглуулах боломжгүй файл гаргахад хүргэж болох юм. Тиймээс хэрэв та файл уруу гаргаж авахыг хүсвэл `mplayer`-ийн `-dumpfile` -ийг ашиглаарай.

input.avi-г MPEG4 кодек уруу MPEG3 аудио кодчилолтойгоор хөрвүүлэхийн тулд (audio/lame шаардлагатай):

```
% mencoder input.avi -oac mp3lame -lameopts br=192 \
     -ovc lavc -lavcopts vcodec=mpeg4:vhq -o output.avi
```

Энэ нь `mplayer` болон `xine` тоглуулах боломжтой гаралтыг үүсгэнэ.

DVD гарчгийг шууд дахин кодчилохын тулд input.avi файлын оронд `dvd://1 -dvd-device /dev/dvd` мөрийг тавьж `root` эрхээр ажиллуулах хэрэгтэй. Эхний удаадаа та өөрийн хийсэн зүйлдээ сэтгэл ханамжгүй байх магадлалтай болохоор гарчгийг файл уруу гаргаж файл дээрээ ажиллахыг зөвлөдөг.

#### 8.4.2.2. xine видео тоглуулагч

xine видео тоглуулагч нь аль ч талаасаа зөвхөн бүгдийг багтаасан видео шийдэл байх зорилготой төсөл биш бөгөөд энэ нь бас дахин ашиглагдах боломжтой үндсэн сан болон залгаасуудаар өргөтгөх боломжтой модульчлагдсан ажиллах файлуудыг үүсгэх зорилготой төсөл юм. Энэ нь багц болон multimedia/xine гэсэн порт хэлбэрээр байдаг.

xine тоглуулагч нь бас л бүрэн гүйцэд боловсроогүй байгаа билээ. Хэдий тийм ч гэсэн сайнаар эхэлж байгаа юм. Практикт xine нь хурдан видео карттай CPU юм уу аль эсвэл XVideo өргөтгөлийн дэмжлэгийг шаарддаг. GUI нь ашиглагдаж болохоор боловч болхи юм.

Үүнийг бичиж байх үед CSS кодчилол хийгдсэн DVD-үүдийг тоглуулах xine-тай цуг ирдэг оролтын модуль байгаагүй. Үүнд зориулан бүтээгдсэн модулиудтай гуравдагч талын бүтээлтүүд байдаг боловч эдгээрийн аль нь ч FreeBSD-ийн портын цуглуулгад байдаггүй.

MPlayer-тэй харьцуулах юм бол xine нь хэрэглэгчийн хувьд илүүг хийдэг боловч нөгөө талаараа хэрэглэгчид зарим нэг илүү нарийн тааруулах хяналтын боломжийг өгдөггүй юм. xine видео тоглуулагч нь XVideo интерфэйсүүд дээр хамгийн сайн ажилладаг.

Анхдагчаар xine тоглуулагч нь график хэрэглэгчийн интерфэйс эхлүүлэх болно. Дараа нь тухайн файлыг онгойлгохдоо цэсүүдийг ашиглаж болно:

```
% xine
```

Өөрөөр, GUI-гүйгээр дараах тушаал ашиглан файлыг нэн даруй тоглуулахаар ажиллуулж болно:

```
% xine -g -p mymovie.avi
```

#### 8.4.2.3. transcode буюу код хооронд хөрвүүлэгч хэрэгслүүд

transcode програм хангамж нь тоглуулагч биш, харин видео болон аудио файлуудыг дахин кодчилоход зориулагдсан хэрэгслүүдийн цуглуулга юм. stdin/stdout урсгалын интерфэйсүүдийн хамтаар тушаалын мөрийн хэрэгслүүдийг ашиглан transcode-оор видео файлуудыг нийлүүлэх, эвдэрсэн файлуудыг засах боломжтой болдог.

multimedia/transcode портыг бүтээж байх явцад олон тооны тохируулгуудыг зааж өгч болдог бөгөөд бид дараах тушаалын мөрийг transcode-ийг бүтээхэд ашиглахыг зөвлөдөг:

```
# make WITH_OPTIMIZED_CFLAGS=yes WITH_LIBA52=yes WITH_LAME=yes WITH_OGG=yes \
WITH_MJPEG=yes -DWITH_XVID=yes
```

Санал болгогдсон тохируулгууд ихэнх хэрэглэгчдийн хувьд хангалттай байх ёстой.

`transcode`-ийн багтаамжуудыг харуулахын тулд DivX файлыг PAL MPEG-1 файл (PAL VCD) уруу хэрхэн хөрвүүлэхийг харуулах нэг жишээг үзүүлье:

```
% transcode -i input.avi -V --export_prof vcd-pal -o output_vcd
% mplex -f 1 -o output_vcd.mpg output_vcd.m1v output_vcd.mpa
```

Гарах MPEG файл output_vcd.mpg-г MPlayer-ээр тоглуулах боломжтой байна. Та файлыг бүр Видео CD хийхийн тулд CD-R зөөвөрлөгч уруу шарж болох бөгөөд энэ тохиолдолд та multimedia/vcdimager болон sysutils/cdrdao програмуудыг суулгаж ашиглах хэрэгтэй болно.

`transcode`-д зориулсан гарын авлагын хуудас байдаг боловч та илүү мэдээлэл болон жишээнүүдийн талаар transcode wiki-с бас лавлах хэрэгтэй.

### 8.4.3. Нэмэлт унших зүйлс

FreeBSD-д зориулсан төрөл бүрийн видео програм хангамжийн багцууд хурдацтай хөгжиж байна. Ойрын ирээдүйд энд дурдсан асуудлуудын олонхи нь шийдэгдэх магадлал тун өндөр юм. Тэр болтол FreeBSD-ийн A/V боломжуудыг аль болох ихээр авахыг хүсэж байгаа хүмүүс цугтаа хэд хэдэн FAQ болон зааврууудаас уншсан зүйлсээрээ мэдлэгээ хуваалцаж өөр өөр цөөн хэдэн програмууд ашиглах хэрэгтэй юм. Энэхүү хэсэг нь тийм нэмэлт мэдээллүүдийг уншигчдад өгөх зорилгоор бичигдсэн юм.

MPlayer-ийн баримт нь техникийн хувьд маш сайн мэдээлэлтэй юм. UNIX®-ийн видеоны талаар өндөр түвшний туршлагатай болохыг хүссэн хэн бүхэн эдгээр баримтуудаас зөвлөгөө авах нь зүйтэй юм. MPlayer-ийн захидлын жагсаалт нь баримтыг уншаагүй хүмүүст тийм ч нөхөрсөг бус байдаг учраас тэдэнд алдааны мэдэгдлүүдийг илгээхээр төлөвлөж байгаа бол түр азнаад RTFM (read the fine manual эсвэл read the fucking manual).

xine HOWTO нь бүх тоглуулагчийн хувьд ерөнхий байдаг ажиллагааны сайжруулалтын

тухай бүлгийг агуулдаг.

Төгсгөлд нь уншигчдын туршиж болох зарим ирээдүйтэй програмууд байдаг:

- multimedia/avifile порт хэлбэрээр бас байдаг Avifile.
- multimedia/ogle порт хэлбэрээр бас байдаг Ogle.
- Xtheater
- DVD агуулгыг зохиоход зориулагдсан нээлттэй эх багц multimedia/dvdauthor.

# 8.5. ТВ картуудыг тохируулах

## 8.5.1. Танилцуулга

ТВ картууд нь өөрийн компьютер дээр цацаж байгаа юм уу эсвэл кабелийн ТВ үзэх боломжийг танд олгодог. Тэдгээрийн ихэнх нь нийлмэл видеог RCA юм уу эсвэл S-видео оролтоор хүлээн авдаг бөгөөд эдгээр картуудын зарим нь ФМ радио тааруулагчтай (tuner) ирдэг.

FreeBSD нь Brooktree Bt848/849/878/879 эсвэл Conexant CN-878/Fusion 878a Видео Барих (capture) бичил схемийг bktr(4) драйвертай цуг ашиглан PCI дээр суурилсан ТВ картуудын дэмжлэгийг хангадаг. Та дэмжигдсэн тааруулагчтай цуг ирсэн хавтанг бас шалгах хэрэгтэй. Дэмжигдсэн тааруулагчдын жагсаалтын талаар bktr(4) гарын авлагын хуудаснаас лавлана уу.

## 8.5.2. Драйвер нэмэх

Өөрийн картыг ашиглахын тулд bktr(4) драйверийг дуудах хэрэгтэй болох бөгөөд дараах мөрийг /boot/loader.conf файлд ингэж нэмэн үүнийг хийж болно:

```
bktr_load="YES"
```

Өөрөөр, ТВ картын дэмжлэгийг өөрийн цөмд статикаар эмхэтгэж болох бөгөөд энэ тохиолдолд өөрийн цөмийн тохиргоондоо дараах мөрүүдийг нэмнэ:

```
device   bktr
device  iicbus
device  iicbb
device  smbus
```

Картын бүрэлдэхүүнүүд I2C шугамаар хоорондоо холбогддог болохоор эдгээр нэмэлт төхөөрөмжийн драйверууд нь шаардлагатай байдаг. Дараа нь шинэ цөмөө бүтээж суулгах хэрэгтэй.

Дэмжлэг таны системд нэмэгдсэний дараа та өөрийн машиныг дахин ачаалах хэрэгтэй. Ачаалах процессийн үеэр таны ТВ карт доор дурдсантай төстэйгөөр гарч ирэн харагдах ёстой:

```
bktr0: <BrookTree 848A> mem 0xd7000000-0xd7000fff irq 10 at device 10.0 on pci0
iicbb0: <I2C bit-banging driver> on bti2c0
iicbus0: <Philips I2C bus> on iicbb0 master-only
iicbus1: <Philips I2C bus> on iicbb0 master-only
smbus0: <System Management Bus> on bti2c0
bktr0: Pinnacle/Miro TV, Philips SECAM tuner.
```

Мэдээж эдгээр мэдэгдлүүд нь таны тоног төхөөрөмжөөс хамаарч өөр байж болно. Гэхдээ та тааруулагч зөв олдсон эсэхийг шалгах ёстой; зарим нэг олдсон параметрүүдийг sysctl(8) MIB-үүд болон цөмийн тохиргооны файлын тохируулгуудаар дарж өөрчлөх боломжтой хэвээр байдаг. Жишээ нь хэрэв та тааруулагчийг Philips SECAM тааруулагч уруу болгохыг хүсвэл өөрийн цөмийн тохиргооны файлдаа дараах мөрийг нэмэх хэрэгтэй:

```
options OVERRIDE_TUNER=6
```

эсвэл та sysctl(8)-ийг шууд ашиглаж болно:

```
# sysctl hw.bt848.tuner=6
```

Байдаг тохируулгуудын талаар дэлгэрэнгүйг bktr(4) гарын авлагын хуудас болон /usr/src/sys/conf/NOTES файлаас үзнэ үү.

### 8.5.3. Ашигтай програмууд

Өөрийн ТВ картыг ашиглахын тулд та дараах програмуудын аль нэгийг суулгах хэрэгтэй:

- multimedia/fxtv нь цонхон-дээрх-ТВ болон дүрс/аудио/видео барих (capture) боломжуудыг олгодог.
- multimedia/xawtv нь бас ТВ програм бөгөөд fxtv-тэй адил боломжуудтай байдаг.
- misc/alevt нь Видеотекст/Телетекстийг декод хийн үзүүлдэг.
- audio/xmradio нь зарим нэг ТВ карттай цуг ирдэг ФМ радио тааруулагчийг ашиглах програм юм.
- audio/wmtune нь радио тааруулагчуудад зориулсан ширээний эвтэйхэн програм юм.

Илүү олон програмууд FreeBSD-ийн портын цуглуулгад байдаг.

### 8.5.4. Алдааг олж засварлах

Та өөрийн ТВ картандаа ямар нэг асуудал олох юм бол та эхлээд видео барих бичил схем ба тааруулагч нь bktr(4) драйвераар яг дэмжигдсэн эсэх болон та зөв тохиргооны сонголтуудыг дамжуулсан эсэхээ шалгах хэрэгтэй. Өөрийн ТВ картын тухай төрөл бүрийн асуултууд болон илүү дэмжлэгийн талаар та FreeBSD мультимедиа захидлын жагсаалт захидлын жагсаалтад хандаж түүний архивыг нь ашиглаж болох юм.

# 8.6. MythTV

MythTV нь нээлттэй эхийн PVR програм хангамжийн төсөл юм.

Энэ нь Linux®-н ертөнцөд олон хамаарлуудтай бөгөөд суулгахад хэцүү төвөгтэй програм гэдгээрээ алдартай. FreeBSD-н портын систем ихэнх процессуудыг хялбаршуулдаг боловч зарим нэг бүрэлдэхүүнүүдийг гараар тохируулах шаардлагатай байдаг. Энэ хэсэг MythTV-г суулгаж тохируулахад туслах зорилготой.

## 8.6.1. Тоног төхөөрөмж

MythTV нь енкодер болон тюнер зэрэг видео оролтын төхөөрөмжүүд рүү хандахын тулд V4L-г ашиглахаар хийгдсэн. Одоогийн байдлаар MythTV нь multimedia/webcamd-н дэмждэг USB DVB-S/C/T картуудтай илүү сайн ажилладаг. Яагаад гэвэл webcamd нь V4L хэрэглэгчийн талбарын програмтай байдаг. webcamd-н дэмждэг дурын DVB карт MythTV-тэй ажиллах боловч ажилладаг гэж бүртгэгдсэн картуудын жагсаалтыг эндээс олж болно. Дараах multimedia/pvr250 ба multimedia/pvrxxx багцуудад Hauppauge картуудын хувьд бас драйверууд байдаг боловч тэдгээр нь MythTV-н 0.23 хувилбараас дээшхи хувилбаруудтай ажилладаггүй стандарт бус драйверын интерфэйстэй байдаг.

HTPC дээр байгаа бүх DVB драйверуудын жагсаалт бий.

## 8.6.2. Хамаарлууд

Уян хатан, модулчлагдсан учир MythTV нь хэрэглэгчид өөр өөр машинууд дээр нүүр болон арын програмуудыг ажиллуулах боломжийг олгодог.

Нүүр талын програмын хувьд multimedia/mythtv-frontend болон X сервер шаардлагатай бөгөөд үүнийг x11/xorg-с олж болно. Ер нь бол нүүрэнд ажиллах компьютер бас XvMC болон хэрэв боломжтой бол LIRC-тэй нийцтэй алсын удирдлагыг дэмждэг видео карттай байвал маш сайн.

Ар талын програмын хувьд multimedia/mythtv болон MySQL™ өгөгдлийн сан шаардлагатай бөгөөд нэмэлтээр тюнер ба бичлэг хадгалах хадгалалтын төхөөрөмж байж болох юм. MySQL™ багц нь multimedia/mythtv-г суулгаж байх явцад хамаарал хэлбэрээр автоматаар суух ёстой.

## 8.6.3. MythTV-г суулгах

MythTV-г суулгахын тулд дараах алхмуудыг ашиглана. Эхлээд FreeBSD-н портын цуглуулгаас MythTV-г суулгана:

```
# cd /usr/ports/multimedia/mythtv
# make install
```

MythTV өгөгдлийн санг суулгана:

```
# mysql -uroot -p < /usr/local/shared/mythtv/database/mc.sql
```

Арын тохиргоог хийнэ:

```
# mythtv-setup
```

Арын програмыг ажиллуулна:

```
# echo 'mythbackend_enable="YES"' >> /etc/rc.conf
# service mythbackend start
```

# 8.7. Дүрс скан хийгчид

## 8.7.1. Танилцуулга

FreeBSD дээр дүрс скан хийгчдэд хандах боломжийг FreeBSD-ийн портын цуглуулгад байдаг SANE (Scanner Access Now Easy буюу Скан хийгчид хандах Хандалт Одоо Амархан гэгдэх) API олгодог. SANE нь FreeBSD-ийн зарим төхөөрөмжийн драйверууд ашиглан скан хийгчийн техник хангамжид ханддаг.

FreeBSD нь SCSI болон USB скан хийгчдийг дэмждэг. Ямар нэг тохиргоо хийхээсээ өмнө таны скан хийгч SANE-ээр дэмжигдсэн эсэхийг шалгаарай. SANE нь скан хийгчийн дэмжлэг болон түүний төлвийн тухай мэдээллээр таныг хангах дэмжигдсэн төхөөрөмжүүдийн жагсаалттай байдаг.

## 8.7.2. Цөмийн тохиргоо

Дээр дурдсанаар SCSI болон USB интерфэйсүүд нь дэмжигдсэн байдаг. Таны скан хийгчийн интерфэйсээс болоод өөр өөр төхөөрөмжийн драйверууд шаардагддаг.

### 8.7.2.1. USB интерфэйс

GENERIC цөм нь анхдагчаар USB скан хийгчдийг дэмжихэд шаардлагатай төхөөрөмжийн драйверуудыг агуулдаг. Цөмийг өөрөө тохируулан ашиглах бол дараах мөрүүд өөрийн цөмийн тохиргооны файлд байгаа эсэхийг шалгах хэрэгтэй:

```
device usb
device uhci
device ohci
device ehci
```

Зөв цөмөөр дахин ачаалсны дараа өөрийн USB скан хийгчийг залгана. Таны скан хийгчийг таньж байгааг үзүүлэх мөр системийн мэдэгдлийн буферт (dmesg(8)) гарч ирэх ёстой:

```
ugen0.2: <EPSON> at usbus0
```

Бидний ажиллуулж байгаа скан хийгч нь /dev/ugen0.2 гэсэн төхөөрөмжийн цэгийг ашиглаж байгааг эдгээр мэдэгдлүүд харуулж байна. Энэ жишээний хувьд EPSON Perfection® 1650 USB скан хийгчийг ашигласан.

#### 8.7.2.2. SCSI интерфэйс

Хэрэв таны скан хийгч SCSI интерфэйстэй ирсэн бол ямар SCSI хянагч хавтанг ашиглахаа мэдэх нь чухал юм. Ашиглагдах SCSI бичил схемээс хамаарааd та өөрийн цөмийн тохиргооны файлаа тааруулах хэрэгтэй болно. GENERIC цөм нь хамгийн түгээмэл SCSI хянагчуудыг дэмждэг. NOTES файлыг уншиж өөрийн цөмийн тохиргооны файлд зөв мөрийг нэмсэн эсэхээ шалгаарай. SCSI хувиргагч драйвераас гадна та өөрийн цөмийн тохиргооны файлдаа дараах мөрүүдтэй байх хэрэгтэй:

```
device scbus
device pass
```

Таны цөм зөв эмхэтгэгдэж суулгагдсаны дараа ачаалах үед та системийн мэдэгдлийн буферт төхөөрөмжийг харж чадаж байх ёстой:

```
pass2 at aic0 bus 0 target 2 lun 0
pass2: <AGFA SNAPSCAN 600 1.10> Fixed Scanner SCSI-2 device
pass2: 3.300MB/s transfers
```

Таны скан хийгч системийг ачаалах үед асаагүй байсан ч гэсэн гараар camcontrol(8) тушаалын тусламжтай SCSI шугамын хайлтыг хийж илрүүлэх боломжтой байдаг:

```
# camcontrol rescan all
Re-scan of bus 0 was successful
Re-scan of bus 1 was successful
Re-scan of bus 2 was successful
Re-scan of bus 3 was successful
```

Дараа нь скан хийгч SCSI төхөөрөмжүүдийн жагсаалтад гарч ирэх болно:

```
# camcontrol devlist
<IBM DDRS-34560 S97B>              at scbus0 target 5 lun 0 (pass0,da0)
<IBM DDRS-34560 S97B>              at scbus0 target 6 lun 0 (pass1,da1)
<AGFA SNAPSCAN 600 1.10>           at scbus1 target 2 lun 0 (pass3)
<PHILIPS CDD3610 CD-R/RW 1.00>     at scbus2 target 0 lun 0 (pass2,cd0)
```

SCSI төхөөрөмжүүдийн тухай дэлгэрэнгүй мэдээлэл scsi(4) болон camcontrol(8) гарын авлагын хуудаснуудад байдаг.

### 8.7.3. SANE тохиргоо

SANE систем нь хоёр хэсэгт хуваагддаг: эдгээр нь арын хэсэг (graphics/sane-backends) болон урд хэсэг (graphics/sane-frontends) юм. Арын хэсэг нь скан хийгчид өөрт нь хандах хандалтыг олгодог. SANE-ий дэмжигдсэн төхөөрөмжүүдийн жагсаалт таны дүрс скан хийгчийг ямар арын хэсэг дэмжихийг заадаг. Та өөрийн төхөөрөмжийг ашиглаж чаддаг байхын тулд өөрийн скан хийгчийн хувьд зөв арын хэсгийг тодорхойлох нь зайлшгүй шаардлагатай юм. Урд хэсэг нь график хайлт хийх интерфэйсийн боломжийг олгодог (xscanimage).

Эхний алхам нь graphics/sane-backends порт юм уу эсвэл багцыг суулгах явдал юм. Дараа нь `sane-find-scanner` тушаал ашиглан SANE системээр скан хийгчийн илрүүлэлтийг шалгана:

```
# sane-find-scanner -q
found SCSI scanner "AGFA SNAPSCAN 600 1.10" at /dev/pass3
```

Гаралт нь скан хийгчийн интерфэйсийн төрөл болон системд скан хийгчийг залгахад хэрэглэгдсэн төхөөрөмжийн цэгийг үзүүлэх болно. Үйлдвэрлэгч болон бүтээгдэхүүний загвар гарч ирэхгүй байж болох бөгөөд энэ нь тийм чухал биш юм.

Зарим USB скан хийгчид нь танаас firmware дуудаж ачаалахыг шаарддаг бөгөөд энэ нь арын хэсгийн гарын авлагын хуудсанд тайлбарлагдсан байгаа. Та sane-find-scanner(1) болон linprocfs(7) гарын авлагын хуудаснуудыг бас унших хэрэгтэй юм.

Одоо бид скан хийгчийг скан хийх урд хэсгээр танигдах эсэхийг нь шалгах хэрэгтэй. Анхдагчаар SANE арын хэсгүүд нь sane(1) гэгддэг тушаалын мөрийн хэрэгсэлтэй ирдэг. Энэ тушаал нь танд төхөөрөмжүүдийг жагсааж тушаалын мөрөөс дүрс эзэмшилтийг хийхийг зөвшөөрдөг. `-L` тохируулга нь скан хийгчийн төхөөрөмжүүдийг жагсаахад хэрэглэгддэг:

```
# scanimage -L
device `snapscan:/dev/pass3' is a AGFA SNAPSCAN 600 flatbed scanner
```

Эсвэл жишээ нь USB интерфэйс-д ашиглагдсан USB скан хийгчтэй бол:

```
# scanimage -L
device 'epson2:libusb:/dev/usb:/dev/ugen0.2' is a Epson GT-8200 flatbed scanner
```

Энэ гаралт нь FreeBSD 8.X системийн хувьд бөгөөд `'epson2:libusb:/dev/usb:/dev/ugen0.2'` нь бидэнд скан хийгчийн ашиглаж байгаа арын нэр (`epson2`) ба төхөөрөмжийн цэгийг (`/dev/ugen0.2`) олгож байна.

Гаралт байхгүй юм уу эсвэл скан хийгч танигдсангүй гэсэн мэдэгдэл нь sane(1) нь скан хийгчийг таньж чадаагүйг харуулна. Хэрэв ингэсэн бол та арын хэсгийн тохиргооны файлыг засварлаж ашиглагдах скан хийгч төхөөрөмжийг тодорхойлох хэрэгтэй. /usr/local/etc/sane.d/ сан бүх арын хэсгийн тохиргооны файлуудыг агуулдаг. Энэ танилтын асуудал зарим нэг

USB скан хийгчдийн хувьд тохиолддог.

Жишээ нь USB интерфэйс-д хэрэглэгдсэн USB скан хийгчтэй байхад FreeBSD 8.X дээр скан хийгч нь ямар ч асуудалгүйгээр олдож ажиллаж байгаа боловч FreeBSD-ийн өмнөх (uscanner(4) драйвер хэрэглэгдэж байгаа) хувилбарууд дээр `sane-find-scanner` тушаалыг ажиллуулахад дараах мэдээллийг бидэнд өгөх болно:

```
# sane-find-scanner -q
found USB scanner (UNKNOWN vendor and product) at device /dev/uscanner0
```

Скан хийгч нь зөв олдсон бөгөөд USB интерфэйсийг ашиглан /dev/uscanner0 төхөөрөмжийн цэгт залгагдсан байна. Одоо бид скан хийгч зөв танигдсан эсэхийг шалгаж болно:

```
# scanimage -L

No scanners were identified. If you were expecting something different,
check that the scanner is plugged in, turned on and detected by the
sane-find-scanner tool (if appropriate). Please read the documentation
which came with this software (README, FAQ, manpages).
```

Скан хийгч нь танигдаагүй болохоор бид /usr/local/etc/sane.d/epson2.conf файлыг засварлах хэрэгтэй болно. Ашиглагдсан скан хийгчийн загвар нь EPSON Perfection® 1650 байсан, тэгэхээр бид `epson2` арын хэсгийг скан хийгчийг ашиглахыг мэдлээ. Арын хэсгүүдийн тохиргооны файлууд дахь туслах тайлбаруудыг уншихаа мартуузай. Мөрийн өөрчлөлтүүд нь их энгийн: таны скан хийгчийн хувьд буруу интерфэйстэй бүх мөрүүдийг тайлбар болгоно (бидний хувьд, бидний скан хийгч USB интерфэйсийг ашигладаг болохоор `scsi` гэсэн үгээс эхэлсэн бүх мөрүүдийг тайлбар болгоно), дараа нь файлын төгсгөлд ашиглагдах интерфэйс болон төхөөрөмжийн цэгийг заасан мөрийг нэмнэ. Энэ тохиолдолд бид дараах мөрийг нэмнэ:

```
usb /dev/uscanner0
```

Илүү дэлгэрэнгүй мэдээлэл болон хэрэглээний зөв синтаксын талаар арын хэсгийн тохиргооны файлд байгаа тайлбарууд болон арын хэсгийн гарын авлагын хуудаснаас уншихаа мартуузай. Бид одоо скан хийгч танигдсан эсэхийг шалгаж болно:

```
# scanimage -L
device `epson:/dev/uscanner0' is a Epson GT-8200 flatbed scanner
```

Бидний USB скан хийгч танигдлаа. Үйлдвэрлэгч болон загвар нь скан

> хийгчийнхтэй таарахгүй байвал энэ нь тийм чухал биш юм. Бодолцож үзэх гол түлхүүр зүйл нь бидэнд зөв арын хэсгийн нэр болон зөв төхөөрөмжийн цэгийг өгдөг `epson:/dev/uscanner0` талбар юм.

`scanimage -L` тушаал скан хийгчийг харж чаддаг болсны дараа тохиргоо дуусна. Төхөөрөмж нь одоо скан хийхэд бэлэн болсон байна.

sane(1) нь дүрс эзэмшилтийг тушаалын мөрөөс хийхийг бидэнд зөвшөөрдөг боловч дүрс скан хийхдээ график хэрэглэгчийн интерфэйс ашиглахыг эрхэмлэх нь зүйтэй юм. SANE нь хялбар боловч үр ашигтай график интерфэйсийг санал болгодог: энэ нь xscanimage (graphics/sane-frontends) юм.

Xsane (graphics/xsane) нь өөр нэг алдартай график скан хийгч урд хэсэг юм. Энэ урд хэсэг нь төрөл бүрийн скан хийгч горим (фото хуулбар хийх, факс, гэх мэт), өнгөний засвар, бөөнөөр скан хийх гэх мэт дэвшилтэт боломжуудыг санал болгодог. Эдгээр програмууд нь GIMP залгаас болж ашиглагдах боломжтой байдаг.

### 8.7.4. Бусад хэрэглэгчдэд скан хийгчид хандах боломжийг өгөх

Өмнөх бүх үйлдлүүдийг `root` зөвшөөрлүүдээр хийсэн билээ. Гэхдээ өөр хэрэглэгчдэд скан хийгчид хандах боломжийг өгөх хэрэг танд байж болох юм. Скан хийгчийн ашигладаг төхөөрөмжийн цэгт хандахад унших болон бичих зөвшөөрлүүд хэрэглэгчид хэрэгтэй болно. Жишээ нь бидний USB скан хийгч `operator` бүлгийн эзэмшдэг /dev/usb/0.2.0 гэсэн төхөөрөмжийн жинхэнэ цэг рүү симболын холбоос хийгдсэн /dev/ugen0.2 төхөөрөмжийн цэгийг ашиглаж байна (/dev сангийн агуулгыг харахад үүнийг батлах болно). Симболын холбоос болон төхөөрөмжийн цэгийг `wheel` ба `operator` бүлэг тус тус эзэмшдэг. `joe` хэрэглэгчийг эдгээр бүлэгт нэмснээр түүнд скан хийгчийг ашиглах боломжийг бүрдүүлэх боловч аюулгүй байдлын шалтгаанаас болоод та хэрэглэгчийг аль ч бүлэгт ялангуяа `wheel` бүлэгт нэмэхээсээ өмнө дахин бодох хэрэгтэй. USB төхөөрөмжүүдийг ашиглахад зориулж тусгай бүлэг үүсгэж энэ бүлгийн гишүүдийг скан хийгчид хандах боломжтой болгох нь илүү сайн шийдэл болдог.

Тиймээс бид жишээ нь `usb` бүлгийг ашиглах болно. Эхний алхам нь энэ бүлгийг pw(8) тушаалын тусламжтай үүсгэх явдал юм:

```
# pw groupadd usb
```

Дараа нь /dev/ugen0.2 симболын холбоос хийж /dev/usb/0.2.0 төхөөрөмжийн цэгийг `usb` бүлгийн хувьд зөв, бичих зөвшөөрлүүдийн (`0660` эсвэл `0664`) тусламжтай хандах боломжтой болгох ёстой. Учир нь эдгээр файлын эзэмшигч нь (`root`) тэдгээрт зөвхөн бичиж чаддаг. /etc/devfs.rules файлд дараах мөрүүдийг нэмж энэ бүгдийг хийж болно:

```
[system=5]
add path ugen0.2 mode 0660 group usb
add path usb/0.2.0 mode 0666 group usb
```

Одоо скан хийгчид хандах хандалтыг зөвшөөрөхийн тулд хэрэглэгчдийг `usb` бүлэгт нэмэх л

үлдлээ:

```
# pw groupmod usb -m joe
```

Дэлгэрэнгүйг pw(8) гарын авлагын хуудаснаас үзнэ үү.

# Chapter 9. FreeBSD цөмийг тохируулах нь

## 9.1. Ерөнхий агуулга

Цөм нь FreeBSD үйлдлийн системийн гол зүрх юм. Энэ нь санах ойг удирдах, аюулгүй байдлын хяналтуудыг хийх, сүлжээнд холбогдох, диск уруу хандах зэрэг олон үйлдлүүдийг хариуцан хийдэг. FreeBSD улам илүү динамикаар тохируулагдах болсон боловч зарим тохиолдолд цөмийг дахин тохируулж хөрвүүлэх шаардлага гардаг.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Та магадгүй яагаад өөрт тохируулсан цөм бүтээх хэрэгтэй талаар.
- Цөмийн тохиргооны файлыг хэрхэн бичих эсвэл байгаа тохиргооны файлыг хэрхэн өөрчлөх талаар.
- Цөмийн тохиргооны файлыг хэрхэн ашиглаж шинэ цөм үүсгэж бүтээх талаар.
- Шинэ цөмийг хэрхэн суулгах талаар.
- Хэрэв юм буруугаар эргэвэл хэрхэн алдааг олох талаар.

Энэхүү бүлгийн жишээнүүд дэх тушаалууд нь амжилттай болохын тулд `root` эрхээр ажиллах ёстой.

## 9.2. Яагаад өөрчлөн тохируулсан цөм бүтээх хэрэгтэй вэ?

Уламжлалаар бол FreeBSD нь "monolithic" цөмтэй байсан байна. Энэ нь цөм тоотой хэдэн төхөөрөмжүүдийн жагсаалтыг дэмжсэн нэг том програм байсан гэсэн үг бөгөөд хэрэв та цөмийн ажиллагааг өөрчлөх бол шинэ цөм хөрвүүлж дараа нь компьютераа шинэ цөмөөр ачаалан эхлүүлэх шаардлагатай байсан билээ.

Өнөөдөр FreeBSD нь цөмийн ихэнх ажиллагаагаа шаардлагын дагуу динамикаар ачаалдаг ба цөмөөс буцааж буулгах боломж бүхий модулиудаар тусгаарлагдсан загвар уруу шилжиж байна. Энэ нь цөм шинэ тоног төхөөрөмжид (зөөврийн компьютер дэх PCMCIA картууд зэрэг) дасан зохицож түүнийг хурдан хүртээмжтэй болгох, эсвэл цөм анх хөрвүүлэгдэхдээ цөмд хэрэггүй байсан шинэ ажиллагааг цөмд бий болгох боломжийг бүрдүүлдэг. Үүнийг модульчлагдсан цөм хэмээдэг юм.

Тэгсэн ч гэсэн зарим статик цөмийн тохиргоог заавал хийх шаардлагатай. Зарим тохиолдолд ажиллагаа нь цөмтэй нягт холбоотой учраас динамикаар ачаалахаар хийх боломжгүй байдаг юм. Бас энэ нь энгийнээр бол тэр үйл ажиллагаанд зориулж динамикаар ачаалах модулийг бичих цаг хэнд ч олдоогүй байж болох юм.

Өөрчлөн тохируулсан цөм бүтээх нь BSD дэвшилтэт хэрэглэгчдийн хувьд хамгийн чухал ажиллагаануудын нэг юм. Энэ процесс нь цаг их зарцуулах боловч таны FreeBSD системд олон ашиг өгөх болно.Өргөн хүрээний тоног төхөөрөмжүүдийг дэмжих ёстой GENERIC цөмтэй харьцуулахад өөрчлөн тохируулсан цөм нь зөвхөн таны PC-ний тоног төхөөрөмжүүдийг дэмждэг. Энэ нь дараах хэд хэдэн ашигтай:

- Хурдан ачаалах хугацаа. Цөм нь таны систем дэх тоног төхөөрөмжүүдийг зөвхөн шалгах учраас системийг ачаалах хугацаа мэдэгдэхүйц багасдаг.
- Санах ойн ашиглалт багасна. Өөрчлөн тохируулсан цөм нь ашиглагдахгүй байгаа боломжууд болон төхөөрөмжийн драйверуудыг орхигдуулснаар ихэнхдээ GENERIC цөмөөс бага санах ойг ашигладаг. Цөмийн код нь бусад програмуудад санах ойг ашиглах боломжгүй болгож санах ойд үргэлж байрлаж байдаг учир энэ нь маш чухал юм. Ийм учраас өөрчилсөн цөм нь бага хэмжээний RAM-тай систем дээр ялангуяа ашигтай байдаг.
- Нэмэлт тоног төхөөрөмжийн дэмжлэг. Дууны картууд зэрэг GENERIC цөмд байхгүй төхөөрөмжүүдийн дэмжлэгийг нэмэх боломжийг танд өөрчлөн тохируулсан цөм олгоно.

## 9.3. Системийн тоног төхөөрөмж хайж олох нь

Цөмийн тохиргоо уруу орж үзээд алдахаасаа өмнө машиныхаа тоног төхөөрөмжийн бүртгэлийг олж авах нь ухаалаг явдал юм. FreeBSD нь үндсэн үйлдлийн систем биш тохиолдолд байгаа үйлдлийн системийн тохиргоог харан бүртгэлийн жагсаалтыг хялбархан үүсгэж болно. Жишээ нь Microsoft®-ийн Device Manager буюу төхөөрөмжийн менежер нь суулгагдсан төхөөрөмжүүдийн талаарх чухал мэдээллийг ихэвчлэн агуулдаг. Device Manager нь control panel буюу хяналтын самбарт байрладаг.

Microsoft® Windows®-ийн зарим хувилбаруудад System гэсэн дүрс байдаг бөгөөд энэ нь Device Manager уруу хандах боломжтой дэлгэцийг харуулдаг.

Хэрэв өөр үйлдлийн систем машин дээр байхгүй бол администратор энэ мэдээллийг өөрөө олох хэрэгтэй болно. Нэг арга нь dmesg(8) хэрэгсэл болон man(1) тушаалуудыг ашиглах явдал юм. FreeBSD дээр ихэнх төхөөрөмжийн драйверууд нь дэмжигдсэн тоног төхөөрөмжүүдийн жагсаалтыг харуулсан гарын авлагын хуудастай байдаг бөгөөд ачаалах үед шалгаж байх явцад олдсон тоног төхөөрөмжийг харуулдаг. Жишээ нь дараах мөрүүд нь psm драйвер хулгана олсон гэдгийг харуулж байна:

```
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: [ITHREAD]
psm0: model Generic PS/2 mouse, device ID 0
```

Энэ драйвер нь өөрчлөн тохируулах цөмийн тохиргооны файлд орсон байх эсвэл loader.conf(5) ашиглан ачаалагдсан байх хэрэгтэй болно.

Зарим тохиолдолд `dmesg`-ээс гарч байгаа өгөгдөл нь ачаалалтын шалгалт, илрүүлэлтийн гаралтыг биш зөвхөн системийн мэдэгдлүүдийг үзүүлдэг. Эдгээр тохиолдлуудад /var/run/dmesg.boot файлыг үзэн гаралтыг олж авч болно.

Тоног төхөөрөмжийг олох өөр нэг арга бол илүү дэлгэрэнгүй гаралтыг үзүүлдэг pciconf(8) хэрэгслийг ашиглах явдал юм. Жишээ нь:

```
ath0@pci0:3:0:0:        class=0x020000 card=0x058a1014 chip=0x1014168c rev=0x01
hdr=0x00
```

```
    vendor     = 'Atheros Communications Inc.'
    device     = 'AR5212 Atheros AR5212 802.11abg wireless'
    class      = network
    subclass   = ethernet
```

`pciconf -lv` ашиглан олж авсан энэ бяцхан мэдээлэл нь ath драйвер утасгүй Ethernet төхөөрөмжийг олсныг харуулж байна. `man ath` тушаалыг ашиглавал ath(4) гарын авлагын хуудсыг харуулах болно.

Ашигтай мэлээлэл олж авахын тулд man(1) уруу `-k` тугийг өгөн ашиглаж болно. Дээрхээс ингэж өгч болно:

```
# man -k Atheros
```

Тухайн нэг үг агуулсан гарын авлагын хуудсын жагсаалтыг олж авахын тулд:

```
ath(4)                  - Atheros IEEE 802.11 wireless network driver
ath_hal(4)              - Atheros Hardware Access Layer (HAL)
```

Тоног төхөөрөмжийн жагсаалтаар зэвсэглэснээр цөмийг өөрчлөн тохируулж бүтээх процесс нь арай хялбар болно.

## 9.4. Цөмийн драйверууд, дэд системүүд, болон модулиуд

Өөрчлөн тохируулсан цөмийг бүтээхийн өмнө тэгж хийх шалтгаанаа бодож үзэх хэрэгтэй. Хэрэв тусгайлсан тоног төхөөрөмжийн дэмжлэг хэрэгтэй байгаа бол тэр нь модуль хэлбэрээр аль хэдийн байж байж болох юм.

Цөмийн модулиуд нь /boot/kernel санд байх бөгөөд ажиллаж байгаа цөмд kldload(8)-г ашиглан динамикаар дуудаж болдог. Цөмийн бүх драйверуудын ихэнх нь тусгай модуль болон гарын авлагын хуудастай байдаг. Жишээ нь сүүлийн хэсэг ath гэсэн утасгүй Ethernet драйверийн талаар дурдсан байдаг. Энэ төхөөрөмж нь өөрийн гарын авлагадаа дараах мэдээллийг агуулсан байдаг:

```
Alternatively, to load the driver as a module at boot time, place the
     following line in man:loader.conf[5]:

           if_ath_load="YES"
```

Зааврын дагуу /boot/loader.conf файлд `if_ath_load="YES"` мөрийг нэмснээр энэ модулийг ачаалах үед динамикаар дуудах боломжийг идэвхжүүлнэ.

Гэхдээ зарим тохиолдолд холбоотой модуль байдаггүй. Энэ нь зарим нэг дэд системүүд болон маш чухал драйверуудын хувьд бодит бөгөөд жишээ нь fast file system (FFS) буюу

түргэн файлын систем нь цөмд заавал байх шаардлагатай тохируулга юм. Мөн сүлжээний дэмжлэгийн (INET) хувьд ийм байна. Харамсалтай нь драйвер шаардлагатай эсэхийг хэлэх цорын ганц зам нь модулийг нь шалгах явдал юм.

Төхөөрөмж эсвэл тохируулгын дэмжлэгийг хялбараар устгаж эвдэрхий цөмтэй үлдэж болох юм. Жишээ нь ata(4) драйверийг цөмийн тохиргооны файлаас устгачих юм бол ATA дискийн хөтөчүүд нь loader.conf-д модулийг нэмэлгүйгээр эхлэн ачаалахгүй байж болох юм. Хэрэв эргэлзэж байгаа бол модулийг шалгаад дараа нь ердөө л дэмжлэгийг цөмд үлдээх хэрэгтэй.

# 9.5. Өөрчлөн тохируулсан цөмийг бүтээх ба суулгах нь

Цөмийг бүтээхэд FreeBSD-ийн гүйцэд эх мод байх шаардлагатай.

Эхлээд цөм бүтээх сангаар аялая. Дурдсан бүх сангуудаас гол нь /usr/src/sys сан байх бөгөөд /sys гэсэн замаар бас хандах боломжтой. Энд байгаа хэд хэдэн дэд сангууд цөмийн өөр өөр хэсгүүдийг илэрхийлэх бөгөөд бидний зорилгод хамгийн чухал нь таны өөрчлөн тохируулах цөмийн тохиргоог засварлах arch/conf сангууд болон таны цөм бүтээгдэх шатны талбар compile сан юм. *arch* нь i386, amd64, ia64, powerpc, sparc64, эсвэл pc98 (Японд их ашиглагддаг PC тоног төхөөрөмжийн өөр нэг хөгжүүлэлтийн салбар) зэргийг төлөөлдөг. Тухайн архитектурын сан доторх код зөвхөн тэр архитектуртай холбоотой; бусад кодын хэсэг нь FreeBSD порт хийгдэх боломж бүхий бүх тавцангуудын хувьд адил машинаас чөлөөт код байна. Сангийн бүтцийн логик зохион байгуулалт нь дэмжлэг хийгдсэн төхөөрөмж, файлын систем болон өөрийн дэд санд байгаа тохируулга бүртэй хамт байгааг харж болно.

Энэ бүлэгт байгаа жишээ таныг i386 архитектур ашиглаж байгаа гэж авч үзнэ. Хэрэв таны систем өөр архитектур бол замуудын нэрсдээ тохирох өөрчлөлтүүдийг хийгээрэй.

Хэрэв /usr/src/ сан таны систем дээр байхгүй (эсвэл хоосон бол) эх суугаагүй байна. Бүрэн гүйцэд эх модыг суулгах хялбар аргаар хийхийн тулд Өөрийн эхийг хамгийн сүүлийн хэлбэрт аваачих нь хэсэгт тайлбарласны дагуу csup(1)-г ашиглах явдал юм. Та бас /usr/src/sys/ руу заасан симболын холбоос үүсгэж өгөх ёстой:

```
# ln -s /usr/src/sys /sys
```

Дараа нь arch/conf сан уруу шилжээд GENERIC тохиргооны файлыг та өөрийн цөмдөө өгөх нэр уруу хуул. Жишээ нь:

```
# cd /usr/src/sys/i386/conf
# cp GENERIC MYKERNEL
```

Уламжлалаар бол энэ нэр нь бүгд том үсгээр байдаг, хэрэв та олон өөр өөр төрлийн FreeBSD

машинуудын ажиллагааг хянадаг бол машинуудынхаа нэрээр нэрлэх нь зохимжтой юм. Бид энэ жишээнийхээ зорилгоор MYKERNEL гэж нэрлэе.

> Өөрийн цөмийн тохиргооны файлаа шууд /usr/src доор хадгалах нь буруу байж болох юм. Хэрэв та асуудлуудтай тулгарч байгаа бол /usr/src -ийг устгаад л дахиж эхлэх нь зоригтой алхам байж болох юм. Гэхдээ үүнийг хийгээд хэдэн секундын дараа л та өөрийн өөрчлөн тохируулсан цөмийн тохиргооны файлаа устгасан болохоо мэдэх болно. Мөн GENERIC файлыг шууд засварлах хэрэггүй бөгөөд дараагийн удаа өөрийн эх модыг шинэчлэх үйлдлийг хийхэд дарагдан хуулагдаж таны цөмийн өөрчлөлт алдагдаж магадгүй.
>
> Та цөмийн тохиргооны файлаа өөр газар хадгалж дараа нь i386 сан дахь файл уруу тэмдэгт холбоос үүсгэж болно.
>
> Жишээ нь:
>
> ```
> # cd /usr/src/sys/i386/conf
> # mkdir /root/kernels
> # cp GENERIC /root/kernels/MYKERNEL
> # ln -s /root/kernels/MYKERNEL
> ```

Одоо MYKERNEL-ийг өөрийн дуртай текст засварлагч дээр засаарай. Хэрэв та дөнгөж эхэлж байгаа бол байгаа цорын ганц засварлагч нь vi байж болох бөгөөд түүнийг энд тайлбарлахад хэтэрхий төвөгтэй боловч номын жагсаалтад байгаа өөр олон номнуудад бичсэн байгаа. Гэхдээ FreeBSD нь ее гэдэг хялбар засварлагчийг санал болгодог бөгөөд хэрэв та эхлэн сурагч бол энэ нь таны сонгох засварлагч байх болно. Өөрийн тохиргоог тусгах эсвэл GENERIC файлаас өөрийн хийсэн өөрчлөлтүүдээс ялгахын тулд дээд хэсэгт байгаа мөрүүдийг чөлөөтэй өөрчлөөрэй.

Хэрэв та SunOS™ эсвэл өөр BSD үйлдлийн системийн доор цөм бүтээж байсан бол энэ файлын ихэнх хэсэг нь маш танил байх болно. Хэрэв та DOS зэрэг өөр үйлдлийн системээс ирж байгаа бол нөгөө талаасаа GENERIC тохиргооны файл төвөгтэй юм шиг санагдаж болох бөгөөд Тохиргооны файл хэсгийн тайлбаруудыг удаан нухацтай дагаарай.

> Хэрэв та FreeBSD төслийн хамгийн сүүлийн эхээр өөрийн эх модоо сүүлийн үеийн хэлбэрт авчирсан бол шинэчлэх шатуудаа хэрэгжүүлж эхлэхээсээ өмнө /usr/src/UPDATING файлыг үргэлж шалгаж байх нь чухал юм. Энэ файл нь шинэчилсэн эх код доторх тусгай анхаарал шаардлагатай чухал асуудлууд эсвэл хэсгүүдийн талаар тайлбарладаг. /usr/src/UPDATING нь үргэлж таны FreeBSD хувилбартай таардаг бөгөөд энэ гарын авлагаас илүү шинэ мэдээлэлтэй, сүүлийн үеийнх байдаг.

Та цөмд зориулан эх кодоо хөрвүүлэх шаардлагатай.

**Procedure: Цөмийг бүтээх нь**

Цөмийг бүтээхэд FreeBSD-ийн гүйцэд эх мод байх шаардлагатай.

1. /usr/src сан уруу орно:

```
# cd /usr/src
```

2. Цөмийг хөрвүүлнэ:

```
# make buildkernel KERNCONF=MYKERNEL
```

3. Шинэ цөмийг суулгана:

```
# make installkernel KERNCONF=MYKERNEL
```

Анхдагчаар өөрчлөн тохируулсан цөмийг бүтээхэд *бүх* цөмийн модулиуд бас бүтээгдэнэ. Хэрэв та цөмийг хурдан шинэчлэхийг эсвэл зөвхөн өөрчлөн тохируулсан модулиудыг бүтээхийг хүсэж байгаа бол цөмийг бүтээж эхлэхээсээ өмнө /etc/make.conf файлыг засварлах хэрэгтэй:

```
MODULES_OVERRIDE = linux acpi sound/sound sound/driver/ds1 ntfs
```

Энэ хувьсагч нь бүгдийг биш бүтээх модулиудын жагсаалтыг тодорхойлдог.

```
WITHOUT_MODULES = linux acpi sound ntfs
```

Энэ хувьсагч нь бүтээх процессод оруулахгүй байх дээд түвшний модулиудын жагсаалтыг тодорхойлдог. Цөмийг бүтээх процессийн явцад танд хэрэг болохуйц бусад хувьсагчуудын тухайд make.conf(5) гарын авлагын хуудсанд хандаж үзнэ үү.

Шинэ цөм /boot/kernel санд /boot/kernel/kernel нэрээр хуулагдах бөгөөд хуучин цөм нь /boot/kernel.old/kernel уруу хуулагдана. Одоо системийг унтраагаад шинэ цөмийг ашиглан дахин ачаал. Хэрэв ямар нэг юм болохгүй болбол энэ бүлгийн төгсгөлд байгаа алдааг олж засварлах заавар танд хэрэгтэй байж болох юм. Таны шинэ цөм ачаалахгүй тохиолдолд хэрхэн сэргээх талаар тайлбарласан хэсгийг заавал уншаарай.

Ачаалах loader(8) ба тохиргоо зэрэг ачаалах процесстой холбоотой бусад файлууд /boot -д хадгалагдана. Гуравдагч этгээдийн эсвэл өөрчлөн тохируулсан модулиуд /boot/kernel-д байрлах бөгөөд гэхдээ модулиудыг хөрвүүлсэн цөмийн адил сүүлийн үеийн хэлбэрт байлгах нь маш чухал гэдгийг хэрэглэгчид мэдэх шаардлагатай. Хөрвүүлсэн цөмтэй хамт ажиллуулахааргүй модулиуд нь тогтворгүй байдал эсвэл буруу ажиллагаанд

хүргэж болзошгүй юм.

## 9.6. Тохиргооны файл

Тохиргооны файлын ерөнхий хэлбэр нь маш энгийн билээ. Мөр болгон түлхүүр үг бөгөөд нэг болон хэд хэдэн нэмэлт өгөгдлөөс тогтоно. Амархан болгох үүднээс ихэнх мөрүүд нь зөвхөн нэг нэмэлт өгөгдөлтэй байна. `#` тэмдэгтийн ард байгаа зүйлс тайлбар бөгөөд хаягдаж тооцогдоно. Дараах хэсэгт түлхүүр үг болгоныг GENERIC -д жагсаасан дарааллаар нь тайлбарлаж байна.  Архитектураас хамааралтай тохируулгууд болон төхөөрөмжийн ядраамаар жагсаалтын талаар GENERIC файл байгаа сангийн нэгэн адил санд байрлах NOTES файлаас үзнэ үү. Архитектураас хамааралгүй тохируулгуудын талаар /usr/src/sys/conf/NOTES файлаас үзнэ үү.

Тохиргооны файлуудад ашиглах боломжтой шинэ `include` тохируулга бий болсон. Энэ нь өөр нэг тохиргооны файлыг тухайн тохиргооны файлд оруулах боломжийг бүрдүүлэх бөгөөд ингэснээр тухайн файлын хувьд харьцангуй бага өөрчлөлтүүдийг арчлах боломжтой болгодог. Жишээ нь хэрэв танд цөөн тооны нэмэлт тохируулга эсвэл драйверуудтай GENERIC цөм шаардлагатай бол энэ нь GENERIC-ийн хувьд цөөн өөрчлөлтийг арчлах боломжийг танд олгоно:

```
include GENERIC
ident MYKERNEL

options         IPFIREWALL
options         DUMMYNET
options         IPFIREWALL_DEFAULT_TO_ACCEPT
options         IPDIVERT
```

Энэ загвар нь тохиргооны файлуудыг эхнээс нь бичих уламжлалт аргын хажууд хамаагүй илүү боломжийг олгодог гэдэгтэй ихэнх администраторууд санал нэг байдаг: локал тохиргооны файл нь зөвхөн GENERIC цөмөөс ялгаатай локал өөрчлөлтүүдийг харуулах бөгөөд шинэчлэлт хийгдэхэд GENERIC-д нэмэгдсэн шинэ боломжууд нь `nooptions` эсвэл `nodevice` тохируулгуудаар тусгайлан заагдаагүй л бол локал цөмд нэмэгддэг. Энэ бүлгийн үлдсэн хэсэг ердийн тохиргооны файлын агуулга ба төрөл бүрийн тохируулгын үүрэг болон ажиллах төхөөрөмжүүдийг тайлбарлах болно.

Тест хийх зорилгоор ихэнхдээ бүх байгаа тохируулгууд агуулсан файлыг бүтээхдээ дараах тушаалыг `root` эрхээр ажиллуулна:

```
# cd /usr/src/sys/i386/conf && make LINT
```

Дараах жишээ нь шаардлагатай бол тодотгох зорилгоор оруулсан төрөл бүрийн нэмэлт тайлбар бүхий GENERIC цөмийн тохиргооны файл юм. Энэ жишээ нь таны /usr/src/sys/i386/conf/GENERIC дахь хуулбартай их ойрхон таарах ёстой.

```
machine     i386
```

Энэ нь машины архитектур юм. Энэ нь `amd64`, `i386`, `ia64`, `pc98`, `powerpc`, эсвэл `sparc64` -ийн аль нэг байх ёстой.

```
cpu          I486_CPU
cpu          I586_CPU
cpu          I686_CPU
```

Дараах тохируулга нь таны системд байгаа CPU-ийн төрлийг заана. Та олон CPU мөртэй байж болох боловч (хэрэв, жишээ нь та `I586_CPU` эсвэл `I686_CPU` хоёрын алийг ашиглахаа сайн мэдэхгүй байгаа бол) өөрчлөн тохируулсан цөмийн хувьд зөвхөн байгаа CPU-гээ заах нь зүйтэй юм. Хэрэв та өөрийн CPU-ийн төрлийг сайн мэдэхгүй байгаа бол /var/run/dmesg.boot файлыг шалгаж ачаалах үеийн мэдээллүүдийг үзэж болно.

```
ident          GENERIC
```

Энэ нь цөмийг тодорхойлох нэр юм. Хэрэв та түрүүний жишээнүүдэд дурдсан зааврыг дагасан бол өөрийн цөмийг нэрлэсэн шигээ өөрөөр хэлбэл `MYKERNEL` хэмээн өөрчлөх хэрэгтэй. `ident` мөрд оруулсан утга нь таныг цөмийг ачаалах үед хэвлэгдэн гарах учир та өөрийн ердийн цөмөөс шинэ цөмөө тусад нь хадгалахыг хүсвэл шинэ цөмдөө өөр нэр өгөх нь ашигтай байдаг (өөрөөр хэлбэл та туршилтын цөм бүтээхийг хүсвэл).

```
#To statically compile in device wiring instead of /boot/device.hints
#hints          "GENERIC.hints"         # Default places to look for devices.
```

device.hints(5) нь төхөөрөмжүүдийн драйверуудын тохируулгуудыг хийхэд ашиглагдана. loader(8)-ийн ачаалах үе шалгах анхдагч байрлал нь /boot/device.hints байна. `hints` тохируулгыг ашиглаад та эдгээр зөвлөгөөнүүдийг статикаар хөрвүүлж болно. Тэгэхэд /boot дотор device.hints файл үүсгэх шаардлагагүй болох юм.

```
makeoptions     DEBUG=-g          # Build kernel with gdb(1) debug symbols
```

FreeBSD -г бүтээх ердийн процесс нь дибаг (debug) хийх мэдээллийг агуулдаг бөгөөд цөмийг бүтээх үед `-g` тохируулгыг gcc(1) уруу өгснөөр дибаг (debug) хийх мэдээлэл идэвхждэг.

```
options          SCHED_ULE          # ULE scheduler
```

FreeBSD -ийн анхдагч системийн төлөвлөгч/хуваарилагч. Үүнийг үлдээ.

```
options          PREEMPTION         # Enable kernel thread preemption
```

Цөм дэх урсгалуудыг (thread) өөр илүү өндөр давуу эрхтэй урсгалуудаар солих боломжийг бүрдүүлнэ. Энэ нь харилцан ажиллах болон таслах урсгалуудыг (interrupt threads) хүлээлгэлгүйгээр аль болох түргэн ажиллуулахад тусалдаг.

```
options         INET            # InterNETworking
```

Сүлжээний дэмжлэг. Сүлжээнд холбогдохгүй ч гэсэн энэ тохиргоог үлдээгээрэй. Ихэнх програмууд эргэн холбогдох (loopback буюу өөрөөр хэлбэл өөрийн PC дотор сүлжээний холболт хийх) сүлжээг шаарддаг учир энэ нь үндсэндээ зайлшгүй шаардлагатай.

```
options         INET6           # IPv6 communications protocols
```

Энэ нь IPv6 холбооны протоколуудыг идэвхжүүлдэг.

```
options         FFS             # Berkeley Fast Filesystem
```

Энэ нь энгийн хатуу дискний файлын систем. Энэ тохируулгыг хатуу дискнээс ачаалах бол үлдээгээрэй.

```
options         SOFTUPDATES     # Enable FFS Soft Updates support
```

Энэ тохируулга нь Зөөлөн Шинэчлэлүүдийг цөм идэвхжүүлдэг бөгөөд диск уруу бичих хандалтыг хурдасгахад тусалдаг. Хэдийгээр энэ боломжийг цөмөөр хангадаг боловч диск дээр идэвхжүүлэх шаардлагатай. Таны системийн дискнүүд дээр Зөөлөн Шинэчлэлүүд идэвхжсэн эсэхийг mount(8) -ийн үр дүнгээр хянаарай. Хэрэв та `soft-updates` тохируулгыг олж харахгүй байгаа бол tunefs(8) (одоо байгаа системийн хувьд) эсвэл newfs(8) (шинэ файлын системийн хувьд) ашиглан идэвхжүүлэх хэрэгтэй.

```
options         UFS_ACL         # Support for access control lists
```

Энэ тохируулга нь хандалтыг хянах жагсаалтын дэмжлэгийг цөмд идэвхжүүлдэг. Энэ нь өргөтгөсөн шинж чанарууд ба UFS2-ийг ашиглахад тулгуурлаж байгаа бөгөөд энэ боломжийн талаар Файлын системийн хандалт хянах жагсаалтууд(ACL-үүд) дээр дэлгэрэнгүй тайлбарласан байдаг. ACL-үүд эхэндээ идэвхжүүлсэн байдаг бөгөөд хэрэв урьд нь файлын систем дээр ашиглагдаж байсан бол хандалтыг хянах жагсаалтыг устгаж файлуудыг хамгаалсан аргыг урьдчилан тааж болшгүй байдалт хүргэдэг учир энэ тохируулгыг хааж болохгүй.

```
options         UFS_DIRHASH     # Improve performance on big directories
```

Энэ тохируулга нь нэмэгдэл санах ойг зарлагадаж том сангуудад хийх дискний үйлдлүүдийг хурдасгах ажиллагааг оруулдаг. Та том сервер эсвэл харилцан ажиллах ажлын станцад

зориулж энэ тохируулгыг ерөнхийдөө хадгалах хэрэгтэй бөгөөд хэрэв та FreeBSD-г санах ой чухал жижиг систем ба дискний хандалтын хурдны ач холбогдол багатай галт хана мэтийн систем дээр ашиглаж байгаа бол устгаарай.

```
options         MD_ROOT         # MD is a potential root device
```

Энэ тохируулга нь санах ой дээр тулгуурласан, root төхөөрөмж болж ашиглагдах виртуал дискний дэмжлэгийг идэвхжүүлдэг.

```
options         NFSCLIENT       # Network Filesystem Client
options         NFSSERVER       # Network Filesystem Server
options         NFS_ROOT        # NFS usable as /, requires NFSCLIENT
```

Сүлжээний файлын систем. Хэрэв та TCP/IP-аар UNIX® файлын серверээс хуваалтыг холболт хийх төлөвлөгөөгүй бол эдгээрийг тайлбар болгон хааж болно.

```
options         MSDOSFS         # MSDOS Filesystem
```

MS-DOS® файлын систем. Хэрэв та DOS хэлбэржүүлсэн хатуу дискний хуваалтыг ачаалах үед холболт хийх төлөвлөгөөгүй бол үүнийг айлгүйгээр тайлбар болгон хааж болно. Энэ нь дээр тайлбарласны дагуу эхний удаа DOS хуваалтыг холболт хийхэд автоматаар ачаалагдах болно. Мөн маш сайн emulators/mtools програм хангамж нь холболт болон салгалт хийлгүйгээр DOS уян дискнүүдэд хандах боломжийг бүрдүүлдэг (энэ нь MSDOSFS-ийг огт шаарддаггүй).

```
options         CD9660          # ISO 9660 Filesystem
```

CDROM-уудад зориулсан ISO 9660 файлын систем. Хэрэв та CDROM хөтлөгч байхгүй эсвэл CD-ээс өгөгдлийг хааяа холболт хийдэг бол (таныг анх өгөгдлийн CD-г холболт хийх үед динамикаар ачаалагддаг учраас) тайлбар болгож хааж болно. Дууны CD-үүд энэ файлын системийг хэрэглэдэггүй.

```
options         PROCFS          # Process filesystem (requires PSEUDOFS)
```

Процессийн файлын систем. Энэ нь ps(1) шиг програмууд процессуудын ажиллаж байгаа талаар дэлгэрэнгүй мэдээлэл танд өгөх боломжийг бүрдүүлдэг /proc дээр холболт хийгдсэн "хуурамч" файлын систем юм. Ихэнх дибаг хийх ба монитор хийх хэрэгслүүд PROCFS -гүйгээр ажиллахаар хийгдсэн байдаг: суулгалтууд нь энэ файлын системийг анхдагчаар холболт хийхгүй, тиймээс PROCFS-ийг ашиглах нь ихэнх тохиолдолд шаардлагагүй байдаг.

```
options         PSEUDOFS        # Pseudo-filesystem framework
```

Цөмүүд PROCFS ашиглаж байгаа бол бас PSEUDOFS дэмжлэгийг оруулах шаардлагатай.

```
options         GEOM_PART_GPT          # GUID Partition Tables.
```

GUID Хуваалтын Хүснэгтүүдийн дэмжлэгийг нэмдэг. GPT нь диск бүрийн хувьд их олон тооны хуваалт байх боломжийг бүрдүүлдэг бөгөөд стандарт тохиргоогоор 128 байдаг.

```
options         COMPAT_43         # Compatible with BSD 4.3 [KEEP THIS!]
```

4.3BSD-тэй нийцтэй байх. Үүнийг энэ чигээр орхи; хэрэв үүнийг тайлбар болгож хаавал зарим програмууд сонин ажиллаж эхэлнэ.

```
options         COMPAT_FREEBSD4   # Compatible with FreeBSD4
```

Энэ тохируулга нь FreeBSD-ийн хуучин хувилбарууд дээр хөрвүүлэгдсэн, хуучин системийн дуудлагуудын интерфэйсүүдийг ашигладаг програмуудыг дэмжихэд шаардлагатай. Энэ тохируулгыг хуучин програмууд ашиглаж болзошгүй бүх i386™ системүүд дээр ашиглахыг зөвлөж байна; ia64 ба sparc64 зэрэг 5.X дээр зөвхөн дэмжлэг хийгдсэн тавцангууд энэ тохируулгыг шаарддаггүй.

```
options         COMPAT_FREEBSD5   # Compatible with FreeBSD5
```

Энэ тохируулга нь FreeBSD 5.X системийн дуудлагын интерфэйсүүдийг ашигладаг, FreeBSD 5.X хувилбарууд дээр эмхэтгэгдсэн програмуудыг дэмжихэд шаардлагатай.

```
options         COMPAT_FREEBSD6   # Compatible with FreeBSD6
```

Энэ тохируулга нь FreeBSD 6.X системийн дуудлагын интерфэйсүүдийг ашигладаг, FreeBSD 6.X хувилбарууд дээр эмхэтгэгдсэн програмуудыг дэмжихэд шаардлагатай.

```
options         COMPAT_FREEBSD7   # Compatible with FreeBSD7
```

Энэ тохируулга нь FreeBSD 8 болон түүнээс хойшхи хувилбарууд дээр FreeBSD 7.X системийн дуудлагын интерфэйсүүдийг ашигладаг, FreeBSD 7.X хувилбарууд дээр эмхэтгэгдсэн програмуудыг дэмжихэд шаардлагатай.

```
options         SCSI_DELAY=5000  # Delay (in ms) before probing SCSI
```

Энэ нь цөмийг SCSI төхөөрөмжүүдийг шалгаж эхлэхээс өмнө 5 секунд түр зогсооно. Хэрэв та зөвхөн IDE хатуу дисктэй бол үүнийг орхиж болно, эсрэг тохиолдолд ачаалалтыг хурдасгахын тулд энэ тоог багасгаж болно. Мэдээж та үүнийг хийгээд FreeBSD таны SCSI төхөөрөмжүүдийг танихгүй бол та дахин үүнийг ихэсгэж болно.

```
options         KTRACE          # ktrace(1) support
```

Энэ нь дибаг хийхэд ашигтай, цөмийн процессийг дагах боломжийг идэвхжүүлдэг.

```
options         SYSVSHM         # SYSV-style shared memory
```

Энэ тохируулга нь System V хуваалцсан санах ойн боломж бүрдүүлдэг. Энэ боломжийн өргөн хэрэглээнүүдийн нэг нь X дэх XSHM өргөтгөл бөгөөд үүнийг график их шаарддаг олон програмууд автоматаар илүү хурд авахын тулд ашигладаг. Хэрэв та X ашигладаг бол үүнийг заавал оруулахыг хүсэх болно.

```
options         SYSVMSG         # SYSV-style message queues
```

System V мэдээллүүдийн дэмжлэг. Энэ тохируулга нь зөвхөн хэдхэн зуун байтыг цөмд нэмдэг.

```
options         SYSVSEM         # SYSV-style semaphores
```

System V семафорын дэмжлэг. Нэг их өргөн ашиглагддаггүй боловч хэдхэн зуун байтыг цөмд нэмдэг.

ipcs(1) тушаалын -p тохируулга нь эдгээр System V боломж бүрийг ашигласан процессуудыг жагсаадаг.

```
options         _KPOSIX_PRIORITY_SCHEDULING # POSIX P1003_1B real-time extensions
```

Жинхэнэ-хугацааны (real-time) өргөтгөлүүд 1993 оны POSIX®-д нэмэгдсэн.Портуудын цуглуулгаас зарим програмууд эдгээрийг ашигладаг (StarOffice™).

```
options         KBD_INSTALL_CDEV  # install a CDEV entry in /dev
```

Энэ тохируулга нь гарын төхөөрөмжийн цэгийг /dev-д үүсгэх боломжийг олгоход шаардлагатай байдаг.

```
options         ADAPTIVE_GIANT    # Giant mutex is adaptive.
```

Giant гэдэг нь харилцан нэгийгээ оруулахгүй байх зарчмын (унтах мутекс) нэр бөгөөд энэ нь цөмийн их эх үүсвэрүүдийг хамгаалдаг. Өнөөдөр энэ нь үйл ажиллагааны саатаж байгаа хүлээн авах боломжгүй хэсэг бөгөөд үүнийг эх үүсвэр бүрийг хамгаалах цоожуудаар идэвхтэйгээр сольж байгаа билээ. ADAPTIVE_GIANT тохируулга нь хэсэг бүлэг мутексуудад Giant-ийг адаптиваар эргэхээр оруулдаг. Энэ нь урсгал (thread) Giant мутексийг цоожлохыг хүсэж байх үед, гэхдээ энэ нь өөр CPU дээр урсгалаар цоожлогдсон байна, эхний урсгал цоож

сулрахыг хүлээн ажилласаар байна. Хэвийн үед урсгал унтаа байдалд эргэж орох бөгөөд өөрийн дараагийн ажиллах боломжийг хүлээнэ. Та итгэлгүй байгаа бол үүнийг орхино уу.

FreeBSD 8.0-RELEASE болон түүнээс хожуу үеийн хувилбаруудад бүх мутексууд нь `NO_ADAPTIVE_MUTEXES` тохируулгатайгаар бүтээгдэж зохицох чадваргүй гэж тохируулагдаагүй л бол анхдагчаар адаптив буюу зохицох чадвартай байдаг. Иймээс одоо Giant нь зохицох чадвартай бөгөөд `ADAPTIVE_GIANT` тохируулга нь цөмийн тохиргооноос хасагдсан юм.

```
device          apic            # I/O APIC
```

apic төхөөрөмж нь I/O APIC-ийг тасалдал хүргэхэд ашиглах боломжийг нээдэг. apic төхөөрөмж нь UP болон SMP цөмүүдэд хоёуланд нь ашиглагдаж болох бөгөөд гэхдээ SMP цөмд зайлшгүй шаардлагатай. `options SMP` мөрийг нэмж олон процессорын дэмжлэгийг оруулна уу.

apic төхөөрөмж нь зөвхөн i386 архитектур дээр байдаг бөгөөд энэ тохиргооны мөрийг бусад архитектурууд дээр ашиглах ёсгүй юм.

```
device          eisa
```

Хэрэв та EISA эх хавтантай бол үүнийг оруулаарай. Энэ нь EISA шугамын бүх төхөөрөмжүүдийн хувьд автомат илрүүлэлт болон тохиргооны дэмжлэгийг нээж өгдөг.

```
device          pci
```

Хэрэв та PCI эх хавтантай бол үүнийг оруулаарай. Энэ нь PCI картуудыг автомат илрүүлэлт болон PCI-аас ISA шугам уруу гарах дэмжлэгийг нээж өгдөг.

```
# Floppy drives
device          fdc
```

Энэ нь уян диск хөтлөгчийн хянагч.

```
# ATA and ATAPI devices
device          ata
```

Энэ драйвер бүх ATA болон ATAPI төхөөрөмжүүдийг дэмждэг. Орчин үеийн машинуудын бүх PCI ATA/ATAPI төхөөрөмжүүдийг илрүүлэхийн тулд зөвхөн нэг `device ata` мөр таны цөмд хэрэгтэй.

```
device          atadisk                 # ATA disk drives
```

Энэ нь `device ata` мөртэй цуг ATA дискний төхөөрөмжүүдэд хэрэгтэй.

```
device          ataraid                 # ATA RAID drives
```

Энэ нь `device ata` мөртэй цуг ATA RAID хөтлөгчүүдэд хэрэгтэй.

```
device          atapicd                 # ATAPI CDROM drives
```

Энэ нь `device ata` мөртэй цуг ATA CDROM хөтлөгчүүдэд хэрэгтэй.

```
device          atapifd                 # ATAPI floppy drives
```

Энэ нь `device ata` мөртэй цуг ATA уян дискний хөтлөгчүүдэд хэрэгтэй.

```
device          atapist                 # ATAPI tape drives
```

Энэ нь `device ata` мөртэй цуг ATA соронзон хальсны хөтлөгчүүдэд хэрэгтэй.

```
options         ATA_STATIC_ID           # Static device numbering
```

Энэ нь хянагчийн дугаарыг статик болгох бөгөөд энэнгүйгээр төхөөрөмжийн дугаарууд динамикаар өгөгддөг.

```
# SCSI Controllers
device          ahb         # EISA AHA1742 family
device          ahc         # AHA2940 and onboard AIC7xxx devices
options         AHC_REG_PRETTY_PRINT    # Print register bitfields in debug
            # output.  Adds ~128k to driver.
device          ahd         # AHA39320/29320 and onboard AIC79xx devices
options         AHD_REG_PRETTY_PRINT    # Print register bitfields in debug
            # output.  Adds ~215k to driver.
device          amd         # AMD 53C974 (Teckram DC-390(T))
device          isp         # Qlogic family
#device         ispfw       # Firmware for QLogic HBAs- normally a module
device          mpt         # LSI-Logic MPT-Fusion
#device         ncr         # NCR/Symbios Logic
device          sym         # NCR/Symbios Logic (newer chipsets + those of `ncr')
device          trm         # Tekram DC395U/UW/F DC315U adapters

device          adv         # Advansys SCSI adapters
device          adw         # Advansys wide SCSI adapters
device          aha         # Adaptec 154x SCSI adapters
device          aic         # Adaptec 15[012]x SCSI adapters, AIC-6[23]60.
device          bt          # Buslogic/Mylex MultiMaster SCSI adapters
```

```
device          ncv         # NCR 53C500
device          nsp         # Workbit Ninja SCSI-3
device          stg         # TMC 18C30/18C50
```

SCSI хянагчууд. Таны системд байхгүй байгааг тайлбар болгон хаана уу. Хэрэв та зөвхөн IDE системтэй бол эдгээр мөрүүдийг бүгдийг устгаж болно. `*_REG_PRETTY_PRINT` мөрүүд нь өөр өөрийнхөө тохирох драйверуудад зориулагдсан дибаг хийх тохируулгууд юм.

```
# SCSI peripherals
device          scbus       # SCSI bus (required for SCSI)
device          ch          # SCSI media changers
device          da          # Direct Access (disks)
device          sa          # Sequential Access (tape etc)
device          cd          # CD
device          pass        # Passthrough device (direct SCSI access)
device          ses         # SCSI Environmental Services (and SAF-TE)
```

SCSI захын төхөөрөмжүүд. Дахин хэлэхэд таны системд байхгүй байгааг тайлбар болгон хаагаарай эсвэл та зөвхөн IDE тоног төхөөрөмжтэй бол эдгээр мөрүүдийг бүгдийг устгаж болно.

USB umass(4) драйвер болон бусад цөөн хэдэн драйверууд жинхэнэ SCSI төхөөрөмжүүд биш боловч SCSI дэд системийг ашигладаг. Тийм болохоор хэрэв цөмийн тохиргоонд тийм драйверууд орсон байвал SCSI дэмжлэгийг устгаагүй эсэхээ шалгаарай.

```
# RAID controllers interfaced to the SCSI subsystem
device          amr         # AMI MegaRAID
device          arcmsr      # Areca SATA II RAID
device          asr         # DPT SmartRAID V, VI and Adaptec SCSI RAID
device          ciss        # Compaq Smart RAID 5*
device          dpt         # DPT Smartcache III, IV - See NOTES for options
device          hptmv       # Highpoint RocketRAID 182x
device          hptrr       # Highpoint RocketRAID 17xx, 22xx, 23xx, 25xx
device          iir         # Intel Integrated RAID
device          ips         # IBM (Adaptec) ServeRAID
device          mly         # Mylex AcceleRAID/eXtremeRAID
device          twa         # 3ware 9000 series PATA/SATA RAID

# RAID controllers
device          aac         # Adaptec FSA RAID
device          aacp        # SCSI passthrough for aac (requires CAM)
device          ida         # Compaq Smart RAID
device          mfi         # LSI MegaRAID SAS
device          mlx         # Mylex DAC960 family
device          pst         # Promise Supertrak SX6000
```

```
device          twe         # 3ware ATA RAID
```

Дэмжигдсэн RAID хянагчууд. Хэрэв танд эдгээрээс нэг нь ч байхгүй бол тайлбар болгон хааж эсвэл устгаж болно.

```
# atkbdc0 controls both the keyboard and the PS/2 mouse
device          atkbdc      # AT keyboard controller
```

Гарын хянагч (atkbdc) AT гар ба PS/2 загварын заагч төхөөрөмжүүдэд I/O үйлчилгээнүүдийг хангадаг. Энэ хянагчийг гарын драйвер (atkbd) болон PS/2 заагч төхөөрөмжийн драйвер (psm) шаарддаг.

```
device          atkbd       # AT keyboard
```

atkbd драйвер нь atkbdc хянагчтай цуг AT гарын хянагчид залгасан AT 84 гар болон AT өргөжүүлсэн гаруудад хандах боломж олгодог.

```
device          psm         # PS/2 mouse
```

Хэрэв таны хулгана PS/2 порт уруу залгагдаж байгаа бол энэ төхөөрөмжийг ашиглана уу.

```
device          kbdmux         # keyboard multiplexer
```

Олон гар залгах боломжийн анхны дэмжлэг. Хэрэв та систем дээрээ нэгээс олон гар ашиглахгүй бол энэ мөрийг аюулгүйгээр устгаж болно.

```
device          vga         # VGA video card driver
```

Видео картын драйвер.

```
device          splash      # Splash screen and screen saver support
```

Эхлэх үе дэх хоромхон зуур гарах дэлгэц! Дэлгэц амраагчид нь үүнийг бас шаарддаг.

```
# syscons is the default console driver, resembling an SCO console
device          sc
```

sc нь анхдагч консолийн драйвер бөгөөд SCO консолыг дуурайдаг. Ихэнх бүрэн дэлгэцийн програмууд консол уруу termcap зэрэг терминалийн өгөгдлийн баазын сангийн тусламжтайгаар ханддаг бөгөөд үүнийг ашиглах эсэх эсвэл VT220-тай нийцтэй консол драйвер болох vt-ийг ашиглах эсэх нь хамаагүй юм. Та нэвтэрсний дараа бүрэн дэлгэцийн

програмууд энэ консол дээр ажиллахдаа асуудалтай байвал өөрийн TERM хувьсагчаа scoansi болгоорой.

```
# Enable this for the pcvt (VT220 compatible) console driver
#device          vt
#options         XSERVER         # support for X server on a vt console
#options         FAT_CURSOR      # start with block cursor
```

Энэ нь VT100/102-той арагшаагаа нийцтэй, VT220-той нийцтэй консол драйвер юм. Энэ нь sc -тэй нийцгүй зарим зөөврийн компьютер дээр сайн ажилладаг. Та нэвтэрсний дараа өөрийн TERM хувьсагчаа vt100 эсвэл vt220 болгоорой. Мөн энэ драйвер нь sc төхөөрөмжүүдэд зориулсан termcap эсвэл terminfo оруулгууд ихэнхдээ байхгүй байдаг асар олон тооны өөр өөр машинууд уруу сүлжээгээр дамжин холбогдох үед ашигтай байж болно - vt100 нь виртуалаар дурын тавцан дээр байх ёстой.

```
device          agp
```

Хэрэв та систем дээрээ AGP карттай бол үүнийг оруулна уу. Энэ нь AGP болон AGP GART-ийн дэмжлэгийг эдгээр боломжуудаас тогтсон хавтанд зориулж нээнэ.

```
# Power management support (see NOTES for more options)
#device          apm
```

Тэжээлийн Нарийн Удирдлагын дэмжлэг. Зөөврийн компьютеруудад ашигтай, гэхдээ энэ нь GENERIC-д анхдагчаар хаалттай байдаг.

```
# Add suspend/resume support for the i8254.
device          pmtimer
```

APM болон ACPI зэрэг тэжээл удирдах үйл явцуудад зориулсан таймер төхөөрөмжийн драйвер.

```
# PCCARD (PCMCIA) support
# PCMCIA and cardbus bridge support
device          cbb             # cardbus (yenta) bridge
device          pccard          # PC Card (16-bit) bus
device          cardbus         # CardBus (32-bit) bus
```

PCMCIA дэмжлэг. Зөөврийн компьютер ашиглаж байгаа бол энэ танд хэрэгтэй.

```
# Serial (COM) ports
device          sio             # 8250, 16[45]50 based serial ports
```

Эдгээр нь MS-DOS®/Windows® ертөнцөд COM портууд хэмээгддэг сериал портууд юм .

Хэрэв та COM4 дээр дотуур модемтой бөгөөд COM2 дээр сериал порттой бол FreeBSD -ээс хандахын тулд та модемийн IRQ-г 2 (ойлгоход амаргүй техникийн шалтгаанаар, IRQ2 = IRQ 9) болгож өөрчлөх хэрэгтэй. Хэрэв та олон порттой сериал карттай бол өөрийн /boot/device.hints файлд нэмэх зөв утгуудын талаар дэлгэрэнгүй мэдээллийг sio(4) гарын авлагаас шалгаарай. Зарим видео картууд (S3 бичил схем дээр үндэслэснүүдийг дурдаж болно) 0x*2e8 хэлбэртэй IO хаягуудыг ашигладаг бөгөөд олон хямд сериал картууд 16-бит хаягийн зайг бүрэн декод хийж чаддаггүй учир энэ нь тэдгээр карттай зөрчилдөж COM4 портыг бараг л ашиглах боломжгүй болгодог.

Сериал порт болгон бусдаас ялгаатай IRQ -тай байх (хуваалцсан тасалдал ашиглахыг дэмждэг олон порттой картуудын нэгийг ашиглаж байгаагаас бусад тохиолдолд) шаардлагатай учир COM3 болон COM4-ийн анхдагч IRQ-үүдийг ашиглаж болохгүй.

```
# Parallel port
device          ppc
```

Энэ нь ISA-шугамын параллел порт интерфэйс юм.

```
device          ppbus      # Parallel port bus (required)
```

Параллел портын шугамын дэмжлэгийг хангадаг.

```
device          lpt        # Printer
```

Параллел порт хэвлэгчүүдийн дэмжлэг.

Параллел хэвлэгчийн дэмжлэгийг нээхэд дээрх гурав гурвуулаа шаардлагатай.

```
device          plip       # TCP/IP over parallel
```

Энэ нь параллел сүлжээний интерфэйсд зориулсан драйвер юм.

```
device          ppi        # Parallel port interface device
```

Ерөнхий-зориулалтын I/O ("geek port") + IEEE1284 I/O.

```
#device         vpo        # Requires scbus and da
```

Энэ нь Iomega Zip хөтлөгчид зориулагдсан юм. `scbus` болон `da` дэмжлэгийг шаарддаг. EPP 1.9 горимд байгаа портуудын тусламжтай хамгийн сайн ажиллагаанд хүрдэг.

```
#device         puc
```

Хэрэв та puc(4) цавуу драйвераар дэмжигддэг "дүлий" сериал эсвэл параллел PCI карттай бол энэ төхөөрөмжийг тайлбар болгосныг болиулаарай.

```
# PCI Ethernet NICs.
device          de          # DEC/Intel DC21x4x (Tulip)
device          em          # Intel PRO/1000 adapter Gigabit Ethernet Card
device          ixgb        # Intel PRO/10GbE Ethernet Card
device          txp         # 3Com 3cR990 (Typhoon)
device          vx          # 3Com 3c590, 3c595 (Vortex)
```

Төрөл бүрийн PCI сүлжээний картуудын драйверууд. Эдгээрээс таны системд байхгүйг тайлбар болгон хааж эсвэл устгаарай.

```
# PCI Ethernet NICs that use the common MII bus controller code.
# NOTE: Be sure to keep the 'device miibus' line in order to use these NICs!
device          miibus      # MII bus support
```

MII шугамын дэмжлэг нь зарим PCI 10/100 Ethernet NIC-үүдэд шаардлагатай бөгөөд тухайлбал MII-д нийцтэй дамжуулагч-хүлээн авагчууд эсвэл MII-тэй адил ажилладаг дамжуулагч-хүлээн авагчийн хяналтын интерфэйсүүдийг дурдаж болно. `device miibus` мөрийг цөмийн тохиргоонд нэмснээр ердийн miibus API болон тусдаа драйвераар дэмжигдээгүй PHY-уудад зориулсан ердийн драйвер зэрэг бүх PHY драйверуудын дэмжлэгийг оруулах болно.

```
device          bce         # Broadcom BCM5706/BCM5708 Gigabit Ethernet
device          bfe         # Broadcom BCM440x 10/100 Ethernet
device          bge         # Broadcom BCM570xx Gigabit Ethernet
device          dc          # DEC/Intel 21143 and various workalikes
device          fxp         # Intel EtherExpress PRO/100B (82557, 82558)
device          lge         # Level 1 LXT1001 gigabit ethernet
device          msk         # Marvell/SysKonnect Yukon II Gigabit Ethernet
device          nge         # NatSemi DP83820 gigabit ethernet
device          nve         # nVidia nForce MCP on-board Ethernet Networking
device          pcn         # AMD Am79C97x PCI 10/100 (precedence over 'lnc')
device          re          # RealTek 8139C+/8169/8169S/8110S
device          rl          # RealTek 8129/8139
device          sf          # Adaptec AIC-6915 (Starfire)
device          sis         # Silicon Integrated Systems SiS 900/SiS 7016
device          sk          # SysKonnect SK-984x & SK-982x gigabit Ethernet
device          ste         # Sundance ST201 (D-Link DFE-550TX)
device          stge        # Sundance/Tamarack TC9021 gigabit Ethernet
```

```
device          ti          # Alteon Networks Tigon I/II gigabit Ethernet
device          tl          # Texas Instruments ThunderLAN
device          tx          # SMC EtherPower II (83c170 EPIC)
device          vge         # VIA VT612x gigabit ethernet
device          vr          # VIA Rhine, Rhine II
device          wb          # Winbond W89C840F
device          xl          # 3Com 3c90x (Boomerang, Cyclone)
```

MII шугамын хянагчийн кодыг ашигладаг драйверууд.

```
# ISA Ethernet NICs.  pccard NICs included.
device          cs          # Crystal Semiconductor CS89x0 NIC
# 'device ed' requires 'device miibus'
device          ed          # NE[12]000, SMC Ultra, 3c503, DS8390 cards
device          ex          # Intel EtherExpress Pro/10 and Pro/10+
device          ep          # Etherlink III based cards
device          fe          # Fujitsu MB8696x based cards
device          ie          # EtherExpress 8/16, 3C507, StarLAN 10 etc.
device          lnc         # NE2100, NE32-VL Lance Ethernet cards
device          sn          # SMC's 9000 series of Ethernet chips
device          xe          # Xircom pccard Ethernet

# ISA devices that use the old ISA shims
#device         le
```

ISA Ethernet драйверууд. Аль картууд аль драйвераар дэмжигддэг талаар дэлгэрэнгүйг /usr/src/sys/i386/conf/NOTES-ээс харна уу.

```
# Wireless NIC cards
device          wlan            # 802.11 support
```

802.11-ийн ерөнхий дэмжлэг. Энэ мөр нь утасгүй сүлжээнд шаардлагатай.

```
device          wlan_wep        # 802.11 WEP support
device          wlan_ccmp       # 802.11 CCMP support
device          wlan_tkip       # 802.11 TKIP support
```

802.11 төхөөрөмжүүдэд зориулагдсан криптограф дэмжлэг. Хэрэв та шифрлэлт болон 802.11i нууцлалын протоколуудыг ашиглахаар шийдсэн бол эдгээр мөрүүд шаардлагатай.

```
device          an          # Aironet 4500/4800 802.11 wireless NICs.
device          ath             # Atheros pci/cardbus NIC's
device          ath_hal         # Atheros HAL (Hardware Access Layer)
device          ath_rate_sample # SampleRate tx rate control for ath
device          awi         # BayStack 660 and others
device          ral         # Ralink Technology RT2500 wireless NICs.
```

```
device          wi          # WaveLAN/Intersil/Symbol 802.11 wireless NICs.
#device         wl          # Older non 802.11 Wavelan wireless NIC.
```

Төрөл бүрийн утасгүй сүлжээний картуудын дэмжлэг.

```
# Pseudo devices
device   loop          # Network loopback
```

Энэ нь TCP/IP-д зориулсан өөртөө эргэн холбогдох ерөнхий төхөөрөмж юм. Хэрэв та `localhost` (өөрөөр бас `127.0.0.1`) уруу telnet эсвэл FTP хийвэл энэ нь тань уруу энэ төхөөрөмжөөр эргэж ирнэ.Энэ нь *зайлшгүй шаардлагатай*.

```
device   random        # Entropy device
```

Шифрлэлтийн хувьд аюулгүй дурын тоо үүсгэгч.

```
device   ether         # Ethernet support
```

Танд Ethernet карт байгаа тохиолдолд зөвхөн `ether` хэрэгтэй. Энэ нь ерөнхий Ethernet протоколын кодыг агуулдаг.

```
device   sl            # Kernel SLIP
```

`sl` нь SLIP-ийн дэмжлэг юм. Энэ нь хялбар суулгаж тохируулдаг, модемоос модем уруу холболт хийхэд илүү зохицсон, илүү чадалтай PPP-ээр бараг бүхэлдээ солигдсон юм.

```
device   ppp           # Kernel PPP
```

Энэ нь дайл-ап холболтын цөм дэх PPP дэмжлэгт зориулагдсан. Мөн `tun`-ийг ашиглаж илүү уян хатан чанар болон шаардлагаар залгах зэрэг боломжууд бүхий PPP-ийн хэрэглэгчийн талбарын програм маягаар хийгдсэн хувилбар ч бас байдаг.

```
device   tun           # Packet tunnel.
```

Энэ хэрэглэгчийн талбарын PPP програм хангамжид ашиглагддаг. Дэлгэрэнгүй мэдээллийг энэ номын PPP хэсгээс үзнэ үү.

```
device   pty           # Pseudo-ttys (telnet etc)
```

Энэ нь "псевдо-терминал" эсвэл жинхэнэ биш (simulated) нэвтрэх порт юм. Энэ нь ирж байгаа `telnet` болон `rlogin` сессүүд, xterm, болон Emacs зэрэг бусад програмуудад

ашиглагддаг.

```
device   md            # Memory disks
```

Санах ойн диск псевдо-төхөөрөмжүүд.

```
device   gif           # IPv6 and IPv4 tunneling
```

Энэ нь IPv4 дээгүүрх IPv6, IPv6 дээгүүрх IPv4, IPv4 дээгүүрх IPv4, болон IPv6 дээгүүрх IPv6 хоолойнуудыг хийдэг. `gif` төхөөрөмж нь "авто-клон" хийх чадвартай бөгөөд төхөөрөмжийн цэгүүдийг хэрэгцээгээрээ үүсгэдэг.

```
device   faith         # IPv6-to-IPv4 relaying (translation)
```

Энэ псевдо-төхөөрөмж нь түүн уруу илгээсэн пакетуудыг барьж аваад IPv4/IPv6 хувиргагч дэмон уруу замыг өөрчлөн явуулдаг.

```
# The `bpf' device enables the Berkeley Packet Filter.
# Be aware of the administrative consequences of enabling this!
# Note that 'bpf' is required for DHCP.
device   bpf           # Berkeley packet filter
```

Энэ нь Беркли Пакет Шүүгч юм. Энэ псевдо-төхөөрөмж нь цацаж байгаа сүлжээн (өөрөөр хэлбэл Ethernet) дэх бүх пакетуудыг барьж авдаг, бүгдийг сонсох горимд сүлжээний интерфэйсүүдийг шилжүүлэх боломж олгодог. Эдгээр пакетуудыг дискэнд хадгалах болон эсвэл tcpdump(1) програмаар шалгаж болно.

Анхдагч чиглүүлэгчийн (гарц) болон бусад IP хаягийг dhclient(8) олж авахад bpf(4) төхөөрөмжийг бас ашигладаг. Хэрэв та DHCP ашиглаж байгаа бол үүнийг тайлбар болголгүй орхиорой.

```
# USB support
device          uhci          # UHCI PCI->USB interface
device          ohci          # OHCI PCI->USB interface
device          ehci          # EHCI PCI->USB interface (USB 2.0)
device          usb           # USB Bus (required)
#device         udbp          # USB Double Bulk Pipe devices
device          ugen          # Generic
device          uhid          # Human Interface Devices
device          ukbd          # Keyboard
device          ulpt          # Printer
device          umass         # Disks/Mass storage - Requires scbus and da
device          ums           # Mouse
device          ural          # Ralink Technology RT2500USB wireless NICs
device          urio          # Diamond Rio 500 MP3 player
```

```
device          uscanner      # Scanners
# USB Ethernet, requires mii
device          aue           # ADMtek USB Ethernet
device          axe           # ASIX Electronics USB Ethernet
device          cdce          # Generic USB over Ethernet
device          cue           # CATC USB Ethernet
device          kue           # Kawasaki LSI USB Ethernet
device          rue           # RealTek RTL8150 USB Ethernet
```

Төрөл бүрийн USB төхөөрөмжүүдийн дэмжлэг.

```
# FireWire support
device          firewire      # FireWire bus code
device          sbp           # SCSI over FireWire (Requires scbus and da)
device          fwe           # Ethernet over FireWire (non-standard!)
```

Төрөл бүрийн Firewire төхөөрөмжүүдийн дэмжлэг.

FreeBSD-ийн дэмждэг нэмэлт төхөөрөмжүүдийн талаар дэлгэрэнгүй мэдээллийг /usr/src/sys/i386/conf/NOTES файлаас үзнэ үү.

### 9.6.1. Том санах ойн тохируулгууд (PAE)

Том санах ой бүхий машинууд Хэрэглэгч+Цөмийн Виртуал хаягийн зайн (KVA) 4 гигабайт хязгаараас их уруу хандах шаардлагатай байдаг. Энэ хязгаараас болоод Интел Pentium® Pro болон сүүлийн үеийн CPU-үүдэд 36 битийн физик хаягийн зайд хандах дэмжлэгийг нэмсэн.

Intel® Pentium® Pro болон сүүлийн үеийн CPU-үүдийн Физик Хаягийн Өргөтгөл (PAE) боломж нь 64 гигабайт хүртэлх санах ойн тохиргоог зөвшөөрдөг. FreeBSD нь энэхүү боломжийг FreeBSD-ийн одоо байгаа бүх гаргасан хувилбаруудын цөмийн тохиргооны PAE тохируулгаар дэмждэг. Интелийн санах ойн архитектурын хязгаараас болж 4 гигабайтаас дээш болон доош санах ойн ялгаа байхгүй. 4 гигабайтаас дээшхи санах ой нь ашиглаж болох санах ойн санд нэмэгддэг.

PAE дэмжлэгийг цөмд нээхдээ цөмийн тохиргооны файлд дараах мөрийг нэмнэ:

```
options         PAE
```

FreeBSD дэх PAE дэмжлэг зөвхөн Intel® IA-32 процессоруудад байдаг. Мөн FreeBSD дэх PAE дэмжлэг нь өргөн шалгагдаагүй бөгөөд FreeBSD-ийн бусад тогтвортой боломжуудтай харьцуулахад бета чанарынхад тооцогддог.

FreeBSD дэх PAE дэмжлэг нь цөөн хэдэн хязгаарууд тай:

- VM зайн 4 гигабайтаас илүүд процесс хандаж чадахгүй.
- bus_dma(9) интерфэйс ашигладаггүй төхөөрөмжийн драйверууд PAE-г идэвхжүүлсэн цөм дээр өгөгдлийн эвдрэлийг үүсгэх бөгөөд ашиглахыг зөвлөдөггүй юм. Иймээс учраас PAE-г

идэвхжүүлсэн цөм дээр ажилладаггүй бүх драйверуудыг оруулаагүй PAE цөмийн тохиргооны файл FreeBSD-д байдаг.

- Зарим нэг тохируулгууд санах ойн эх үүсвэрийн хэрэглээг физик санах ойн хэмжээгээр тодорхойлдог. Эдгээр тохируулгууд нь PAE системийн их санах ойгоос болж хэрэгцээгүй илүү санах ойг гаргадаг. Тийм нэг жишээнүүдийн нэг нь sysctl-ийн `kern.maxvnodes` тохируулга бөгөөд энэ нь цөм дэх хамгийн их байж болох vnode-уудын тоог хянадаг. Энэ болон бусад тохируулгуудын утгыг боломжийн утгаар тааруулахыг зөвлөж байна.
- Магадгүй цөмийн виртуал хаягийн (KVA) зайг ихэсгэх эсвэл KVA-ийн шавхалтад хүргэхгүйн тулд байнга их ашиглагддаг (дээр дурдсаныг харна уу) цөмийн эх үүсвэрийн хэмжээг багасгах шаардлагатай байж болох юм. `KVA_PAGES` цөмийн тохируулга нь KVA зайг ихэсгэхэд ашиглагдаж болно.

Ажиллагаа болон тогтвортой байдлыг хангах үүднээс tuning(7) гарын авлагатай танилцахыг зөвлөж байна. pae(4) гарын авлага нь FreeBSD-ийн PAE дэмжлэгийн тухай хамгийн сүүлийн үеийн мэдээллийг агуулдаг.

# 9.7. Хэрэв ямар нэг юм буруутвал

Өөрчлөн тохируулсан цөмийг бүтээж байх үед 4 төрлийн асуудал гарч болзошгүй байдаг. Тэдгээр нь:

### `config` амжилттүй болох

Хэрэв таныг цөмийн тайлбарыг config(8)-т өгөхөд тушаал амжилттүй болбол та хаа нэгтээ энгийн алдаа хийсэн болов уу. Аз болоход config(8) асуудалтай байгаа мөрийн дугаарыг хэвлэх учир та алдаатай мөрийг хурдан олох болно. Жишээ нь, хэрэв та доор дурдсаныг харвал:

```
config: line 17: syntax error
```

Түлхүүр үг зөв бичигдсэн эсэхийг GENERIC цөм болон бусад баримтаас харьцуулан шалгаж үзээрэй.

### `make` амжилттүй болох

Хэрэв `make` тушаал амжилттүй болбол энэ ихэвчлэн цөмийн тайлбар дахь config(8)-ийн олж чадахааргүй тийм ч ноцтой бус алдааг дохиолдог. Дахин хэлэхэд, өөрийн тохиргоог нягтлаарай, тэгээд хэрэв та асуудлыг шийдэж чадахгүй бол FreeBSD ерөнхий асуултууд захидлын жагсаалт уруу өөрийн цөмийн тохиргоотой цахим захидал илгээгээрэй, ингэхэд хурдан шинжилгээ хийгдэх болно.

#### Цөм ачаалахгүй байх:

Хэрэв таны шинэ цөм ачаалахгүй бол эсвэл таны төхөөрөмжүүдийг танихгүй байгаа бол бүү цочирд! Аз болоход FreeBSD нийцгүй цөмүүдээс сэргэхэд зориулсан маш сайн механизмтай байдаг. Ердөө л FreeBSD-ийн ачаалагчаас ачаалах цөмөө сонгоно. Системийн ачаалах меню гарч ирэх үед та үүнд хандах боломжтой болно. "Escape to a loader prompt" тохируулга 6-ын тоог сонго. Тушаал хүлээх мөрөн дээр `boot kernel.old` эсвэл зөв ачаалах өөр бусад цөмийн нэрийг бичээрэй. Цөмийг дахин тохируулах

явцдаа ажилладаг цөмийг гарын дор хадгалж байх нь үргэлж ухаалаг санаа байдаг.

Сайн цөмийг ачаалсныхаа дараа та өөрийн тохиргооны файлаа дахин шалгаж цөмөө дахин бүтээхээр оролдоорой. Нэг тус дэм болох эх үүсвэр бол бусад зүйлсээс гадна амжилттай ачаалалт бүр дэх цөмийн бүх мэдээллүүдийн бичлэгийг хийдэг /var/log/messages файл юм. Мөн dmesg(8) тушаал нь сүүлийн ачаалалт дахь цөмийн мэдээллүүдийг хэвлэдэг.

Хэрэв та цөмийг бүтээхэд асуудалтай байгаа бол GENERIC болон өөр бусад ажилладаг цөмийг дараагийн бүтээх явцад устгагдахааргүй өөр нэртэйгээр гарын дор хадгалж байгаарай. Та kernel.old -д найдаж болохгүй, учир нь шинэ цөмийг суулгах явцад kernel.old нь хамгийн сүүлд суулгагдсан бөгөөд магадгүй ажиллагаагүй тийм цөмөөр дарагдан бичигддэг. Ажилладаг цөмийг зөв /boot/kernel байр уруу аль болох хурдан шилжүүлээрэй, эсвэл ps(1) зэрэг тушаалууд зөв ажиллахгүй байж магадгүй юм. Ингэхийн тулд сайн цөм байгаа сангийн нэрийг өөрчлөөрэй:

```
# mv /boot/kernel /boot/kernel.bad
# mv /boot/kernel.good /boot/kernel
```

**Цөм ажиллах боловч ps(1) ажиллахгүй болох**

Хэрэв та системийн хэрэгслүүд бүтээсэн хувилбараас өөр цөмийн хувилбарыг суулгасан бол, жишээ нь -RELEASE дээр -CURRENT цөм бүтээсэн бол системийн төлвийн тушаалууд болох ps(1) болон vmstat(8) зэрэг нь ажиллахаа больно. Та өөрийн цөмтэйгөө адил хувилбарын эх модтой бүтээгдсэн ертөнцийг дахин хөрвүүлж суулгах хэрэгтэй. Энэ нь нэг шалтгаан бөгөөд ерөнхийдөө үйлдлийн системийн бусад зүйлсээс өөр хувилбарын цөмийг ашиглах нь тийм ч сайн санаа биш юм.

# Chapter 10. Хэвлэлт

## 10.1. Ерөнхий агуулга

FreeBSD дээр хамгийн хуучин цохидог хэвлэгчдээс авахуулаад хамгийн сүүлийн хэвлэгчид хүртэл, мөн тэдгээрийн хооронд байгаа хэвлэгчид зэрэг төрөл бүрийн хэвлэгчдийг хэвлэхэд ашиглаж болдог бөгөөд энэ нь таны ажиллаж байгаа програмуудаас өндөр чанарын хэвлэсэн гаралтыг бүтээх боломж олгох юм.

FreeBSD-г бас сүлжээн дэх хэвлэх сервер болгон тохируулж бас болдог; энэ боломжид FreeBSD нь бусад FreeBSD компьютерууд, Windows® болон Mac OS® хостууд зэрэг төрөл бүрийн бусад компьютеруудаас хэвлэх ажлуудыг хүлээн авдаг. FreeBSD нь зөвхөн нэг ажлыг тухайн үед хэвлэхийг баталгаажуулж ихэнх хэвлэлтийг аль хэрэглэгчид болон машинууд хийж байгаа талаар статистикууд цуглуулж хэний хэвлэсэн зүйл хэнийх болохыг үзүүлэх "сурталчилгаа" хуудсуудыг бүтээх зэрэг олон үйлдлийг хийж чаддаг.

Энэ бүлгийг уншиж дууссаны дараа та дараах зүйлсийг мэдэх болно:

- FreeBSD-ийн хэвлэгчийн түр хадгалагчийг хэрхэн тохируулах талаар.
- Ирж байгаа баримтуудыг таны хэвлэгчид ойлгодог хэвлэх хэлбэршилтэд оруулж хөрвүүлэх зэрэг тусгай хэвлэх ажлуудыг өөрөөр зохицуулдаг хэвлэх шүүлтүүрүүдийг хэрхэн суулгах талаар.
- Толгой эсвэл сурталчилгаа хуудаснуудыг өөрийн хэвлэх зүйл дээрээ хэрхэн идэвхжүүлэх талаар.
- Бусад компьютерууд уруу холбогдсон хэвлэгчдээр хэрхэн хэвлэх талаар.
- Сүлжээнд шууд холбогдсон хэвлэгчдээр хэрхэн хэвлэх талаар.
- Хэвлэх ажлуудын хэмжээг хязгаарлах болон зарим хэрэглэгчдийг хэвлэхийг болиулах зэрэг хэвлэгчийн хязгаарлалтуудыг хэрхэн хянах талаар.
- Хэвлэгчийн статистикууд болон хэвлэгчийн хэрэглээнд зориулсан бүртгэлийг хэрхэн хадгалж байх талаар.
- Хэвлэх асуудлуудыг хэрхэн олж засварлах талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- Шинэ цөмийг хэрхэн тохируулж суулгах талаар мэдэх (FreeBSD цөмийг тохируулах нь).

## 10.2. Танилцуулга

FreeBSD дээр хэвлэгчдийг ашиглахын тулд та тэдгээрийг LPD түр хадгалах систем буюу ердөө л LPD гэж бас нэрлэгддэг Берклигийн шугаман хэвлэгчийн түр хадгалах системтэй ажиллахаар болгож тохируулж болох юм. Энэ нь FreeBSD дээрх стандарт хэвлэгч хянагч систем юм. Энэ бүлэг нь LPD-г танилцуулж түүний тохиргоог тайлбарлах болно.

Хэрэв та LPD юм уу эсвэл бусад хэвлэгчийн түр хадгалах системийг мэддэг бол Үндсэн тохируулга хэсэг уруу шууд орохыг хүсэж болох юм.

LPD нь хостын хэвлэгчдийн талаар бүгдийг хянадаг. Энэ нь хэд хэдэн зүйлсийг хариуцдаг:

- Залгагдсан хэвлэгчид болон сүлжээн дэх өөр хостуудад залгагдсан хэвлэгчдэд хандах хандалтыг хянадаг.
- Файлууд хэвлэхээр өгөх боломжийг хэрэглэгчдэд зөвшөөрдөг; эдгээр өгөлтүүд нь *jobs* буюу *ажлууд* гэгддэг.
- Хэвлэгч болгоны хувьд *queue* буюу дарааллыг зохицуулж олон хэрэглэгчид нэгэн зэрэг хэвлэгчид хандах хандалтаас сэргийлдэг.
- Хэрэглэгчид хэвлэсэн тоо томшгүй олон юмнаас өөрсдийн хэвлэсэн ажлуудыг хялбархан олдог байхын тулд энэ нь *толгой хуудаснуудыг* (бас *сурталчилгаа* эсвэл *тэсрэлт* хуудсууд гэгддэг) хэвлэдэг.
- Цуваа портууд дээр холбогдсон хэвлэгчдийн холбооны параметрүүдэд анхаарлаа хандуулдаг.
- Өөр хост дээр байгаа LPD түр хадгалагч уруу сүлжээгээр ажлууд илгээж чаддаг.
- Төрөл бүрийн хэвлэгчийн хэлнүүд эсвэл хэвлэгчийн боломжуудад зориулж хэвлэх ажлуудыг хэлбэршүүлэх тусгай шүүлтүүрүүдийг ажиллуулж чаддаг.
- Хэвлэгчийн хэрэглээг бүртгэж чаддаг.

Тохиргооны файл (/etc/printcap) болон тусгай шүүлтүүр програмууд ашиглан олон төрлийн хэвлэгч тоног төхөөрөмжүүдийн хувьд дээр дурдсануудын заримууд болон бүгдийг хийдэг байхаар LPD системийг та идэвхжүүлж чадна.

### 10.2.1. Яагаад заавал түр хадгалагчийг ашиглах ёстой гэж

Ганц хэрэглэгчийн систем дээр түр хадгалагч хэрэгтэй хэвээр байдаг бөгөөд түр хадгалагчийг ашиглах ёстой юм, учир нь:

- LPD нь ажлуудыг ард хэвлэдэг; та өгөгдлийг хэвлэгч уруу хуулагдахыг хүлээх хэрэггүй юм.
- LPD нь огноо/цаг бүхий толгой нэмэх эсвэл тусгай файлын хэлбэршилтээс (TeX DVI файл зэрэг) хэвлэгчийн ойлгох хэлбэршилт уруу хөрвүүлдэг шүүлтүүрүүдээр дамжуулан хэвлэгдэх ажлыг тохиромжтойгоор ажиллуулдаг. Та гараараа эдгээр алхмуудыг хийх шаардлагагүй юм.
- Хэвлэх боломж бүхий чөлөөтэй, арилжааны олон програмууд нь таны систем дээрх түр хадгалагчтай ярилцахыг ихэвчлэн хүлээж байдаг. Түр хадгалагч системийг тохируулснаар танд байгаа болон таны сүүлд суулгаж болох бусад програм хангамжуудыг илүү амархнаар та дэмжих болно.

## 10.3. Үндсэн тохируулга

LPD түр хадгалах системтэй хэвлэгчдийг ашиглахын тулд өөрийн хэвлэгчийн тоног төхөөрөмж болон LPD програм хангамжийг тохируулах хэрэгтэй болно. Энэ баримт нь тохиргооны хоёр түвшинг тайлбарладаг:

- Хэвлэгчийг хэрхэн холбохыг сурахын тулд Хэвлэгчийн хялбар тохиргоо хэсгийг үзэж

LPD-д хэрхэн түүнтэй холбоо тогтоохыг хэлж хэвлэгч уруу цэвэр текст файлуудыг хэвлэх.

- Төрөл бүрийн тусгай файлын хэлбэршилтүүдийг хэрхэн хэвлэх, толгой хуудаснуудыг хэрхэн хэвлэх, сүлжээгээр хэрхэн хэвлэх, хэвлэгчдэд хандах хандалтыг хэрхэн хянах болон хэрхэн хэвлэгчийн бүртгэлийг хийхийг сурахын тулд Хэвлэгчийн илүү нарийн тохиргоо хэсгийг үзэх.

## 10.3.1. Хэвлэгчийн хялбар тохиргоо

Энэ хэсэг нь хэвлэгчийг ашиглахын тулд хэвлэгчийн тоног төхөөрөмж болон LPD програм хангамжийг хэрхэн тохируулахыг хэлж өгнө. Энэ нь үндсэн ойлголтуудад сургана:

- Тоног төхөөрөмжийн тохиргоо хэсэг нь хэвлэгчийг таны компьютер дээрх порт уруу холбох зарим зааврыг өгдөг.
- Програм хангамжийн тохиргоо хэсэг нь LPD түр хадгалагчийн тохиргооны файлыг (/etc/printcap) хэрхэн тохируулахыг үзүүлдэг.

Хэвлэх өгөгдлийг хүлээн авахдаа компьютерийн локал интерфэйсүүдийг биш сүлжээний протоколыг ашигладаг хэвлэгчийг та тохируулж байгаа бол Сүлжээнд холбогдсон өгөгдлийн урсгалын интерфэйсүүдтэй хэвлэгчид хэсгийг үзнэ үү.

Энэ хэсэг нь "Хэвлэгчийн хялбар тохиргоо" гэгддэг боловч яг үнэндээ нэлээн төвөгтэй юм. Хэвлэгчийг өөрийн компьютер болон LPD түр хадгалагчтай цуг ажиллуулна гэдэг хамгийн хэцүү хэсэг юм. Толгой хуудаснууд болон бүртгэл хийх зэрэг дэвшилтэт тохируулгууд нь хэвлэгчийг ажиллуулсны дараа нэлээн амархан байдаг.

### 10.3.1.1. Тоног төхөөрөмжийн тохиргоо

Энэ хэсэг нь өөрийн PC-тэй хэвлэгч холбох төрөл бүрийн аргуудын талаар хэлдэг. Энэ нь портууд ба кабелиудын талаар болон FreeBSD-г хэвлэгчтэй харилцдаг болгохын тулд танд хэрэг болох цөмийн тохиргооны талаар бас өгүүлнэ.

Хэрэв та өөрийн хэвлэгчийг аль хэдийн холбосон бөгөөд өөр үйлдлийн систем дээр түүгээр амжилттай хэвлэсэн бол Програм хангамжийн тохиргоо хэсэг уруу алгасаж болох юм.

#### 10.3.1.1.1. Портууд ба кабелиуд

PC дээр ашиглагдах зориулалттай худалдаалж байгаа хэвлэгчид нь ерөнхийдөө доорх гурван интерфэйсийн аль нэг юм уу эсвэл олон интерфэйстэй ирдэг:

- *Цуваа* интерфэйсүүд буюу бас RS-232 эсвэл COM портууд гэгддэг интерфэйсүүд нь өгөгдлийг хэвлэгч уруу илгээхдээ таны компьютер дээрх цуваа портыг ашигладаг. Цуваа интерфэйсүүд нь компьютерийн аж үйлдвэрлэлд түгээмэл байдаг бөгөөд кабелиуд нь бэлэн байж байдгаас гадна хийхэд хялбар байдаг. Цуваа интерфэйсүүд нь заримдаа тусгай кабелиудыг шаарддаг бөгөөд танаас төвөгтэй холбооны сонголтуудыг тохируулахыг шаардаж болох юм. Ихэнх PC-ний цуваа портууд нь хамгийн их дамжуулах хурдны хувьд 115200 bps хурдтай байдаг бөгөөд их том график хэвлэх ажлыг үүгээр хийх нь практикийн хувьд төвөгтэй болгодог.
- *Зэрэгцээ* интерфэйсүүд нь өгөгдлийг хэвлэгч уруу илгээхдээ таны компьютерийн зэрэгцээ портыг ашигладаг. Зэрэгцээ интерфэйсүүд нь PC-ийн зах зээлд нийтлэг байдаг бөгөөд RS-

232 цуваагаас илүү хурдан байдаг. Кабелиуд нь байдаг боловч гараар хийхэд илүү төвөгтэй байдаг. Зэрэгцээ интерфэйсүүдэд холбооны сонголтууд ихэвчлэн байдаггүй болохоор тэдгээрийн тохиргоог маш хялбар болгодог.

Зэрэгцээ интерфэйсүүд нь хэвлэгчийн холбогчийн төрлийн нэрээр нэрлэгдсэн "Centronics" интерфэйсүүд гэж заримдаа хэлэгддэг.

- Universal Serial Bus буюу Универсал Цуваа Шугам гэгддэг USB интерфэйсүүд нь зэрэгцээ болон RS-232 цуваа интерфэйсүүдээс бүр илүү хурдаар ажиллаж чаддаг. Кабелиуд нь хялбар бөгөөд хямд байдаг. USB нь RS-232 болон Зэрэгцээ интерфэйсүүдээс хэвлэхийн хувьд илүү хүчирхэг боловч UNIX® системүүд дээр тийм ч сайн дэмжигдсэн байдаггүй. Энэ асуудлыг тойрон гарахын тулд олон хэвлэгчид байдаг USB болон Зэрэгцээ интерфэйсүүдтэй тийм хэвлэгч худалдан авах явдал юм.

Ерөнхийдөө зэрэгцээ интерфэйсүүд нь зөвхөн нэг талын холбоог (компьютераас хэвлэгч уруу) санал болгодог бол цуваа болон USB нь хоёр талын холбоог өгдөг. Шинэ зэрэгцээ портууд (EPP болон ECP) болон хэвлэгчид нь IEEE-1284 нийцтэй кабель ашиглаж байгаа бол FreeBSD дээр хоёр тал уруу чиглэсэн холбоог хийж чаддаг.

Зэрэгцээ портоор хэвлэгч уруу чиглэсэн хоёр талын холбоог ерөнхийдөө хоёр аргын аль нэгээр хийдэг. Эхний арга нь хэвлэгчийн ашигладаг хувийн хэлээр ярьдаг FreeBSD-ийн өөрчлөн бүтээсэн хэвлэгчийн драйверийг хэрэглэдэг. Энэ нь бэхэн хэвлэгчдэд нийтлэг байдаг бөгөөд бэхний түвшин болон бусад төлвийн мэдээллийг үзүүлэхэд хэрэглэгддэг. Хоёр дахь аргыг хэвлэгч PostScript®-г дэмждэг үед ашигладаг.

PostScript® ажлууд нь жинхэнэдээ хэвлэгч уруу илгээсэн програмууд юм; тэдгээр нь цаас бүтээх ерөөсөө шаардлагагүй бөгөөд үр дүнгүүдийг компьютер уруу шууд буцааж болох юм. PostScript® нь PostScript® програм дахь алдаанууд, эсвэл цаас гацсан зэрэг асуудлуудыг компьютерт хэлэхийн тулд хоёр талын холбоог бас ашигладаг. Таны хэрэглэгчид тийм мэдээллийг аваад талархах байх. Бас PostScript® хэвлэгчийн хувьд үр ашигтай бүртгэл хийх хамгийн шилдэг арга нь хоёр талын холбоог шаарддаг: хэвлэгчээс хуудасны тоог (өөрийн амьдралын хугацаандаа нийт хичнээн хуудас хэвлэсэн талаар) та асууж дараа нь хэрэглэгчийн ажлыг илгээж тэгээд дахиад хуудасны тоог асуудаг. Хоёр уттын нэгээс нөгөөг хасаад хэрэглэгчээс хичнээн цаасны төлбөр авахыг та мэдэх болно.

#### 10.3.1.1.2. Зэрэгцээ портууд

Хэвлэгчийг зэрэгцээ интерфэйс ашиглаад залгахын тулд Centronics кабелийг хэвлэгч болон компьютерийн хооронд холбоно. Хэвлэгч, компьютер эсвэл хоёулантай нь цуг ирсэн зааврууд танд бүрэн гүйцэд заавар өгөх ёстой.

Компьютер дээрээ аль зэрэгцээ портыг ашигласнаа санах хэрэгтэй. Эхний зэрэгцээ порт нь FreeBSD-д ppc0 байх бөгөөд хоёр дахь ppc1 гэх мэтээр байна. Хэвлэгчийн төхөөрөмжийн нэр мөн адил схемийг ашигладаг: /dev/lpt0 нь эхний зэрэгцээ порт дээрх хэвлэгч гэх мэтээр байна.

#### 10.3.1.1.3. Цуваа портууд

Цуваа интерфэйсүүдийг ашиглан хэвлэгчийг залгахдаа зөв цуваа кабелийг хэвлэгч болон компьютерийн хооронд холбоно. Хэвлэгч, компьютер эсвэл хоёулантай нь цуг ирсэн

зааврууд танд бүрэн гүйцэд заавар өгөх ёстой.

Хэрэв та "зөв цуваа кабель" нь юу вэ гэдэгт эргэлзэж байвал та дараах боломжуудаас аль нэгийг туршиж үзэхийг хүсэж болох юм:

- *Модемийн* кабель нь кабелийн нэг талд байгаа холбогчийн зүү бүрийг нөгөө талд байгаа холбогчийн харгалзах зүүнд шулуухан холбодог. Энэ төрлийн кабелийг "DTE-ээс DCE" кабель гэж бас нэрлэдэг.
- *null-модем* кабель нь зарим зүүнүүдийг шууд шулуухнаар, заримуудыг нь хооронд нь солбидог (жишээ нь өгөгдлийг хүлээн авахын тулд өгөгдлийг илгээдэг) бөгөөд заримыг нь дотроо холбогч бүрийн таг дотор богино холбодог. Энэ төрлийн кабелийг "DTE-ээс DTE" кабель гэж бас нэрлэдэг.
- Зарим нэг ховор хэвлэгчдэд шаардагддаг *цуваа хэвлэгчийн* кабель нь null-модем кабельтай адил боловч дотроо богино холбохын оронд зарим дохионуудыг өөрсдийн эсрэг талдаа илгээдэг.

Та хэвлэгчийн хувьд холбооны параметрүүдийг бас тохируулах хэрэгтэй бөгөөд эдгээрийг хэвлэгч дээрх нүүрэн талын хяналтууд юм уу эсвэл DIP шилжүүлэгчдээр хийдэг. Таны компьютер болон хэвлэгч дэмждэг хамгийн их bps-ийг (bits per second буюу секундэд дамжих битийн тоо, заримдаа *baud хурд*) сонгох хэрэгтэй. 7 юм уу эсвэл 8 өгөгдлийн бит; байхгүй (none), тэгш (even), эсвэл сондгой (odd) parity; болон 1 эсвэл 2 стоп битийг сонгох хэрэгтэй. Бас урсгал хянах протоколыг сонгох хэрэгтэй: байхгүй (none) юм уу аль эсвэл XON/XOFF (бас "in-band" эсвэл "software" гэгддэг) урсгал хяналтыг сонгох хэрэгтэй. Дараах програм хангамжийн тохиргоонд зориулж эдгээр тохируулгуудыг санах хэрэгтэй.

#### 10.3.1.2. Програм хангамжийн тохиргоо

Энэ хэсэг нь FreeBSD дээр LPD түр хадгалах системээр хэвлэхэд шаардлагатай програм хангамжийн тохируулгыг тайлбарладаг.

Энд шаардлагатай алхмуудыг дурдав:

1. Хэвлэгчийн хувьд ашиглаж байгаа портод зориулж шаардлагатай бол өөрийн цөмийг тохируулна; Таны юу хийх ёстойг Цөмийн тохиргоо хэсэг хэлж өгнө.
2. Хэрэв та зэрэгцээ порт ашиглаж байгаа бол зэрэгцээ портынхоо хувьд холбооны горимыг тохируулах хэрэгтэй; Зэрэгцээ портын хувьд холбооны горимыг тохируулах нь хэсгээс дэлгэрэнгүйг үзээрэй.
3. Үйлдлийн систем өгөгдлийг хэвлэгч уруу илгээж чадаж байгаа эсэхийг тест хийнэ. Үүнийг хэрхэн хийх тухай зарим нэгэн зөвлөгөөнүүдийг Хэвлэгчийн холбоонуудыг шалгах нь хэсэг өгдөг.
4. /etc/printcap файлыг засаж хэвлэгчдээ зориулж тохируулна. Үүнийг хэрхэн хийх талаар сүүлд нь энэ бүлгээс олох болно.

##### 10.3.1.2.1. Цөмийн тохиргоо

Үйлдлийн системийн цөм тусгайлсан хэдэн төхөөрөмжүүдтэй ажиллахаар эмхэтгэгдсэн

байдаг. Таны хэвлэгчид зориулагдсан цуваа болон зэрэгцээ интерфэйс нь эдгээрийн нэг хэсэг юм. Тийм болохоор хэрэв таны цөм ингэж тохируулагдаагүй бол нэмэлт цуваа эсвэл зэрэгцээ портын дэмжлэгийг нэмэх хэрэгтэй байж болох юм.

Таны ашиглаж байгаа цөм цуваа интерфэйсийг дэмжиж байгаа эсэхийг мэдэхийн тулд доор дурдсаныг бичнэ:

```
# grep sioN /var/run/dmesg.boot
```

Энд байгаа *N* нь тэгээс эхлэх цуваа портын дугаар юм. Хэрэв та доор дурдсантай төстэй гаралтыг харвал:

```
sio2 at port 0x3e8-0x3ef irq 5 on isa
sio2: type 16550A
```

цөм нь портыг дэмждэг гэсэн үг юм.

Цөм нь зэрэгцээ интерфэйсийг дэмждэг эсэхийг мэдэхийн тулд доор дурдсаныг бичнэ:

```
# grep ppcN /var/run/dmesg.boot
```

Энд байгаа *N* нь тэгээс эхлэх зэрэгцээ портын дугаар юм. Хэрэв та доор дурдсантай төстэй гаралтыг харвал:

```
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
ppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/8 bytes threshold
```

цөм нь портыг дэмждэг гэсэн үг юм.

Таны хэвлэгчдээ зориулан ашиглаж байгаа зэрэгцээ эсвэл цуваа портыг үйлдлийн систем таньж хэрэглэдэг байхын тулд та өөрийн цөмийг дахин тохируулах хэрэгтэй болж болох юм.

Цуваа портод зориулж дэмжлэг нэмэхийн тулд цөмийн тохиргооны тухай хэсгийг үзнэ үү. Зэрэгцээ портод зориулж дэмжлэг нэмэхийн тулд тэр хэсгийг *болон* дараагийн хэсгийг үзнэ үү.

#### 10.3.1.3. Зэрэгцээ портын хувьд холбооны горимыг тохируулах нь

Зэрэгцээ интерфэйсийг ашиглаж байхдаа FreeBSD нь хэвлэгчийн хувьд тасалдлаар зохицуулагдах юм уу эсвэл байнга шалгах холбооны алийг ашиглахыг та сонгож болно. FreeBSD дээрх ердийн хэвлэгчийн төхөөрөмжийн драйвер (lpt(4)) нь портын бичил схемийг ppc(4) драйвераар хянадаг ppbus(4) системийг ашигладаг.

- *interrupt-driven буюу тасалдлаар зохицуулагдах* арга нь GENERIC цөмд анхдагч байдаг. Энэ арганд үйлдлийн систем нь хэвлэгч өгөгдөл хүлээн авахад хэзээ бэлэн байгааг

тодорхойлохын тулд IRQ шугамыг ашигладаг.

- *polled* буюу байнга шалгах арга нь үйлдлийн системд хэвлэгчээс өгөгдөл хүлээн авахад бэлэн байгаа эсэхийг нь давталттайгаар асуухыг зааж өгдөг. Бэлэн гэж хариулбал цөм илүү өгөгдөл илгээдэг.

Тасалдлаар зохицуулагдах арга нь ихэвчлэн хурдан байдаг боловч нандин IRQ шугамыг ихээр ашигладаг. Зарим шинэ HP хэвлэгчид нь зарим (яг тодорхой ойлгогдоогүй) хугацааны асуудлуудаас болоод тасалдлын горимд зөв ажилладаггүй гэгддэг. Эдгээр хэвлэгчдэд байнга шалгах горим хэрэгтэй. Аль ажиллаж байгааг нь л та ашиглах хэрэгтэй. Зарим хэвлэгчид нь аль алин дээр нь ажилладаг боловч тасалдлын горим асар удаан байдаг.

Та холбооны горимыг хоёр аргаар тохируулж болно: цөмийг тохируулах замаар эсвэл lptcontrol(8) програм ашиглан хийж болно.

*Цөмийг тохируулах замаар холбоог тохируулахдаа:*

1. Өөрийн цөмийн тохиргооны файлаа засварлана. `ppc0` оруулгыг хайна. Хэрэв та хоёр дахь зэрэгцээ портыг тохируулж байгаа бол `ppc1`-г ашиглаарай. Гурав дахь портын хувьд `ppc2` гэх мэтээр ашиглана.
   - Хэрэв та тасалдлаар зохицуулагдах горимыг хүсэж байгаа бол дараах мөрийг:

     ```
     hint.ppc.0.irq="N"
     ```

     /boot/device.hints файлд засварлаж *N*-ий оронд зөв IRQ дугаарыг солих хэрэгтэй. Цөмийн тохиргооны файл ppc(4) драйверийг бас агуулсан байх ёстой:

     ```
     device ppc
     ```

   - Хэрэв та байнга шалгадаг горимыг хүсэж байвал өөрийн /boot/device.hints файлаас дараах мөрийг арилгана:

     ```
     hint.ppc.0.irq="N"
     ```

     Зарим тохиолдолд энэ нь FreeBSD дээр портыг байнга шалгах горимд оруулахад хангалттүй байдаг. Ихэнхдээ энэ нь acpi(4) драйвераас гардаг бөгөөд энэ нь төхөөрөмжүүдийг шалгаж болон залгаж чаддаг болохоор хэвлэгчийн порт уруу хандах горимыг хянаж чаддаг байна. Энэ асуудлыг засварлахын тулд та өөрийн acpi(4) тохиргоог шалгах хэрэгтэй.
2. Файлыг хадгална. Дараа нь цөмд тохиргоо хийн бүтээж цөмийг суулгаад дахин ачаална. Илүү дэлгэрэнгүйг цөмийн тохиргоо хэсгээс үзнэ үү.

*Холбооны горимыг* lptcontrol(8)-р тохируулахын тулд:

1. Доор дурдсаныг бичиж:

```
# lptcontrol -i -d /dev/lptN
```

   `lptN`-д зориулж тасалдлаар зохицуулагдах горимыг тохируулна.

2. Доор дурдсаныг бичиж:

```
# lptcontrol -p -d /dev/lptN
```

   `lptN`-д зориулж байнга шалгах горимыг тохируулна.

Эдгээр тушаалуудыг өөрийн /etc/rc.local файлд нэмж таны системийг ачаалах болгонд горимыг тохируулдаг байхаар хийж болох юм. Илүү мэдээллийг lptcontrol(8)-с үзнэ үү.

#### 10.3.1.4. Хэвлэгчийн холбоог шалгах нь

Түр хадгалагч системийг тохируулахаасаа өмнө та үйлдлийн системийг өгөгдлийг амжилттайгаар өөрийн хэвлэгч уруу илгээж чадаж байгаа эсэхийг шалгах хэрэгтэй. Хэвлэгчийн холбоо болон түр хадгалагч системийг тусад нь дибаг хийх нь хамаагүй хялбар байдаг.

Хэвлэгчийг тест хийхийн тулд бид ямар нэг текст түүн үрүү илгээнэ. Илгээгдсэн тэмдэгтүүдийг тэр даруй нь хэвлэх хэвлэгчдийн хувьд lptest(1) програм төгс байдаг: энэ нь бүх 96 хэвлэх боломжтой ASCII тэмдэгтүүдийг 96 мөрөнд үүсгэдэг.

PostScript® (эсвэл бусад хэлэн дээр суурилсан) хэвлэгчийн хувьд бидэнд илүү төвөгтэй тест хэрэгтэй. Доор дурдсантай төстэй жижиг PostScript® програм хангалттай байх болно:

```
%!PS
100 100 moveto 300 300 lineto stroke
310 310 moveto /Helvetica findfont 12 scalefont setfont
(Is this thing working?) show
showpage
```

Дээрх PostScript® кодыг файлд хийгээд доорх хэсгүүдэд гарч байгаа жишээнүүдэд үзүүлсэн шигээр ашиглаж болно.

Энэ баримт нь хэвлэгчийн хэл гэдгийг Hewlett Packard-ийн PCL биш харин PostScript®-тэй адил хэлийг хэлж байгаа юм. PCL нь мундаг ажиллагаатай боловч та цэвэр текстийг өөрийнх нь escape (зугтах) дарааллуудтай нь хольж болдог. PostScript® нь цэвэр текстийг шууд хэвлэж чаддаггүй бөгөөд энэ нь тусгай зохицуулалтууд хийж өгөх ёстой тийм хэвлэгчийн хэл юм.

#### 10.3.1.4.1. Зэрэгцээ хэвлэгчийг шалгах нь

Энэ хэсэг нь зэрэгцээ порт уруу холбогдсон хэвлэгчтэй FreeBSD холбогдож чадах эсэхийг хэрхэн шалгахыг хэлж өгөх болно.

*Зэрэгцээ порт дээрх хэвлэгчийг тест хийхийн тулд:*

1. su(1) ашиглан `root` болно.
2. Хэвлэгч уруу өгөгдөл илгээнэ.
   - Хэрэв хэвлэгч цэвэр текст хэвлэж чаддаг бол lptest(1)-г ашиглана. Доор дурдсаныг бичнэ:

     ```
     # lptest > /dev/lptN
     ```

     Энд байгаа *N* нь тэгээс эхлэх зэрэгцээ портын дугаар юм.
   - Хэрэв хэвлэгч PostScript® эсвэл бусад хэвлэгчийн хэлийг ойлгодог бол жижиг програм хэвлэгч уруу илгээх хэрэгтэй. Доор дурдсаныг бичнэ:

     ```
     # cat > /dev/lptN
     ```

     Тэгээд мөр мөрөөр програмыг анхааралтай оруулах хэрэгтэй. Учир нь `RETURN` эсвэл `ENTER` дарсны дараа та засварлаж чадахгүй болох юм. Програмаа оруулж дуусаад `CONTROL+D` юм уу эсвэл файлын төгсгөл товчлуур ямар байна тэрийг дарах хэрэгтэй.

     Мөн өөрөөр програмыг файлд хийж доор дурдсаныг бичин өгч болно:

     ```
     # cat file > /dev/lptN
     ```

     Энд байгаа *file* нь таны илгээхийг хүсэж байгаа програмыг агуулсан файлын нэр юм.

Та ямар нэгэн зүйл харах ёстой. Текст зөв харагдахгүй байвал санаа зовсны хэрэггүй; бид иймэрхүү зүйлсийг сүүлд нь засах болно.

#### 10.3.1.4.2. Цуваа хэвлэгчийг шалгах нь

Энэ хэсэг нь цуваа порт уруу холбогдсон хэвлэгчтэй FreeBSD холбогдож чадах эсэхийг хэрхэн шалгахыг хэлж өгөх болно.

*Цуваа порт дээрх хэвлэгчийг тест хийхдээ:*

1. su(1) ашиглан `root` болно.

2. /etc/remote файлыг засварлана. Дараах мөрийг нэмнэ:

```
printer:dv=/dev/port:br#bps-rate:pa=parity
```

Энд байгаа *port* нь цуваа портод (`ttyu0`, `ttyu1`, гэх мэт) зориулсан төхөөрөмжийн оруулга, *bps-rate* нь хэвлэгчийн холбогдох секундэд илгээх битийн хурд, болон *parity* нь хэвлэгчийн шаарддаг parity юм (`even`, `odd`, `none`, эсвэл `zero`).

Цуваа шугамаар гурав дахь цуваа порт уруу 19200 bps хурдаар parity байхгүйгээр холбогдсон хэвлэгчид зориулсан жишээ оруулга энд байна:

```
printer:dv=/dev/ttyu2:br#19200:pa=none
```

3. tip(1) ашиглан хэвлэгч уруу холбогдоно. Доор дурдсаныг бичнэ:

```
# tip printer
```

Хэрэв энэ алхам нь ажиллахгүй бол /etc/remote файлыг дахин засварлаж /dev/ttyuN -ийн оронд /dev/cuaaN-г ашиглаж үзээрэй.

4. Өгөгдлийг хэвлэгч уруу илгээнэ.
   - Хэрэв хэвлэгч цэвэр текст хэвлэж чаддаг бол lptest(1)-г ашиглана. Доор дурдсаныг бичнэ:

     ```
     % $lptest
     ```

   - Хэрэв хэвлэгч PostScript® эсвэл бусад хэвлэгчийн хэлийг ойлгодог бол жижиг програм хэвлэгч уруу илгээх хэрэгтэй. Програмыг мөр мөрөөр *маш анхааралтайгаар* бичиж оруулах хэрэгтэй. Учир нь арилгах болон бусад засварлах товчлуурууд хэвлэгчийн хувьд чухал байж болох юм. Та програмыг бүгдийг нь хүлээж авсан гэж хэвлэгчид таниулахын тулд тусгай файлын төгсгөл товчлуурыг хэвлэгчид зориулж оруулах хэрэгтэй. PostScript®-ийн хэвлэгчдийн хувьд `CONTROL+D` дарна.

     Мөн өөрөөр та програмыг файлд хийж доор дурдсаныг бичин оруулж болно:

     ```
     % >file
     ```

     Энд байгаа *file* нь програмыг агуулж байгаа файлын нэр юм. tip(1) файлыг илгээсний дараа шаардлагатай файлын төгсгөл товчлуурыг дарах хэрэгтэй.

Та ямар нэгэн зүйл хэвлэгдэхийг харах ёстой. Текст зөв харагдахгүй байвал санаа зовсны хэрэггүй; бид иймэрхүү зүйлсийг сүүлд нь засах болно.

#### 10.3.1.5. Түр хадгалагчийг идэвхжүүлэх нь: /etc/printcap файл

Энэ үед таны хэвлэгч залгагдаж таны цөм түүнтэй холбогдохоор тохируулагдсан (хэрэв шаардлагатай бол) бөгөөд та хэвлэгч уруугаа зарим энгийн өгөгдлийг илгээж чаддаг байна. Одоо бид таны хэвлэгч уруу хандах хандалтыг хянахын тулд LPD-г тохируулахад бэлэн боллоо.

Та /etc/printcap файлыг засварлаж LPD-г тохируулна. LPD түр хадгалагч систем нь түр хадгалагч ашиглагдах болгонд энэ файлыг уншдаг учир энэ файл уруу хийгдсэн шинэчлэл шууд нөлөөлөх болно.

printcap(5) файлын хэлбэршилт хялбархан. /etc/printcap файлд засвар хийхдээ өөрийн дуртай текст засварлагчийг ашиглах хэрэгтэй. Хэлбэршилт нь /usr/shared/misc/termcap болон /etc/remote зэрэг бусад боломжийн файлуудтай төстэй байдаг. Хэлбэршилтийн тухай бүрэн мэдээллийг cgetent(3)-с үзнэ үү.

Энгийн түр хадгалагчийн тохиргоо дараах алхмуудаас тогтоно:

1. Хэвлэгчид зориулж нэр (болон хэдэн тохиромжтой alias-уудыг) сонгож /etc/printcap файлд хийж өгнө; Нэрлэх тухай дэлгэрэнгүй мэдээллийг Хэвлэгчийг нэрлэх нь хэсгээс үзнэ үү.
2. `sh` боломжийг оруулан толгой хуудаснуудыг болиулах хэрэгтэй (анхдагчаар идэвхтэй байдаг); Дэлгэрэнгүй мэдээллийг Толгой хуудаснуудыг дарах нь хэсгээс үзнэ үү.
3. Түр хадгалах санг үүсгэж түүний байрлалыг `sd` боломжоор зааж өгөх хэрэгтэй; Дэлгэрэнгүй мэдээллийг Түр хадгалах санг үүсгэх нь хэсгээс үзнэ үү.
4. Хэвлэгчид зориулж ашиглахаар /dev оруулгыг тохируулж `lp` боломжоор /etc/printcap файлд түүнийг тэмдэглэх хэрэгтэй; Дэлгэрэнгүй мэдээллийг Хэвлэгчийн төхөөрөмжийг таних нь хэсгээс үзнэ үү. Хэрэв хэвлэгч цуваа порт дээр байгаа бол холбооны параметрүүдийг Түр хадгалагчийн холбооны параметрүүдийг тохируулах нь хэсэгт хэлэлцсэн `ms#` боломжоор тохируулах хэрэгтэй.
5. Цэвэр текст оруулах шүүлтүүрийг суулгана; Дэлгэрэнгүйг Текст шүүгчийг суулгах нь хэсгээс үзнэ үү.
6. lpr(1) тушаалаар ямар нэг юм хэвлэж тест хийнэ. Илүү дэлгэрэнгүйг Турших нь болон Алдааг олж засварлах хэсгүүдээс үзнэ үү.

PostScript® хэвлэгчид зэрэг хэлэн дээр суурилсан хэвлэгчид цэвэр текстийг шууд хэвлэж чаддаггүй. Дээр болон дараагийн хэсгүүдэд тайлбарласан энгийн тохиргоо нь хэрэв та ийм хэвлэгч суулгаж байгаа бол таныг зөвхөн хэвлэгчийн ойлгодог файлыг хэвлэнэ гэж тооцдог.

Хэрэглэгчид нь цэвэр текстийг таны системд суулгагдсан дурын хэвлэгч уруу өгч хэвлэж болно гэж ихэвчлэн боддог. LPD уруу залгагдсан програмууд өөрсдийн хэвлэлтийг хийхдээ бас ингэж тооцдог. Хэрэв та тийм хэвлэгч суулгаж хэвлэгчийн хэл дээр ажлуудаа хэвлэх *болон* цэвэр текст ажлуудыг хэвлэж чаддаг байхыг хүсэж байгаа бол дээр дурдсан энгийн

тохиргоонд нэмэлт алхам нэмж өгөх зайлшгүй шаардлагатай: цэвэр текстээс PostScript® уруу автоматаар хөрвүүлэх програмыг суулгах хэрэгтэй. Цэвэр текст ажлуудыг PostScript® хэвлэгчид дээр тааруулах нь гэж нэрлэгдсэн хэсэг үүнийг хэрхэн хийхийг хэлж өгнө.

#### 10.3.1.5.1. Хэвлэгчийг нэрлэх нь

Эхний (хялбар) алхам бол өөрийн хэвлэгчид зориулж нэр сонгох явдал юм. Ажиллагаатай нь холбогдуулж юм уу эсвэл этгээд чамин нэр сонгох нь яг үнэндээ хамаагүй юм. Учир нь та хэвлэгчид зориулж бас хэд хэдэн alias буюу өөр нэрсийг өгч болох юм.

/etc/printcap-д тодорхойлогдсон хэвлэгчдийн хамгийн багаар бодоход аль нэг `lp` гэсэн alias-тай байх ёстой. Энэ нь анхдагч хэвлэгчийн нэр юм. Хэрэв хэрэглэгчдэд `PRINTER` орчны хувьсагч байхгүй юм уу эсвэл LPD тушаалуудыг өгөх тушаалын мөрөнд хэвлэгчийн нэрийг зааж өгөөгүй бол `lp` нь тэдгээрийн ашиглаж сурсан анхдагч хэвлэгч болох юм.

Хэвлэгчийн хамгийн сүүлийн alias-д хэвлэгчийн бүрэн тайлбарыг үйлдвэрлэгч болон загварыг оруулан өгөх нь нийтлэг практик байдаг.

Нэр болон зарим нийтлэг alias-уудыг сонгосныхоо дараа тэдгээрийг /etc/printcap файлд хийж өгөх хэрэгтэй. Хэвлэгчийн нэр хамгийн зүүн баганаас эхлэх ёстой. Alias болгоныг босоо шугамаар тусгаарлаж хамгийн сүүлийн alias-ийн ард тодорхойлох цэг тавьна.

Дараах жишээн дээр хоёр хэвлэгчийг (Diablo 630 шугамын хэвлэгч болон Panasonic KX-P4455 PostScript® лазерийн хэвлэгч) тодорхойлох /etc/printcap файлаас бид эхлэх болно:

```
#
#  /etc/printcap for host rose
#
rattan|line|diablo|lp|Diablo 630 Line Printer:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:
```

Энэ жишээн дээр эхний хэвлэгч нь `rattan` гэж нэрлэгдсэн бөгөөд `line`, `diablo`, `lp`, болон `Diablo 630 Line Printer` гэсэн alias-уудтай байна. Энэ нь `lp` alias-тай болохоор энэ нь бас анхдагч хэвлэгч юм. Хоёр дахь нь `bamboo` гэж нэрлэгдсэн бөгөөд `ps`, `PS`, `S`, `panasonic`, болон `Panasonic KX-P4455 PostScript v51.4` гэсэн alias-уудтай байна.

#### 10.3.1.5.2. Толгой хуудаснуудыг дарах нь

LPD түр хадгалах систем нь ажил бүрийн хувьд анхдагчаар *толгой хуудас* хэвлэдэг. Толгой хуудас нь том сайхан үсгээр бичигдсэн ажлыг хүссэн хэрэглэгчийн нэр, ажил ирсэн хост болон ажлын нэрийг агуулдаг. Харамсалтай нь энэ бүх нэмэлт текст нь энгийн хэвлэгчийн тохиргоог дибаг хийх замд саад болдог учир бид толгой хуудаснуудыг дарж хаах болно.

Толгой хуудаснуудыг дарахын тулд /etc/printcap файл дахь хэвлэгчийн оруулгад `sh` боломжийг нэмэх хэрэгтэй. Энд `sh` нэмсэн /etc/printcap файлын жишээ байна:

```
#
#  /etc/printcap for host rose - no header pages anywhere
```

```
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:
```

Бидний зөв хэлбэршилт хэрхэн хэрэглэснийг хараарай: эхний мөр нь хамгийн зүүн талын баганаас эхэлж дараа дараачийн мөрүүд нь хойш зай авагдсан байна. Оруулга дахь сүүлийн мөрөөс бусад мөр бүр урагш налуу тэмдэгтээр төгссөн байна.

#### 10.3.1.5.3. Түр хадгалах санг үүсгэх нь

Түр хадгалагчийн энгийн тохиргоон дахь дараагийн алхам нь хэвлэгдэх ажлууд хэвлэгдэх хүртлээ байдаг, түр хадгалагчийн бусад дэмжлэгийн файлууд бас байдаг *түр хадгалах сан*г үүсгэх явдал юм.

Түр хадгалах сангийн хувьсамтгайгаас болоод эдгээр сангуудыг /var/spool-д хийх нь зуршил болсон зүйл юм. Түр хадгалах сангуудын агуулгыг заавал нөөцлөх ч бас шаардлагагүй юм. Тэдгээрийг үүсгэх нь mkdir(1) тушаалыг ажиллуулахтай адил хялбар юм.

Доор үзүүлсэн шиг хэвлэгчийн нэртэй адил нэрээр санг үүсгэх нь бас зуршил болсон байдаг:

```
# mkdir /var/spool/printer-name
```

Гэхдээ та сүлжээндээ их олон хэвлэгчидтэй бол түр хадгалах сангуудаа зөвхөн LPD-ээр хэвлэхийн тулд нөөцөлсөн нэг сангийн доор хийхийг хүсэж болох юм. Бид `rattan` болон `bamboo` гэсэн хоёр жишээ хэвлэгчдээ зориулж үүнийг хийх болно:

```
# mkdir /var/spool/lpd
# mkdir /var/spool/lpd/rattan
# mkdir /var/spool/lpd/bamboo
```

Хэрэглэгчдийн хэвлэх ажлын нууцлалын талаар та бодож байгаа бол түр хадгалах санд олон нийт хандах боломжгүй болгож түүнийг хамгаалахыг хүсэж болох юм. Түр хадгалах сангуудыг `daemon` хэрэглэгч болон `daemon` бүлэг эзэмшиж, өөр хэнээр ч биш зөвхөн эдгээрээр уншигдах, бичигдэх, болон хайлт хийгдэх боломжтой байх ёстой. Бид үүнийг өөрсдийн жишээ хэвлэгчдэд зориулж хийх болно:

```
# chown daemon:daemon /var/spool/lpd/rattan
# chown daemon:daemon /var/spool/lpd/bamboo
# chmod 770 /var/spool/lpd/rattan
# chmod 770 /var/spool/lpd/bamboo
```

Төгсгөлд нь та эдгээр сангуудын талаар LPD програмд /etc/printcap файл ашиглан хэлж өгөх

хэрэгтэй. Түр хадгалах сангийн замыг `sd` боломжтой цуг зааж өгнө:

```
#
#  /etc/printcap for host rose - added spooling directories
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
    :sh:sd=/var/spool/lpd/rattan:
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
    :sh:sd=/var/spool/lpd/bamboo:
```

Хэвлэгчийн нэр эхний баганаас эхэлж харин хэвлэгчийг тайлбарлах бусад бүх оруулгууд догол мөрөөс эхэлж мөр бүрийн төгсгөл урагш налуу тэмдэгтээр төгсөх ёстой.

Хэрэв та түр хадгалах санг `sd`-ээр зааж өгөхгүй бол түр хадгалах систем нь анхдагчаар /var/spool/lpd-г ашиглах болно.

#### 10.3.1.5.4. Хэвлэгч төхөөрөмжийг таних нь

Тоног төхөөрөмжийн тохиргоо хэсэгт бид порт болон тохирох /dev санг FreeBSD нь хэвлэгчтэй холбогдохдоо ашиглахыг мэдлээ. Одоо бид LPD-д тэр мэдээллийг хэлнэ. Түр хадгалах систем нь хэвлэх ажилтай байгаа үедээ шүүгч програмын өмнөөс заасан төхөөрөмжийг онгойлгох болно (шүүгч програм нь өгөгдлийг хэвлэгч уруу дамжуулахыг хариуцдаг).

`lp` боломж ашиглан /etc/printcap файлд /dev оруулгын замыг жагсааж өгнө.

Бидний ажиллах жишээн дээр `rattan` нь эхний зэрэгцээ порт дээр, `bamboo` нь зургаа дахь цуваа порт дээр байгаа гэж үзье; энд /etc/printcap файлд хийгдэх нэмэлтүүд байна:

```
#
#  /etc/printcap for host rose - identified what devices to use
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan:\
        :lp=/dev/lpt0:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:\
        :lp=/dev/ttyu5:
```

Хэрэв та өөрийн /etc/printcap файл дээрээ хэвлэгчдээ зориулж `lp` боломжийг зааж өгөхгүй бол LPD нь анхдагчаар /dev/lp-г ашиглах болно. /dev/lp нь одоогоор FreeBSD дээр байхгүй байгаа.

Хэрэв таны суулгаж байгаа хэвлэгч зэрэгцээ порт уруу холбогдсон бол Текст шүүгчийг суулгах нь хэсэг уруу алгасаарай. Үгүй бол дараагийн хэсэг дэх зааврыг дагахаа битгий мартаарай.

#### 10.3.1.5.5. Түр хадгалагчийн холбооны параметрүүдийг тохируулах нь

Зэрэгцээ портууд дээрх хэвлэгчдийн хувьд LPD нь bps хурд, parity, болон бусад цуваа холбооны параметрүүдийг хэвлэгч уруу өгөгдөл илгээдэг шүүгч програмын өмнөөс тохируулж чаддаг. Энэ нь давуу талуудтай, учир нь:

- Энэ нь /etc/printcap файлыг засварлан өөр холбооны параметрүүдийг турших боломжийг танд олгодог; та шүүгч програмыг дахин эмхэтгэх шаардлагагүй.
- Энэ нь өөр өөр цуваа холбооны тохируулгуудтай байж болох олон хэвлэгчдэд зориулж нэг шүүгч програмыг ашиглахаар болгож түр хадгалах системийг идэвхжүүлдэг.

Дараах /etc/printcap боломжууд нь `lp` боломжид жагсаагдсан төхөөрөмжийн цуваа холбооны параметрүүдийг хянадаг:

`br#bps-rate`

: Төхөөрөмжийн холбооны хурдыг *bps-rate* болгож тохируулах бөгөөд *bps-rate* нь секундэд 50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, эсвэл 115200 бит байж болно.

`ms#stty-mode`

: Төхөөрөмжийг онгойлгосны дараа терминал төхөөрөмжид зориулж сонголтуудыг тохируулна. stty(1) нь байж болох сонголтуудыг тайлбарладаг.

`lp` боломжоор заагдсан төхөөрөмжийг LPD онгойлгох үед энэ нь төхөөрөмжийн үзүүлэлтүүдийг `ms#` боломжоор заагдсанаар тохируулдаг. Тухайлбал, stty(1) гарын авлагын хуудсанд тайлбарлагдсан байгаа `parenb`, `parodd`, `cs5`, `cs6`, `cs7`, `cs8`, `cstopb`, `crtscts`, болон `ixon` горимууд юм.

Зургаа дахь цуваа порт дээр өөрсдийн жишээ хэвлэгчээ нэмье. Бид bps хурдыг 38400 гэж тохируулна. Горимын хувьд бид `-parenb` тохируулгаар parity байхгүй, `cs8` тохируулгаар 8 бит тэмдэгтүүд, `clocal` тохируулгаар модемийн хяналт байхгүй, `crtscts` тохируулгаар тоног төхөөрөмжийн урсгалын хяналт байхгүй гэж тохируулах болно:

```
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:\
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:
```

#### 10.3.1.5.6. Текст шүүгчийг суулгах нь

Одоо бид хэвлэгч уруу ажлууд илгээхийн тулд ямар текстийн шүүгчийг ашиглахыг LPD-д хэлэхэд бэлэн боллоо. *Оролтын шүүгч* гэж бас нэрлэгддэг *Текст шүүгч* нь хэвлэх ажил байхад LPD-ийн ажиллуулдаг програм юм. LPD нь хэвлэгчид зориулж текст шүүгчийг ажиллуулахдаа шүүгчийн стандарт оролтыг хэвлэх ажил уруу тохируулж өгдөг бөгөөд стандарт гаралтыг `lp` боломжоор заагдсан хэвлэгчийн төхөөрөмж уруу тохируулдаг. Шүүгч нь стандарт оролтоос ажлыг уншиж хэвлэгчид зориулж шаардлагатай орчуулгыг хийж үр дүнг стандарт гаралт уруу бичихээр хийгдсэн байдаг. Ингэснээр хэвлэгдэх болно. Текст шүүгчийн тухай дэлгэрэнгүй мэдээллийг Шүүгчид хэсгээс үзнэ үү.

Бидний энгийн хэвлэгчийн тохиргооны хувьд текст шүүгч нь хэвлэгч уруу ажлыг

илгээхийн тулд `/bin/cat` тушаалыг ажиллуулах жижиг бүрхүүлийн скрипт байж болох юм. FreeBSD нь устгах болон доогуур зурах зэрэг тэмдэгтийн урсгалуудтай сайн ажиллаж чаддаггүй хэвлэгчдэд зориулсан тэдгээр тэмдэгтүүдийг зохицуулдаг lpf гэсэн өөр шүүгчтэй ирдэг. Тэгээд мэдээж та хүссэн өөр бусад шүүгч програмыг ашиглаж болно. `lpf` шүүгч нь lpf: текстийн шүүгч хэсэгт дэлгэрэнгүй тайлбарлагдсан байгаа.

Эхлээд /usr/local/libexec/if-simple гэсэн бүрхүүлийн скриптийг энгийн текст шүүгч болгоё. Энэ файлдаа өөрийн дуртай засварлагчаа ашиглан дараах текстийг оруулъя:

```
#!/bin/sh
#
# if-simple - Simple text input filter for lpd
# Installed in /usr/local/libexec/if-simple
#
# Simply copies stdin to stdout.  Ignores all filter arguments.

/bin/cat && exit 0
exit 2
```

Файлыг ажиллах боломжтой болгоно:

```
# chmod 555 /usr/local/libexec/if-simple
```

/etc/printcap файлд `if` боломжоор үүнийг зааж LPD-д ашиглах ёстойг нь хэлж өгнө. Бид үүнийг /etc/printcap жишээ файл дахь одоогоор бидэнд байгаа өөрсдийн хоёр хэвлэгчдээ нэмэх болно:

```
#
#  /etc/printcap for host rose - added text filter
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan:\ :lp=/dev/lpt0:\
        :if=/usr/local/libexec/if-simple:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:\
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:\
        :if=/usr/local/libexec/if-simple:
```

if-simple скриптийн хуулбарыг /usr/shared/examples/printing сангаас олж болно.

#### 10.3.1.5.7. LPD-г идэвхжүүлэх

lpd(8) нь `lpd_enable` хувьсагчаар хянагддаг /etc/rc-с ажилладаг. Энэ хувьсагчийн анхдагч утга нь `NO` байдаг. Тэгж хийгээгүй байгаа бол та:

```
lpd_enable="YES"
```

мөрийг /etc/rc.conf файлд нэмж өөрийн машиныг дахин ачаалах юм уу эсвэл ердөө л lpd(8)-г ажиллуулах хэрэгтэй.

```
# lpd
```

#### 10.3.1.5.8. Турших нь

Та энгийн LPD тохиргооны төгсгөлд хүрлээ. Харамсалтай нь баяр хүргэх арай л болоогүй байна. Учир нь бид тохиргоог тест хийж асуудал байвал засварлах хэрэгтэй хэвээр байгаа билээ. Тохиргоог тест хийхийн тулд ямар нэгэн юм хэвлэхийг оролдож үзээрэй. LPD системээр хэвлэхийн тулд ажлыг хэвлэхээр илгээдэг lpr(1) тушаалыг ашиглана.

Та lpr(1)-г Хэвлэгчийн холбоог шалгах нь хэсэгт танилцуулсан lptest(1) програмтай цуг хамтатган ямар нэг тест текст үүсгэхийн тулд ашиглаж болно.

*Энгийн LPD тохиргоог тест хийхийн тулд:*

Доор дурдсаныг бичнэ:

```
# lptest 20 5 | lpr -Pprinter-name
```

Энд байгаа *printer-name* нь /etc/printcap файлд заагдсан хэвлэгчийн нэр (эсвэл alias) юм. Анхдагч хэвлэгчийг тест хийхийн тулд lpr(1)-г ямар нэгэн `-P` нэмэлт өгөгдөлгүйгээр бичих хэрэгтэй. Дахин хэлэхэд PostScript®-г хүлээж байдаг хэвлэгчийг тест хийж байгаа бол lptest(1)-г ашиглахын оронд PostScript® програмыг тэр хэл дээр нь илгээх хэрэгтэй. Ингэхийн тулд та програмыг файлд хийж `lpr file` гэж бичиж болно.

PostScript® хэвлэгчийн хувьд та програмын үр дүнг авах ёстой. Хэрэв та lptest(1)-г ашиглаж байгаа бол таны авсан үр дүн иймэрхүү харагдах ёстой:

```
!"#$%&'()*+,-./01234
"#$%&'()*+,-./012345
#$%&'()*+,-./0123456
$%&'()*+,-./01234567
%&'()*+,-./012345678
```

Хэвлэгчийг цааш тест хийхийн тулд илүү том програмуудыг (хэлэн дээр суурилсан хэвлэгчдийн хувьд) татаж авах юм уу эсвэл lptest(1)-г өөр нэмэлт өгөгдлүүдтэй ажиллуулж үзээрэй. Жишээ нь `lptest 80 60` тушаал нь мөр бүртээ 80 тэмдэгттэй 60 мөрийг үүсгэх болно.

Хэрэв хэвлэгч ажиллаагүй бол Алдааг олж засварлах хэсгийг үзнэ үү.

# 10.4. Хэвлэгчийн илүү нарийн тохиргоо

Энэ хэсэгт онцгой загварчлагдсан файлыг шүүгчид, хуудасны толгой хэсэг, сүлжээгээр хэвлэх мөн хэвлэгчийг хэрэглэх эрх болон хязгаарлалт гэсэн зүйлүүдийн тухай зааварлах болно.

## 10.4.1. Шүүлтүүрүүд

Хэдийгээр LPD нь сүлжээний протокол, дараалал, хандалтын хяналт гэх мэтчилэн хэвлэлтийн асуудлуудыг хариуцдаг ч *жинхэнэ* ажлыг нь *шүүлтүүрүүд* хийдэг. Шүүлтүүрүүд нь тусгай програмууд бөгөөд хэвлэгчийн төхөөрөмжүүд болон онцгой шаардлагуудыг нь хариуцаж байдаг. Бид хамгийн энгийн суулгац хийх үед их амархан бөгөөд бараг бүх хэвлэгчид тохирдог текст шүүлтүүр суулгасан билээ. (Текст шүүлтүүр суулгах нь хэсэгт).

Хэвлэх төрлийн хувиргалт, хэвлэгчийн онцгой шинж чанар зэрэг давуу чанаруудыг нь хэрэглэхийн тулд танд шүүлтүүрийг зөв ойлгох шаардлага гарна. Ийм ажлуудыг зөвхөн шүүлтүүр хийдэг. Тэгээд муу мэдээ нь юу вэ гэхээр тэдгээр бүгдийг *та* тодорхойлж өгөх ёстой. Сайн мэдээ нь юу вэ гэхээр тэдгээрийн ихэнх нь боломжтой бэлэн байдаг бөгөөд хэрэв бичихээр бол тун амархан бичигддэг юм.

Мөн түүнчлэн FreeBSD /usr/libexec/lpr/lpf гэдэг нэртэй, бараг бүх хэвлэгчид ердийн текст хэвлэж чаддаг шүүлтүүртэй ирдэг. (энэ нь файл дотор байгаа зай болон том зай авалтуудыг таньдаг бөгөөд мөн эрх үүсгэх гээд түүний чаддаг бүх зүйл нь үүгээр дуусдаг) FreeBSD-н портын цуглуулга дотор хэд хэдэн шүүлтүүр болон шүүлтүүрийн бүрдэл хэсгүүд байдаг.

Энэ хэсэгт танд танилцуулах зүйлсүүдийг харуулъя:

- Шүүлтүүр хэрхэн ажилладаг вэ хэсэгт хэвлэх явцад шүүлтүүр ямар үүрэг гүйцэтгэдэг талаар тайлбарлахыг оролдох болно. LPD хэвлэх үедээ шүүлтүүрийг "хөшигний цаана" хэрхэн ашигладаг талаар та энд ойлгох болно. Үүнийг мэдсэнээр хэрэв өөр үед та олон шүүлтүүр суулгах үед хүндрэл гарвал даван туулахад тань хэрэг болно.
- LPD нь хэвлэгч бүрийг анхны тохируулгаараа ердийн текстийг хэвлэж чадна гэж авч үздэг. Энэ нь ердийн текстийг шууд хэвлэж чаддаггүй PostScript® хэвлэгчид (эсвэл бусад програмын хэл дээр үндэслэсэн хэвлэгчид ) дээр хүндрэл гаргадаг. PostScript® хэвлэгчид дээр ердийн текст хэвлэдэг болох  хэсэгт иймэрхүү хүндрэлийг даван туулах талаар хэлэх болно. Хэрэв танд PostScript® хэвлэгч байдаг бол энэ хэсгийг уншаарай.
- PostScript® нь маш олон програмын гаралт болж байдаг. Тэр бүү хэл зарим хүмүүс шууд PostScript® бичиж чаддаг. Харамсалтай нь PostScript® хэвлэгч нар үнэтэй. PostScript® биш төрлийн хэвлэгчийг PostScript® мэт хэрэглэх хэсэгт танд PostScript® өгөгдлийг *PostScript® биш төрлийн* хэвлэгч дээр хэвлэж болдог талаар зааварлах болно. Хэрэв танд PostScript® хэвлэгч байхгүй бол энэ хэсгийг та уншаарай.
- Хувиргагч шүүлтүүрүүд хэсэг танд онцгой төрлийн файлууд болох график эсвэл өөр төрлийн бичиглэлтэй файлуудыг хэрхэн таны хэвлэгчийн ойлгодог төрөл рүү хувиргаж хэвлэх талаар өгүүлэх болно. Энэ хэсгийг уншсаны дараа, та жишээ нь хэрэглэгчид `lpr -t` тушаал өгөн troff өгөгдөл хэвлэх эсвэл `lpr -d` тушаал өгөн TeX DVI өгөгдөл хэвлэх эсвэл `lpr -v` тушаал өгөн зураг төрлийн файл хэвлэх зэрэг зүйлүүдийг мэдэх болно. Энэ хэсгийг заавал унших хэрэгтэй гэж танд зөвлөж байна.

- Гаралтын шүүлтүүр  хэсэг нь LPD-н нэг их хэрэглэгддэг шинж чанарууд болох гаралтын шүүлтүүрүүдийн тухай өгүүлэх болно: Та хуудасны толгой хэсгийг хэвлэж болж л байвал (Хуудасны толгой хэсэг хэсэгт харна уу) энэ хэсгийг та алгасаж болно.
- lpf: текст шүүлтүүр  хэсэгт FreeBSD-тэй хамт нэлээн төгс ажиллагаатай мөрөөр нь хэвлэдэг (лазер хэвлэгч нар мөр мөрөөр нь хэвлэдэг туузан хэвлэгч шиг ажилладаг) `lpf` текст шүүлтүүрийн тухай өгүүлэх болно. Хэрэв та хэвлэгчийнхээ эрхийг түргэн бэлэн болгох эсвэл таны хэвлэгч хоосон зайг утаа байна гэж хүлээж аваад байвал та яах аргагүй `lpf`-тэй ажиллах хэрэг гарна.

Доорх хэсэгт харуулах янз бүрийн скриптийн хуулбарыг /usr/shared/examples/printing сангаас олж болно.

#### 10.4.1.1. Шүүлтүүр хэрхэн ажилладаг вэ

Өмнө дурдсанаар шүүлтүүр нь хэвлэгчтэй холбогдох төхөөрөмжөөс чөлөөтэй хэсгийг зохицуулах LPD-ээр эхлүүлэгдэн ажилладаг програм юм.

Ажил дахь файлыг LPD хэвлэхийг хүсэхдээ шүүлтүүр програмыг эхлүүлдэг. Энэ нь шүүлтүүрийн стандарт оролтыг хэвлэх файл уруу, өөрийн стандарт гаралтыг хэвлэгч уруу, өөрийн стандарт алдааг алдаа бүртгэх файл (/etc/printcap файл дахь `lf` боломжид эсвэл анхдагчаар /dev/console) уруу тохируулдаг.

LPD аль шүүлтүүрийг эхлүүлэх болон /etc/printcap файлд юу байгаа болон lpr(1) тушаалын мөр дэх ажилд зориулж хэрэглэгч ямар нэмэлт өгөгдлүүд зааснаас шүүлтүүрийн нэмэлт өгөгдлүүд хамаардаг. Жишээ нь хэрэв хэрэглэгч `lpr -t` тушаал бичсэн бол LPD нь хүрэх хэвлэгчийн `tf` боломжид жагсаагдсан troff шүүлтүүрийг эхлүүлэх болно. Хэрэв хэрэглэгч цэвэр текстийг хэвлэхийг хүссэн бол энэ нь `if` шүүлтүүрийг эхлүүлэх байсан (энэ нь бараг л үнэн юм: дэлгэрэнгүйг Гаралтын шүүлтүүрүүд хэсгээс үзнэ үү).

/etc/printcap файлд зааж өгч болох гурван төрлийн шүүлтүүр байдаг:

- *текст шүүлтүүр* буюу *оролтын шүүлтүүр* гэж LPD баримтад толгой эргүүлмээр нэрлэгдсэн шүүлтүүр нь ердийн текст хэвлэлтийг зохицуулдаг. Үүнийг анхдагч шүүлтүүр гэж бодох хэрэгтэй. LPD нь хэвлэгч бүрийг цэвэр текстийг анхдагчаар хэвлэж чаддаг гэж тооцдог бөгөөд устгах тэмдэгтүүд, tab-ууд эсвэл бусад тусгай тэмдэгтүүд хэвлэгчийг будлиулахгүй байлгахад санаа тавих нь текст шүүлтүүрийн ажил юм. Хэрэв та хэвлэгчийн хэрэглээний хувьд бүртгэлтэй тийм орчинд байгаа бол текст шүүлтүүр нь хэвлэгдсэн мөрүүдийн тоог тоолж тэр тоогоо хэвлэгчийн дэмждэг нэг хуудсан дахь мөрийн тоотой жишиж хэвлэсэн хуудаснуудыг бас бүртгэх ёстой. Текст шүүлтүүрийг дараах нэмэлт өгөгдлийн жагсаалттай эхлүүлдэг:

  `filter-name` [ -c ] -w *width* -l *length* -i *indent* -n *login* -h *host acct-file*

  энд байгаа

  **`-c`**

  нь ажил `lpr -l` тушаалаар илгээгдсэн бол гарч ирнэ

***width***

нь /etc/printcap файлд заагдсан `pw` (хуудасны өргөн) боломжийн утга юм. Анхдагчаар 132 байна

***length***

нь `pl` (хуудасны урт) боломжийн утга бөгөөд анхдагчаар 66 байна

***indent***

нь `lpr -i` тушаалаас өгөгдөх догол мөрний хэмжээ юм. Анхдагчаар 0 байна

***login***

нь файлыг хэвлэж байгаа хэрэглэгчийн бүртгэл юм

***host***

нь ажлыг илгээсэн хостын нэр юм

***acct-file***

нь `af` боломжийн бүртгэлийн файлын нэр юм.

- *Хувиргах шүүлтүүр* нь тусгай файлын хэлбэршилтийг хэвлэгчийн цаасан дээр амилуулж чадах хэлбэршилт уруу хувиргадаг. Жишээ нь ditroff маяг хийгдсэн өгөгдлийг шууд хэвлэж болдоггүй боловч ditroff өгөгдлийг хэвлэгчийн ойлгож хэвлэх хэлбэр уруу хувиргах ditroff файлуудад зориулсан хувиргах шүүлтүүрийг та суулгаж болох юм. Тэдгээрийн талаар бүгдийг Хувиргах шүүлтүүрүүд хэсэг хэлж өгдөг. Хэрэв танд хэвлэгчийн бүртгэл хийлт хэрэгтэй бол хувиргах шүүлтүүрүүд бас бүртгэл хийх хэрэгтэй болно. Хувиргах шүүлтүүрүүдийг дараах нэмэлт өгөгдлүүдтэй эхлүүлдэг:

  `filter-name` -x *pixel-width* -y *pixel-height* -n *login* -h *host acct-file*

  энд байгаа *pixel-width* нь `px` боломжийн утга (анхдагчаар 0) бөгөөд *pixel-height* нь `py` боломжийн утга юм (анхдагчаар 0).

- *Гаралтын шүүлтүүр* нь шүүх текст байхгүй байх юм уу эсвэл толгой хуудаснууд идэвхтэй байх тохиолдолд зөвхөн ашиглагддаг. Бидний туршлага дээрээс харахад гаралтын шүүлтүүрүүд ховор ашиглагддаг. Гаралтын шүүлтүүрүүд хэсэг тэдгээрийг тайлбарладаг. Гаралтын шүүлтүүрт зөвхөн хоёр нэмэлт өгөгдөл байдаг:

  `filter-name` -w *width* -l *length*

  эдгээр нь текст шүүлтүүрийн `-w` болон `-l` нэмэлт өгөгдлүүдтэй адил юм.

Шүүлтүүрүүд нь бас дараах гарах төлөвтэйгөөр *гарах* ёстой:

**exit 0**

Шүүлтүүр файлыг амжилттай хэвлэсэн тохиолдолд.

**exit 1**

Шүүлтүүр файлыг хэвлэж чадаагүй боловч LPD-гээр файлыг дахин хэвлүүлэхээр оролдохыг хүссэн. LPD нь шүүлтүүр энэ төлөвтэй гарсан бол түүнийг дахин эхлүүлэх

болно.

**exit 2**

Шүүлтүүр файлыг хэвлэж чадаагүй бөгөөд LPD-гээр файлыг дахин хэвлүүлэхийг хүсээгүй. LPD-нь файлыг гаргаж хаях болно.

FreeBSD хувилбартай цуг ирдэг текст шүүлтүүр /usr/libexec/lpr/lpf нь хуудасны өргөн болон уртын нэмэлт өгөгдлүүдийг ашиглан хэзээ form feed илгээх болон хэвлэгчийн хэрэглээг хэрхэн бүртгэхийг тодорхойлдог. Энэ нь бүртгэлийн оруулгуудыг бичихийн тулд нэвтрэлт, хост болон файлын нэмэлт өгөгдлүүдийн бүртгэл хийлтийг ашигладаг.

Хэрэв та шүүлтүүр худалдаж авахыг хүсэж байгаа бол тэдгээр нь LPD-тэй нийцтэй эсэхийг нь үзээрэй. Хэрэв тийм бол тэдгээр нь дээр тайлбарласан нэмэлт өгөгдлийн жагсаалтуудыг дэмжих ёстой. Та ердийн хэрэглээнд зориулж шүүлтүүрүүд бичих төлөвлөгөөтэй байгаа бол дээрх нэмэлт өгөгдлийн жагсаалт болон гарах кодуудыг дэмждэгээр тэдгээрийг хийгээрэй.

#### 10.4.1.2. PostScript® хэвлэгчид дээр цэвэр текст ажлуудыг тааруулах нь

Та өөрийн компьютер болон PostScript® (эсвэл бусад хэлэн дээр суурилсан) хэвлэгчийнхээ зөвхөн цорын ганц хэрэглэгч бөгөөд өөрийн хэвлэгч уруугаа цэвэр текстийг хэзээ ч илгээхгүй, таны хэвлэгч уруу цэвэр текстийг илгээхийг хүсдэг төрөл бүрийн програмуудын боломжуудыг хэзээ ч ашиглахгүй гэж амлах юм бол та энэ хэсгийн талаар санаа ерөөсөө зовох хэрэггүй юм.

Гэхдээ та PostScript® болон цэвэр текст ажлуудыг хэвлэгч уруу илгээхийг хүсэж байгаа бол өөрийн хэвлэгчийн тохиргоог нэн даруй нэмэх хэрэгтэй юм. Ингэхийн тулд ирж байгаа ажил цэвэр текст үү эсвэл PostScript® үү гэдгийг илрүүлэх текст шүүлтүүртэй байна. Бүх PostScript® ажлууд `%!`-аар (бусад хэвлэгчийн хэлнүүдийн хувьд өөрийн хэвлэгчийн баримтыг үзнэ үү) эхлэх ёстой. Хэрэв тэдгээр нь ажил дахь эхний хоёр тэмдэгт байгаа бол бид PostScript®-ийг ашиглаж ажлын үлдсэн хэсгийг шууд дамжуулж болно. Хэрэв тэдгээр нь файл дахь эхний хоёр тэмдэгт биш бол шүүлтүүр текстийг PostScript® уруу хувиргаж үр дүнг хэвлэх болно.

Үүнийг бид хэрхэн хийх вэ?

Хэрэв танд цуваа хэвлэгч байгаа бол үүнийг хийх агуу арга нь `lprps`-г суулгах явдал юм. `lprps` нь хэвлэгчтэй хоёр талын холбоог хийдэг PostScript® хэвлэгчийн шүүлтүүр юм. Энэ нь хэвлэгчийн төлвийн файлыг хэвлэгчийн дэлгэрэнгүй мэдээллээр шинэчилдэг. Ингэснээр хэрэглэгчид болон администраторууд хэвлэгчийн төлөвийг яг ямар байгааг (`toner low` эсвэл `paper jam` гэх зэрэг) харах боломжтой болох юм. Гэхдээ илүү чухал зүйл бол энэ нь ирж байгаа ажлыг цэвэр текст эсэхийг илрүүлж PostScript® уруу хөрвүүлэх `textps` (`lprps`-тэй цуг ирдэг өөр програм) тушаалыг дууддаг `psif` гэгддэг програмыг агуулдаг явдал юм. Дараа нь хэвлэгч уруу ажлыг илгээхийн тулд `lprps` тушаалыг энэ нь ашигладаг.

`lprps` нь FreeBSD-ийн портын цуглуулгад байдаг (Портын цуглуулга бүлгийг үзнэ үү). Ашиглах цаасны хэмжээний дагуу print/lprps-a4 болон print/lprps-letter портын аль нэгийг суулгаж та болно. `lprps`-г суулгасны дараа `lprps`-ийн хэсэг болох `psif` програмд замыг зааж өгөх хэрэгтэй. Хэрэв та `lprps`-г портын цуглуулгаас суулгасан бол доор дурдсаныг /etc/printcap файл дахь цуваа PostScript® хэвлэгчийн оруулгад ашиглах хэрэгтэй:

```
:if=/usr/local/libexec/psif:
```

LPD-д хэвлэгчийг унших-бичих горимоор онгойлгох боломж олгохын тулд `rw` боломж нь бас орсон байх шаардлагатай.

Хэрэв танд зэрэгцээ PostScript® хэвлэгч байгаа (бөгөөд тиймээс `lprps`-д хэрэгтэй хоёр талын холбоог хэвлэгчтэй цуг ашиглаж чадахгүй байгаа) бол та дараах бүрхүүлийн скриптийг текст шүүлтүүр маягаар ашиглаж болно:

```
#!/bin/sh
#
#  psif - Print PostScript or plain text on a PostScript printer
#  Script version; NOT the version that comes with lprps
#  Installed in /usr/local/libexec/psif
#

IFS="" read -r first_line
first_two_chars=`expr "$first_line" : '\(..\)'`

if [ "$first_two_chars" = "%!" ]; then
    #
    #  PostScript job, print it.
    #
    echo "$first_line" && cat && printf "\004" && exit 0
    exit 2
else
    #
    #  Plain text, convert it, then print it.
    #
    ( echo "$first_line"; cat ) | /usr/local/bin/textps && printf "\004" && exit 0
    exit 2
fi
```

Дээрх скрипт дээр байгаа `textps` нь цэвэр текстийг PostScript® уруу хувиргахаар бидний тусдаа суулгасан програм юм. Та дурын текстээс-PostScript® уруу хувиргадаг хүссэн програмаа ашиглаж болно. FreeBSD-ийн портын цуглуулга (Портын цуглуулга бүлгийг үзнэ үү) нь өргөн боломжтой текстээс-PostScript® уруу хувиргадаг `a2ps` гэгддэг програмыг агуулдаг бөгөөд та магадгүй үүнийг судлахыг хүсэж болох юм.

#### 10.4.1.3. PostScript® биш хэвлэгчид дээр PostScript® дүр үзүүлэх

PostScript® нь өндөр чанарын маяг тохируулалт болон хэвлэлтэд зориулагдсан *албан ёсны* стандарт юм. PostScript® нь гэхдээ *үнэтэй* стандарт юм. Aladdin Enterprises-ийн ачаар харин чөлөөтэй PostScript®-тэй төстэй, FreeBSD дээр ажилладаг Ghostscript гэгддэг програм байдаг. Ghostscript нь ихэнх PostScript® файлуудыг уншиж чаддаг бөгөөд тэдгээрийн хуудаснуудыг PostScript® биш хэвлэгчдийн олон загваруудыг оролцуулаад төрөл бүрийн төхөөрөмжүүд рүү амилуулж чаддаг. Ghostscript-г суулгаж өөрийн хэвлэгчийн хувьд тусгай текстийн шүүлтүүрийг ашиглан та өөрийн PostScript® биш хэвлэгчийг жинхэнэ PostScript® хэвлэгч

шигээр ажиллуулах боломжтой байдаг.

Ghostscript нь FreeBSD-ийн портын цуглуулгад байдаг. Олон хувилбар байдаг бөгөөд хамгийн ихээр ашиглагддаг хувилбар нь print/ghostscript-gpl юм.

PostScript® шиг дүр үзүүлэхийн тулд PostScript® файл хэвлэж байна уу гэдгийг илрүүлэх текст шүүлтүүр бидэнд байна. Хэрэв энэ нь тийм биш бол шүүлтүүр нь файлыг хэвлэгч уруу шууд дамжуулна; үгүй бол энэ нь файлыг хэвлэгчийн ойлгодог хэлбэр уруу эхлээд хувиргах Ghostscript-г ашиглах болно.

Энд жишээ байна: дараах скрипт нь Hewlett Packard DeskJet 500 хэвлэгчдэд зориулсан текст шүүлтүүр юм. Бусад хэвлэгчдийн хувьд `-sDEVICE` нэмэлт өгөгдлийг сольж `gs` (Ghostscript) тушаал уруу өгөх хэрэгтэй. (Ghostscript-ийн тухайн үеийн суулгацын дэмждэг төхөөрөмжүүдийн жагсаалтыг авахын тулд `gs -h` гэж бичих хэрэгтэй.)

```
#!/bin/sh
#
#  ifhp - Print Ghostscript-simulated PostScript on a DeskJet 500
#  Installed in /usr/local/libexec/ifhp

#
#  Treat LF as CR+LF (to avoid the "staircase effect" on HP/PCL
#  printers):
#
printf "\033&k2G" || exit 2

#
#  Read first two characters of the file
#
IFS="" read -r first_line
first_two_chars=`expr "$first_line" : '\(..\)'`

if [ "$first_two_chars" = "%!" ]; then
    #
    #  It is PostScript; use Ghostscript to scan-convert and print it.
    #
    /usr/local/bin/gs -dSAFER -dNOPAUSE -q -sDEVICE=djet500 \
      -sOutputFile=- - && exit 0
else
    #
    #  Plain text or HP/PCL, so just print it directly; print a form feed
    #  at the end to eject the last page.
    #
    echo "$first_line" && cat && printf "\033&l0H" &&
exit 0
fi

exit 2
```

Төгсгөлд нь та шүүлтүүрийн LPD-д `if` боломжийн тусламжтай мэдэгдэх хэрэгтэй:

```
:if=/usr/local/libexec/ifhp:
```

Ингээд л боллоо. Та `lpr plain.text` болон `lpr whatever.ps` гэж бичиж болох бөгөөд хоюулаа амжилттайгаар хэвлэгдэх болно.

#### 10.4.1.4. Хувиргах шүүлтүүрүүд

Хэвлэгчийн энгийн тохиргоо хэсэгт тайлбарласан энгийн тохиргоог дуусгасны дараа магадгүй таны хамгийн эхэнд хийхийг хүссэн зүйл чинь өөрийн дуртай файлын хэлбэршилтүүдэд (цэвэр ASCII текстээс гадна) зориулж хувиргах шүүлтүүрүүдийг суулгах явдал байж болох юм.

##### 10.4.1.4.1. Хувиргах шүүлтүүрүүдийг яагаад суулгах ёстой вэ?

Хувиргах шүүлтүүрүүд нь төрөл бүрийн файлуудыг хэвлэхэд амар болгодог. Жишээ нь бид TeX хэв маяг тохируулах системтэй их ажилладаг бөгөөд бидэнд PostScript® хэвлэгч байна гэж бодъё. TeX-ээс DVI файлыг үүсгэх болгонд бид DVI файлыг PostScript® уруу хувиргалгүйгээр шууд хэвлэж чадахгүй. Тушаалын дараалал иймэрхүү хийгдэнэ:

```
% dvips seaweed-analysis.dvi
% lpr seaweed-analysis.ps
```

DVI файлуудад зориулсан хувиргах шүүлтүүрийг суулгаж LPD-ээр өөрсөддөө зориулан хийлгэж бид гараар хувиргах алхмыг алгасаж болно. Одоо, DVI файлыг авах болгонд бидэнд хэвлэхэд нэг алхам дутуу байх болно:

```
% lpr -d seaweed-analysis.dvi
```

Бидэнд зориулж DVI файлын хувиргалтыг хийх LPD програмд `-d` тохируулгыг зааж өгнө. Хэлбэршүүлэх болон хувиргах тохируулгууд хэсэг хувиргах тохируулгуудыг жагсаасан байгаа.

Таны хүсэх хэвлэгчийн дэмжих хувиргах тохируулга бүрт зориулж *хувиргах шүүлтүүр* суулгаж түүний замыг /etc/printcap файлд зааж өгнө. Хувиргах шүүлтүүр нь энгийн хэвлэгчийн тохиргоон дахь (Текст шүүлтүүрийг суулгах нь хэсгийг үзнэ үү) текст шүүлтүүртэй төстэй. Ялгаа нь цэвэр текст хэвлэхийн оронд энэ шүүлтүүр нь файлыг хэвлэгчийн ойлгодог хэлбэршилт уруу хувиргадаг.

##### 10.4.1.4.2. Аль хувиргах шүүлтүүрүүдийг би суулгах ёстой вэ?

Та ашиглах хувиргах шүүлтүүрүүдээ суулгах ёстой. Хэрэв та маш их DVI өгөгдлийг хэвлэх бол DVI хувиргах шүүлтүүр хэрэглэгдэнэ. Хэрэв танд маш их troff хэвлэх хэрэгтэй бол та troff шүүлтүүрийг суулгахыг хүсэх байх.

Дараах хүснэгтэд LPD-ийн ажилладаг шүүлтүүрүүд, тэдгээрийн /etc/printcap файл дахь боломжийн оруулгууд болон тэдгээрийг `lpr` тушаалаар хэрхэн дуудах талаар дурдсан болой:

| Файлын төрөл | /etc/printcap боломж | lpr тохируулга |
|---|---|---|
| cifplot | `cf` | `-c` |
| DVI | `df` | `-d` |
| plot | `gf` | `-g` |
| ditroff | `nf` | `-n` |
| FORTRAN текст | `rf` | `-f` |
| troff | `tf` | `-f` |
| raster | `vf` | `-v` |
| цэвэр текст | `if` | none, `-p`, or `-l` |

Бидний жишээн дээрх `lpr -d` тушаалыг ашиглах нь хэвлэгчийн хувьд /etc/printcap файл дахь түүний оруулгад `df` боломж хэрэгтэй гэсэн үг юм.

Бусад хүмүүс магадгүй маргаж болох юм. Гэхдээ FORTRAN текст болон plot зэрэг хэлбэршилтүүд нь магадгүй хуучирсан хэлбэршилтүүд юм. Та өөрийн талдаа эдгээр шинэ ойлголтууд юм уу эсвэл ердөө л өөрчлөн тохируулсан шүүлтүүрүүдийг суулгаж аль нэг хэлбэршилтийн сонголтуудыг өгч болох юм. Жишээ нь та Printerleaf файлуудыг (Interleaf ширээний хэвлэх програмаас гарах файлууд) шууд хэвлэхийг хүсчээ. Гэхдээ plot файлуудыг хэзээ ч хэвлэхгүй гэж бодъё. Тэгвэл та `gf` боломжийн доор Printerleaf хувиргах шүүлтүүрийг суулгаж `lpr -g` тушаал нь "Printerleaf файлуудыг хэвлэх" гэсэн үг болохыг өөрийн хэрэглэгчиддээ сургах хэрэгтэй болох юм.

#### 10.4.1.4.3. Хувиргах шүүлтүүрүүдийг суулгах нь

Хувиргах шүүлтүүрүүд нь үндсэн FreeBSD суулгацаас гадна суулгах програмууд бөгөөд /usr/local сан дотор байрлана. /usr/local/libexec сан нь түгээмэл хэрэглэгддэг байрлал юм. Учир нь тэдгээр нь LPD-ийн зөвхөн ажиллуулах тусгай програмууд юм; ердийн хэрэглэгчид нь тэдгээрийг хэзээ ч ажиллуулах хэрэггүй юм.

Хувиргах шүүлтүүрийг идэвхжүүлэхийн тулд /etc/printcap файлд хүрэх хэвлэгчийн хувьд тохирох боломжийн доор түүний замыг зааж өгөх явдал юм.

Бидний жишээн дээр `bamboo` гэж нэрлэгдсэн хэвлэгчийн оруулгад DVI хувиргах шүүлтүүрийг бид нэмэх болно. `bamboo` хэвлэгчийн хувьд шинэ `df` боломжийг оруулсан /etc/printcap файлын жишээг энд дахин үзүүлэв:

```
#
#  /etc/printcap for host rose - added df filter for bamboo
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan:\
        :lp=/dev/lpt0:\
        :if=/usr/local/libexec/if-simple:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:\
```

```
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif:\
        :df=/usr/local/libexec/psdf:
```

DVI шүүлтүүр нь /usr/local/libexec/psdf гэж нэрлэгдсэн бүрхүүлийн скрипт юм. Энд тэр скриптийг үзүүлэв:

```
#!/bin/sh
#
#  psdf - DVI to PostScript printer filter
#  Installed in /usr/local/libexec/psdf
#
# Invoked by lpd when user runs lpr -d
#
exec /usr/local/bin/dvips -f | /usr/local/libexec/lprps "$@"
```

Энэ скрипт нь `dvips`-г шүүлтүүрийн горимд (`-f` нэмэлт өгөгдөл) хэвлэх ажил болох стандарт оролт дээр ажиллуулдаг. Дараа нь PostScript® хэвлэгчийн шүүлтүүр `lprps`-г (PostScript® хэвлэгчид дээр текст ажлуудыг тааруулах нь хэсгийг үзнэ үү) LPD-ийн энэ скрипт уруу дамжуулсан нэмэлт өгөгдлүүдтэй цуг эхлүүлнэ. `lprps` тушаал нь хэвлэсэн хуудаснуудыг бүртгэхийн тулд тэдгээр нэмэлт өгөгдлүүдийг ашиглах болно.

##### 10.4.1.4.4. Хувиргах шүүлтүүрийн зарим жишээнүүд

Хувиргах шүүлтүүрүүдийг суулгах тогтмол алхмууд байдаггүй болохоор энэ хэсэгт зарим нэг ажилладаг жишээнүүдийг энд дурдъя. Эдгээрийг өөрийн шүүлтүүрийг хийхдээ заавар болгон ашиглаарай. Тохирохоор бол шууд ашиглах хэрэгтэй.

Энэ жишээ скрипт нь Hewlett Packard LaserJet III-Si хэвлэгчид зориулсан raster (яг үнэндээ GIF файл) хувиргах шүүлтүүр юм:

```
#!/bin/sh
#
#  hpvf - Convert GIF files into HP/PCL, then print
#  Installed in /usr/local/libexec/hpvf

PATH=/usr/X11R6/bin:$PATH; export PATH
giftopnm | ppmtopgm | pgmtopbm | pbmtolj -resolution 300 \
    && exit 0 \
    || exit 2
```

Энэ нь GIF файлыг зөөгдөх anymap уруу, гарсныг нь зөөгдөх graymap уруу, гарсныг нь зөөгдөх bitmap уруу, тэр гарсныг нь LaserJet/PCL-тэй нийцтэй өгөгдөлд хувиргаж ажилладаг.

Дээрх шүүлтүүрийг ашиглаж байгаа хэвлэгчид зориулсан оруулгатай /etc/printcap файлыг энд үзүүлэв:

```
#
#  /etc/printcap for host orchid
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0:sh:sd=/var/spool/lpd/teak:mx#0:\
        :if=/usr/local/libexec/hpif:\
        :vf=/usr/local/libexec/hpvf:
```

Дараах скрипт нь bamboo гэж нэрлэгдсэн PostScript® хувилагчийн groff маяг тохируулах системээс troff өгөгдөлд зориулсан хувиргах шүүлтүүр юм:

```
#!/bin/sh
#
#  pstf - Convert groff's troff data into PS, then print.
#  Installed in /usr/local/libexec/pstf
#
exec grops | /usr/local/libexec/lprps "$@"
```

Дээрх скрипт нь хэвлэгчтэй хийх холбоог зохицуулахын тулд lprps-г дахин ашигладаг. Хэрэв хэвлэгч зэрэгцээ порт дээр байгаа бол бид энэ скриптийг харин ашиглах болно:

```
#!/bin/sh
#
#  pstf - Convert groff's troff data into PS, then print.
#  Installed in /usr/local/libexec/pstf
#
exec grops
```

Ингээд л боллоо. Шүүлтүүрийг идэвхжүүлэхийн тулд бид дараах оруулгыг /etc/printcap файлд нэмэх хэрэгтэй:

```
:tf=/usr/local/libexec/pstf:
```

Хуучин гаруудыг FORTRAN-с ичихэд хүргэх жишээг энд үзүүлэв. Энэ нь цэвэр текстийг шууд хэвлэж чадах дурын хэвлэгчид зориулсан FORTRAN-текст шүүлтүүр юм. Бид үүнийг teak хэвлэгчид зориулж суулгах болно:

```
#!/bin/sh
#
# hprf - FORTRAN text filter for LaserJet 3si:
# Installed in /usr/local/libexec/hprf
#

printf "\033&k2G" && fpr && printf "\033&l0H" &&
 exit 0
```

```
exit 2
```

Тэгээд бид энэ шүүлтүүрийг идэвхжүүлэхийн тулд `teak` хэвлэгчид зориулж /etc/printcap файлд энэ мөрийг нэмэх болно:

```
:rf=/usr/local/libexec/hprf:
```

Энд сүүлийн бөгөөд зарим талаараа төвөгтэй жишээ байна. Бид DVI шүүлтүүрийг өмнө танилцуулсан LaserJet хэвлэгч `teak`-д нэмэх болно. Эхлээд амархан хэсэг нь: /etc/printcap файлыг DVI шүүлтүүрийн байрлалаар шинэчлэх явдал юм:

```
:df=/usr/local/libexec/hpdf:
```

Одоо хэцүү хэсэг нь: шүүлтүүр хийх явдал юм. Үүний тулд бидэнд DVI-ээс-LaserJet/PCL уруу хувиргах програм хэрэгтэй. FreeBSD-ийн портын цуглуулгад (see Портын цуглуулга) нэг байдаг: print/dvi2xx. Энэ портыг суулгах нь бидэнд хэрэгтэй програм `dvilj2p`-г өгдөг. Энэ нь DVI-г LaserJet IIp, LaserJet III, болон LaserJet 2000-тай нийцтэй кодууд уруу хувиргадаг.

`dvilj2p` нь шүүлтүүр `hpdf`-г нэлээн төвөгтэй болгодог. Учир нь `dvilj2p` нь стандарт оролтоос уншиж чаддаггүй. Энэ нь файлын нэртэй ажиллахыг хүсдэг. Юу нь бүр муу вэ гэхээр файлын нэр нь .dvi гэж төгсөх ёстой. Тэгэхээр /dev/fd/0-г стандарт оролтод зориулж ашиглах нь асуудалтай болно. Бид түр зуурын файлын нэрийг (.dvi-аар төгссөн) /dev/fd/0 уруу болгож холбоос үүсгэн энэ асуудлыг тойрон гарч болох юм. Ингэснээр `dvilj2p` тушаалыг стандарт оролтоос уншуулах юм.

Асуудалтай нэг зүйл нь бид /tmp-г түр зуурын холбоост ашиглаж болохгүй явдал юм. Симболын холбоосуудыг `bin` хэрэглэгч болон бүлэг эзэмшдэг. Тэгээд /tmp санд наалттай (sticky) бит тохируулагдсан байдаг. Шүүлтүүр нь холбоос үүсгэж чадах боловч холбоос нь өөр хэрэглэгчид харьяалагдах болохоор дууссаныхаа дараа цэвэрлэж устгаж чаддаггүй юм.

Харин шүүлтүүр нь түр хадгалах сан (/etc/printcap файлд `sd` боломжоор заагдсан) болох одоо ажиллаж байгаа санд симболын холбоос үүсгэх болно . Энэ нь шүүлтүүрийн ажлаа хийх төгс газар юм. Ялангуяа илүү их чөлөөтэй дискний зай /tmp сан доторхоос илүү түр хадгалах санд (заримдаа) байдаг.

Ингээд эцэст нь шүүлтүүр энэ байна:

```
#!/bin/sh
#
#  hpdf - Print DVI data on HP/PCL printer
#  Installed in /usr/local/libexec/hpdf

PATH=/usr/local/bin:$PATH; export PATH

#
#  Define a function to clean up our temporary files.  These exist
#  in the current directory, which will be the spooling directory
```

```
#  for the printer.
#
cleanup() {
   rm -f hpdf$$.dvi
}

#
#  Define a function to handle fatal errors: print the given message
#  and exit 2.  Exiting with 2 tells LPD to do not try to reprint the
#  job.
#
fatal() {
    echo "$@" 1>&2
    cleanup
    exit 2
}

#
#  If user removes the job, LPD will send SIGINT, so trap SIGINT
#  (and a few other signals) to clean up after ourselves.
#
trap cleanup 1 2 15

#
#  Make sure we are not colliding with any existing files.
#
cleanup

#
#  Link the DVI input file to standard input (the file to print).
#
ln -s /dev/fd/0 hpdf$$.dvi || fatal "Cannot symlink /dev/fd/0"

#
#  Make LF = CR+LF
#
printf "\033&k2G" || fatal "Cannot initialize printer"

#
#  Convert and print.  Return value from dvilj2p does not seem to be
#  reliable, so we ignore it.
#
dvilj2p -M1 -q -e- dfhp$$.dvi

#
#  Clean up and exit
#
cleanup
exit 0
```

#### 10.4.1.4.5. Автомат хувиргалт: Хувиргагч шүүлтүүртэй ижилхэн шүүлтүүрүүд

Эдгээр бүх хувиргагч шүүлтүүрүүд нь таны хэвлэх орчинд тань маш олон ажлыг гүйцэтгэж өгдөг ч хэрэглэгчдийг тэднээс аль нэгийг нь хэрэглэхийг шаарддаг (lpr(1)-н тушаалын мөрөн дээр). Хэрэв таны хэрэглэгчид компьютер дээр сайн гаршаагүй хүмүүс бол иймэрхүү шүүлтүүрийн сонголтууд тэд нарыг залхаах болно. Бүр муудаж болох нөхцөл юу гэвэл буруу сонгосон шүүлтүүрийг буруу файлд хэрэглэснээр хэвлэгчээр олон зуун хуудас үр дүнгүй хэвлэгдэх явдал юм.

Анхнаасаа хувиргагч шүүлтүүрүүдийг суулгахаасаа өмнө эхлээд та энгийн текст шүүлтүүрийг суулгаад (анхдагч шүүлтүүр ) хэвлэгдэх файлын төрлийг нь таниад хэрэгтэй шүүлтүүрийг хэрэглэх нь зүйтэй. `file` гэх мэтийн хэрэгслүүд иймэрхүү зүйлд ашиглагдаж болох юм. Мэдээж *зарим* файлын төрлийг хооронд нь ялгахад бэрх боловч мэдээж тэдэнд зориулж та шүүлтүүр үүсгэж болно.

FreeBSD-н портын цуглуулга дотор `apsfilter` (print/apsfilter) нэртэй автомат хувиргалт хийдэг текст шүүлтүүр бий. Энэ шүүлтүүр нь энгийн текст, PostScript®, DVI болон бараг бүх л төрлийн файлуудыг таньж тэдэнд тохирсон шүүлтүүрийг нь автоматаар ажиллуулдаг.

### 10.4.1.5. Гаралтын шүүлтүүрүүд

LPD түр хадгалагч системд бид нарын нээгээгүй бас нэг шүүлтүүрийн төрөл бий. Гаралтын шүүлтүүр гэгч нь текст шүүлтүүр шиг зөвхөн энгийн текст хэвлэх зориулалттай бөгөөд маш олон зүйлийг хялбаршуулсан байдаг. Хэрэв та текст шүүлтүүр хэрэглээгүй мөртлөө гаралтын шүүлтүүр хэрэглэж байгаа бол:

- LPD нь хэвлэгдэх файл болгонд шүүлтүүр ажиллуулалгүйгээр гаралтын шүүлтүүрийг нэг удаа л эхлүүлдэг.
- LPD нь гаралтын шүүлтүүрт зориулж файлын эхлэл болон төгсгөлийг мэдээлэх алба байхгүй.
- LPD нь хэрэглэгчийн болон компьютерийн нэвтрэх эрхийг дамжуулдаггүй болохоор бүртгэл хийх зориулалт байдаггүй. Үнэн хэрэгтээ бол энэ нь ердөө хоёр утга авдаг:

  `шүүлтүүрийн-нэр` -w*өргөн* -l*урт*

  Энд *өргөн* нь `pw`-н шинж чанар бөгөөд *урт* нь `pl`-н шинж чанар юм.

Гаралтын шүүлтүүрийн амарчлалд бас найдах хэрэггүй. Хэрэв хэвлэгдэх файлуудыг тусдаа хуудаснаас эхлүүлэхийг та хүсэж байгаа бол гаралтын шүүлтүүр *ажиллахгүй*. Үүнд текст шүүлтүүр (бас оролтын шүүлтүүр гэж нэрлэдэг) хэрэглэх хэрэгтэй ба Текст шүүлтүүрийг суулгах нь  хэсгээс харна уу. Мөн цаашлаад үнэн чанартаа бол гаралтын шүүлтүүрүүд нь байт хэлбэрээр урсаж байгаа хэвлэгдэх зүйлсээс онцгой тэмдэгтүүдийг ялгаж LPD-н нэрийн өмнөөс дохио илгээх маягаар ажилладаг *нэлээн төвөгтэй* систем юм.

Хэрэв та толгой хуудаснуудыг хэвлэхийг хүсэж escape дарааллууд юм уу эсвэл эхлүүлэх мөрүүдийг илгээх хэрэгтэй бол гарах шүүлтүүр нь *шаардлагатай* юм. (Гэхдээ хэрэв та толгой хуудаснуудыг хүсэж байгаа хэрэглэгчийн бүртгэлээс төлбөр авах хүсэлтэй байгаа бол энэ нь *дэмий* зүйл юм. Учир нь LPD нь гаралтын шүүлтүүрт ямар ч хэрэглэгч юм уу эсвэл хостын мэдээллийг өгдөггүй юм.)

Нэг хэвлэгч дээр LPD нь гаралтын шүүлтүүр болон текст эсвэл бусад шүүлтүүрүүдийг зөвшөөрдөг. Ийм тохиолдолд LPD нь зөвхөн толгой хуудсыг хэвлэхийн тулд гаралтын шүүлтүүрийг эхлүүлэх болно (Толгой хуудаснууд хэсгийг үзнэ үү). Тэгээд LPD нь хоёр байтыг шүүлтүүр рүү илгээж гаралтын шүүлтүүр нь *өөрийгөө зогсооно* гэж тооцдог: эдгээр хоёр байт нь ASCII 031 дараа нь ASCII 001 юм. Гаралтын шүүлтүүр нь эдгээр хоёр байтыг (031, 001) хараад өөртөө `SIGSTOP` илгээж зогсох ёстой. LPD бусад шүүлтүүрийг ажиллуулж дууссаны дараа гаралтын шүүлтүүрт `SIGCONT` илгээж түүнийг дахин эхлүүлэх болно.

Хэрэв гаралтын шүүлтүүр байгаа боловч текст шүүлтүүр *байхгүй* бөгөөд LPD цэвэр текст ажил дээр ажиллаж байгаа бол LPD нь ажлыг хийхийн тулд гаралтын шүүлтүүрийг ашигладаг. Өмнө хэлсэнчлэн гаралтын шүүлтүүр нь ажлын файл бүрийг дарааллаар form feed-үүд эсвэл бусад цаасны дэвшилт зэрэг хөндлөнгийн оролцоогүйгээр хэвлэх бөгөөд энэ нь таны хүсэж байгаа зүйл *биш* байж болох юм. Бараг бүх тохиолдолд танд текст шүүлтүүр хэрэгтэй.

`lpf` гэж танд урьд нь текст шүүлтүүр хэмээн танилцуулсан шүүлтүүр бас гаралтын шүүлтүүр болон ажилладаг. Хэрэв танд ямар нэгэн урсгалаас байт таньж код илгээх гаралтын шүүлтүүр бичилгүйгээр түргэн-боловч-бултар шүүлтүүр хэрэгтэй бол `lpf`-г хэрэглээрэй. Та мөн бүрхүүл дээр скрипт бичиж хэвлэгчид шаардлагатай анхдагч утгуудыг нь `lpf`-аар тохируулж бас болно.

#### 10.4.1.6. `lpf`: Текст шүүлтүүр

FreeBSD хоёртын түгээлттэй цуг ирдэг /usr/libexec/lpr/lpf програм нь гаралтад (`lpr -i` тушаалаар илгээгдсэн ажил) догол гаргах, үсгэн тэмдэгтүүдийг дамжуулахыг зөвшөөрөх (`lpr -l` тушаалаар илгээгдсэн ажил), ажил дахь устгах тэмдэгтүүд болон tab-уудад зориулж хэвлэх байрлалыг тааруулах болон хэвлэсэн хуудаснуудыг бүртгэж чаддаг текст шүүлтүүр (оролтын шүүлтүүр) юм. Энэ нь бас гаралтын шүүлтүүр шиг ажиллаж чаддаг.

`lpf` нь олон хэвлэх орчнуудад тохирдог. Хэдийгээр энэ нь хэвлэгч уруу эхлүүлэх дарааллуудыг илгээх боломжгүй боловч хэрэгтэй эхлүүлэлтийг хийж `lpf`-г ажиллуулах бүрхүүлийн скрипт бичихийг хялбар болгодог.

Хуудсын бүртгэлийг `lpf`-ээр зөв хийлгэхийн тулд /etc/printcap файл дахь `pw` болон `pl` боломжуудад зөв утгуудыг оруулах хэрэгтэй. Хуудсанд хичнээн текст багтах болон хэрэглэгчийн ажилд хичнээн хуудас байгааг тодорхойлохын тулд энэ нь эдгээр утгуудыг ашигладаг. Хэвлэгчийн бүртгэлийн тухай дэлгэрэнгүй мэдээллийг Хэвлэгчийн хэрэглээг бүртгэх нь хэсгээс үзнэ үү.

### 10.4.2. Толгой хуудаснууд

Хэрэв та *маш их* хэрэглэгчтэй бөгөөд бүгд төрөл бүрийн хэвлэгч ашиглаж байгаа бол та магадгүй *толгой хуудаснууд*ыг ашиглахыг хүсэж болох юм.

Толгой хуудаснууд, бас *сурталчилгаа* эсвэл *тэсрэх хуудаснууд* гэгддэг хуудаснууд нь хуудсууд хэвлэгдсэний дараа ажлууд нь хэнд хамаарахыг тодорхойлдог. Тэдгээр нь ихэвчлэн том, тод үсгээр, магадгүй чимсэн хүрээнүүдтэйгээр хэвлэгддэг. Ингэснээр хэвлэсэн олон хуудаснууд дотроос хэрэглэгчдийн ажлуудыг багтаасан жинхэнэ баримт болон ялгардаг. Энэ нь хэрэглэгчдэд өөрсдийн ажлуудыг хурдан олох боломжийг олгодог. Толгой хуудасны

илэрхий сул тал нь ажил болгоны хувьд илүү хуудас хэвлэгддэг явдал бөгөөд тэдний түр зуурын ашиг хэдхэн минут үргэлжилж тун удалгүй тэдгээрийг хогийн сав эсвэл хогийн овоон дотор орохыг харж болох юм. (Толгой хуудаснууд нь файл болгонтой биш, ажил болгонтой цуг явдгийг санаарай. Тэгэхээр цаас дэмий үрэх нь тийм ч их муу биш байж болох юм.)

Хэрэв таны хэвлэгч цэвэр текстийг шууд хэвлэж чаддаг бол LPD систем нь таны хэвлэсэн зүйлүүдэд толгой хуудаснуудыг автоматаар өгч чаддаг. Хэрэв танд PostScript® хэвлэгч байгаа бол толгой хуудсыг үүсгэхийн тулд танд гадаад програм хэрэгтэй болох юм; PostScript® хэвлэгчид дээрх толгой хуудаснууд хэсгийг үзнэ үү.

#### 10.4.2.1. Толгой хуудаснуудыг идэвхжүүлэх нь

Хэвлэгчийн энгийн тохиргоо хэсэгт бид /etc/printcap файлд sh-ийг ("suppress header" буюу толгойг дарах гэсэн утгатай) зааж өгөн толгой хуудаснуудыг болиулсан. Хэвлэгчийн хувьд толгой хуудаснуудыг идэвхжүүлэхийн тулд ердөө л sh боломжийг устгах хэрэгтэй.

Хэтэрхий амархан санагдаж байна, тийм үү?

Таны зөв. Та хэвлэгч уруу эхлүүлэх мөрүүдийг илгээхийн тулд гаралтын шүүлтүүрийг өгөх хэрэгтэй болж *болох* юм. Энд Hewlett Packard PCL-тэй нийцтэй хэвлэгчдэд зориулсан жишээ гаралтын шүүлтүүр байна:

```
#!/bin/sh
#
#  hpof - Output filter for Hewlett Packard PCL-compatible printers
#  Installed in /usr/local/libexec/hpof

printf "\033&k2G" || exit 2
exec /usr/libexec/lpr/lpf
```

Гаралтын шүүлтүүрийн замыг of боломжид зааж өгөх хэрэгтэй. Дэлгэрэнгүй мэдээллийг Гаралтын шүүлтүүрүүд хэсгээс үзнэ үү.

Бидний урьд нь танилцуулсан teak хэвлэгчид зориулсан жишээ /etc/printcap файл энд байна; бид толгой хуудаснуудыг идэвхжүүлж дээрх гаралтын шүүлтүүрийг нэмсэн:

```
#
#  /etc/printcap for host orchid
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0:sd=/var/spool/lpd/teak:mx#0:\
        :if=/usr/local/libexec/hpif:\
        :vf=/usr/local/libexec/hpvf:\
        :of=/usr/local/libexec/hpof:
```

Одоо хэрэглэгч teak уруу ажлуудаа хэвлэхэд ажил болгонтой толгой хуудас хүлээн авах болно. Хэрэв хэрэглэгчид өөрсдийн хэвлэсэн зүйлсээ хайхад цаг зарцуулахыг хүсэж байгаа

бол толгой хуудаснуудыг дарж ажлаа `lpr -h` тушаалаар илгээж болно; lpr(1)-ийн тохируулгуудын талаар дэлгэрэнгүйг Толгой хуудасны тохируулгууд хэсгээс үзнэ үү.

> LPD нь толгой хуудасны дараа form feed тэмдэгт хэвлэдэг. Хэрэв таны хэвлэгч хуудсаа гаргахын тулд өөр тэмдэгт юм уу эсвэл өөр тэмдэгтүүдийн дараалал ашигладаг бол тэдгээрийг /etc/printcap файлд `ff` боломжоор зааж өгөх хэрэгтэй.

#### 10.4.2.2. Толгой хуудаснуудыг хянах нь

Толгой хуудаснуудыг идэвхжүүлснээр LPD нь хэрэглэгч, хост, болон ажлыг таниулах том үсгүүдтэй *урт толгой* бүхий хуудсыг үүсгэх болно. Энд жишээ байна (`kelly` нь "outline" гэсэн нэртэй ажлыг `rose`-с хэвлэжээ):

```
k                  ll       ll
k                   l        l
k                   l        l
k   k     eeee      l        l     y    y
k  k     e    e     l        l     y    y
k k      eeeeee     l        l     y    y
kk k     e          l        l     y    y
k   k    e    e     l        l     y   yy
k    k    eeee     lll      lll     yyy y
                                        y
                                    y    y
                                     yyyy

                             ll
                    t         l        i
                    t         l
 oooo    u    u   ttttt       l       ii     n nnn     eeee
o    o   u    u     t         l        i     nn   n   e    e
o    o   u    u     t         l        i     n    n   eeeeee
o    o   u    u     t         l        i     n    n   e
o    o   u   uu     t  t      l        i     n    n   e    e
 oooo     uuu u      tt      lll      iii    n    n    eeee

r rrr     oooo     ssss     eeee
rr   r   o    o   s    s   e    e
r        o    o    ss      eeeeee
r        o    o      ss    e
r        o    o   s    s   e    e
r         oooo     ssss     eeee

                                        Job:  outline
                                        Date: Sun Sep 17 11:04:58 1995
```

LPD нь ажлыг шинэ хуудсан дээр эхлүүлэхийн тулд энэ текстийн дараа form feed-г нэмдэг (/etc/printcap дахь хүрэх хэвлэгчийн оруулгад `sf` (form feed-г дарах) байхгүй бол).

Хэрэв та хүсэж байгаа бол LPD нь *богино толгой* хийж чаддаг; /etc/printcap файлд `sb`-г (short banner буюу богино сурталчилгаа) зааж өгөх хэрэгтэй. Толгой хуудас иймэрхүү харагдах болно:

```
rose:kelly  Job: outline  Date: Sun Sep 17 11:07:51 1995
```

Бас анхдагчаар LPD нь эхлээд толгой хуудсыг дараа нь ажлыг хэвлэдэг. Эсрэгээр болгохын тулд /etc/printcap файлд `hl`-г (header last буюу толгой сүүлд) зааж өгөх хэрэгтэй.

#### 10.4.2.3. Толгой хуудаснуудыг бүртгэх нь

LPD-н өөрт нь суугдсан толгой хуудаснууд нь хэвлэлтэд бүртгэл тооцоо хийх шаардлагатай үед *үнэгүй* байх зарчмыг шаарддаг.

Яагаад?

Учир нь гаралтын шүүлтүүр бол толгой хуудас хэвлэгдэх үед ажилладаг тусдаа биеэ даасан гадаад програм бөгөөд бүртгэл хийж болох юм шиг боловч *хэрэглэгч юм уу компьютерийн* тухай ямар ч бүртгэлийн мэдээллээр хангагддаггүй учир хэн дээр ямар тооцоо бүртгэл хийхээ мэдэж чаддаггүй. Зөвхөн текст шүүлтүүрийг өөрчилж "хуудасны тоог нэгээр нэмэх" юм уу эсвэл хувиргах ямар нэгэн шүүлтүүрийг (хэрэглэгч юм уу компьютерийн мэдээлэлтэй) хэрэглэгч `lpr -h` гэсэн тушаалаар дарж болдог. Дарагдсан ч гэсэн тэдгээр толгой хуудаснууд бүртгэл тооцоо хийж байдаг. Үндсэндээ орчны хувьсагчдад дассан хэрэглэгчдэд `lpr -h` тушаал илүү аятайхан байж болох боловч шаардлагатай гэсэн үг биш билээ.

Шүүлтүүр бүр өөрсдийн гэсэн толгой хуудас үүсгэх нь *тийм хангалттай шийдэл бас биш* (хэдийгээр тус бүрдээ төлбөр тооцоо хийж чадах байсан ч ). Хэрэв хэрэглэгч `lpr -h` тушаалаар толгой хуудсыг дарсан ч LPD нь шүүлтүүрүүд рүү `-h` гэсэн сонголтын талаар ямар ч мэдээлэл явуулдаггүй болохоор төлбөр тооцоо хийгдсээр байх болно.

Тэгэхээр танд ямар сонголтууд байна вэ?

Та дараах зүйлсийг хийх боломжтой:

- LPD-н зарчмыг хүлээн зөвшөөрч толгой хуудсанд төлбөр тооцохгүй болох.
- LPRng гэх мэтийн LPD-н ижил төрлийн түр хадгалагчдыг хэрэглэх. Түр хадгалагчидтай ижил төрлийн хадгалагчид хэсэгт LPD-г орлуулж болох түр хадгалагчдын талаар өгүүлсэн буй.
- *Гайгүй сэргэлэн* гаралтын шүүлтүүр бичих. Гаралтын шүүлтүүр гэдэг нь ердийн үед бол хэвлэгчийг тохируулах юм уу эсвэл энгийн тэмдэгт хувиргуур хийдэг програм юм. Ийм шүүлтүүр нь толгой хуудас болон энгийн текст зэрэгт тун таардаг ( хэрэв тэнд ямар ч текст (оролтын) шүүлтүүр байхгүй бол). Гэхдээ хэрэв тэнд энгийн текст хэвлэх текст шүүлтүүр байгаа бол LPD нь гаралтын шүүлтүүрийг зөвхөн толгой хуудсанд ажиллуулдаг. Тэгээд гаралтын шүүлтүүр нь толгой хуудаснаас LPD-н үүсгэсэн хэрэглэгч болон компьютерт тооцсон тооцоог салгаж авч чаддаг. Энэ арганд буй нэг хүндрэл бол гаралтын шүүлтүүр нь ямар бүртгэлийн файл хэрэглэхээ мэдэхгүйд байгаа юм (энэ файл нь `af` тохиргоогоор дамжигддаггүй). Гэвч хэрэв танд бэлэн бүртгэл файл байгаа бол тэдгээрийг гаралтын шүүлтүүр дотор бичиж өгч болно. Тооцоог салгаж авах хэсэгт нэмэр

болгохын тулд `sh` (short header буюу богино толгой) тохиргоог /etc/printcap файл дотор оруулах нь зүйтэй. Тэгээд ч энэ бүхэн танд төвөгтэй санагдаж магадгүй бөгөөд хэрэглэгчид толгой хуудсанд төлбөр тооцдоггүй өгөөмөр сүлжээ зохицуулагчийг илүүд үзэж бас болох юм.

#### 10.4.2.4. PostScript® хэвлэгч дээрх толгой хуудаснууд

Өмнө өгүүлснээр LPD нь маш олон хэвлэгчдэд зориулж энгийн текст толгой хуудас үүсгэж чаддаг. Мэдээж PostScript® нь энгийн текстийг шууд хэвлэж чаддаггүй болохоор LPD-н энэ шинж чанар бараг хэрэггүй билээ.

Толгой хуудас байлгах нэг арга бол хувиргах шүүлтүүр болгон толгой хуудас үүсгэх юм. Ингэхийн тулд шүүлтүүрүүд нь хэрэглэгч болон компьютерийн мэдээллийг хүлээж аваад толгой хуудас үүсгэдэг байхаар зохицуулах хэрэгтэй. Энэ аргын нэг муу тал нь хэрэглэгчид `lpr -h` тушаал ашигласан ч толгой хуудастай хуудас хэвлээд байх болно.

Энэ аргыг жаахан судалж үзэцгээе. Дараах скрипт гурван утга хүлээж аваад(хэрэглэгчийн нэвтрэх нэр, компьютерийн нэр, хэвлэгдэх ажлын нэр) PostScript® толгой хуудас үүсгэж байна:

```
#!/bin/sh
#
#  make-ps-header - make a PostScript header page on stdout
#  Installed in /usr/local/libexec/make-ps-header
#

#
#  These are PostScript units (72 to the inch).  Modify for A4 or
#  whatever size paper you are using:
#
page_width=612
page_height=792
border=72

#
#  Check arguments
#
if [ $# -ne 3 ]; then
    echo "Usage: `basename $0` <user> <host> <job>" 1>&2
    exit 1
fi

#
#  Save these, mostly for readability in the PostScript, below.
#
user=$1
host=$2
job=$3
date=`date`
```

```
#
#  Send the PostScript code to stdout.
#
exec cat <<EOF
%!PS

%
%  Make sure we do not interfere with user's job that will follow
%
save

%
%  Make a thick, unpleasant border around the edge of the paper.
%
$border $border moveto
$page_width $border 2 mul sub 0 rlineto
0 $page_height $border 2 mul sub rlineto
currentscreen 3 -1 roll pop 100 3 1 roll setscreen
$border 2 mul $page_width sub 0 rlineto closepath
0.8 setgray 10 setlinewidth stroke 0 setgray

%
%  Display user's login name, nice and large and prominent
%
/Helvetica-Bold findfont 64 scalefont setfont
$page_width ($user) stringwidth pop sub 2 div $page_height 200 sub moveto
($user) show

%
%  Now show the boring particulars
%
/Helvetica findfont 14 scalefont setfont
/y 200 def
[ (Job:) (Host:) (Date:) ] {
200 y moveto show /y y 18 sub def }
forall

/Helvetica-Bold findfont 14 scalefont setfont
/y 200 def
[ ($job) ($host) ($date) ] {
        270 y moveto show /y y 18 sub def
} forall

%
% That is it
%
restore
showpage
EOF
```

Одоо ямар ч хувиргах шүүлтүүр болон текст шүүлтүүрүүд нь энэ скриптийг дуудаж толгой хуудас үүсгэж хэвлэж болно. Энд одоо өмнө үзүүлж байсан DVI хувиргах шүүлтүүрийг жаахан өөрчилж толгой хуудас үүсгэхийг харуулав:

```
#!/bin/sh
#
#  psdf - DVI to PostScript printer filter
#  Installed in /usr/local/libexec/psdf
#
#  Invoked by lpd when user runs lpr -d
#

orig_args="$@"

fail() {
    echo "$@" 1>&2
    exit 2
}

while getopts "x:y:n:h:" option; do
    case $option in
        x|y)  ;; # Ignore
        n)    login=$OPTARG ;;
        h)    host=$OPTARG ;;
        *)    echo "LPD started `basename $0` wrong." 1>&2
              exit 2
              ;;
    esac
done

[ "$login" ] || fail "No login name"
[ "$host" ] || fail "No host name"

( /usr/local/libexec/make-ps-header $login $host "DVI File"
  /usr/local/bin/dvips -f ) | eval /usr/local/libexec/lprps $orig_args
```

Хэрэглэгчийн нэр болон компьютерийн нэрийг хэрхэн салгаж авч байгааг анхаарах хэрэгтэй. Бусад шүүлтүүр дээр ийм салгалт хийх нь ижилхэн. Текст шүүлтүүр нь хэрэглэгчийн нэр болон компьютерийн нэрийг арай өөр аргаар салгаж авдаг. (Шүүлтүүр хэрхэн ажилладаг вэ хэсгээс харна уу).

Өмнө өгүүлсний дагуу дээр үзүүлсэн энэ загвар нь хэдийгээр тун энгийн бүтээгдсэн ч гэсэн "suppress header page" сонголтыг ( `-h` сонголт) `lpr` дээр хорьдог. Хэрэв хэрэглэгчид жаахан хэмнэхийг хүссэн ч (эсвэл толгой хуудасны төлбөрт өгөх хэдэн төгрөгөө) шүүлтүүр болгон толгой хуудас хэвлэдэг болохоор дээрх аргыг хэрэглээд үр дүн өгөхгүй.

Хэвлэгдэх бүр толгой хуудсыг хорихын тулд Толгой хуудасны тооцоо бүртгэл хэсэгт үзүүлсэн дараах бяцхан аргыг хэрэглэнэ: гаралтын шүүлтүүр бичээд түүгээрээ LPD-н үүсгэсэн толгой хэсгийг салгаж аван PostScript® хувилбарыг нь үүсгэх. Хэрэв хэрэглэгч `lpr -h` тушаал

хэрэглэвэл LPD толгой хэсэг үүсгэхгүй бөгөөд бас таны гаралтын шүүлтүүрийг ажиллуулахгүй. Бусад үед гаралтын шүүлтүүр нь LPD-с текстийг уншаад түүнд тохирсон PostScript® толгой хуудсыг хэвлэгчид өгдөг.

Хэрэв танд цуваа холболттой PostScript® хэвлэгч байгаа бол `lprps`-г та хэрэглэж болох бөгөөд энэ нь дээр ярьсан зүйлийг гүйцэтгэж чаддаг `psof` нэртэй гаралтын шүүлтүүртэй хамт ирдэг. Тэмдэглэж хэлэхэд, `psof` нь толгой хуудсанд тооцоо боддоггүй.

## 10.4.3. Сүлжээгээр хэвлэх

FreeBSD нь хэвлэх ажлыг алсад буй компьютер руу илгээж сүлжээгээр хэвлэхийг бас дэмждэг. Сүлжээгээр хэвлэх нь хоёр зүйлийг хамарч байдаг.

- Алсад буй компьютерт залгагдсан хэвлэгч рүү хандах. Та нэг хэвлэгчийг нэг компьютер дээр тохирох цуваа юм уу зэрэгцээ холболтоор хэвлэгчийг залгана. Дараа нь та LPD-г суулгаж алсад буй компьютер уг хэвлэгч залгагдсан компьютер руу хандаж болохыг тохируулна. Алсад буй компьютерт залгагдсан хэвлэгч хэсэгт үүнийг хэрхэн хийх талаар зааварласан бий.
- Шууд сүлжээнд залгагдсан хэвлэгч рүү хандах. Ийм хэвлэгч нь ердийн цуваа юм уу эсвэл зэрэгцээ холболтын интерфэйсээс гадна (эсвэл оронд нь) сүлжээний залгууртай байдаг. Ийм хэвлэгч дараах маягаар ажилладаг:
  - Хэвлэгч LPD-н дамжуулах протоколыг таньж ойлгох ба алсаас ирүүлсэн ажлуудад дараалал үүсгэж чаддаг. Ийм тохиолдолд уг хэвлэгч нь өөр дээрээ LPD суулгасан ердийн компьютер шиг ажиллах болно. Алсад буй компьютерт залгагдсан хэвлэгчид хэсэгт иймэрхүү хэвлэгчийг хэрхэн тохируулах талаар харна уу.
  - Хэвлэгч нь сүлжээгээр өгөгдлийг урсгал мэт хүлээж авдаг тийм төрлийн холболт дэмждэг байж болох юм. Энэ тохиолдолд та нэг компьютерт уг хэвлэгчийг "залгаад" суулгасан компьютер нь ирж буй ажлуудыг түр хадгалах хариуцлагыг нь дааж хэвлэгч рүүгээ өгөгдлийг урсгаж байхаар тохируулах хэрэгтэй. Сүлжээний өгөгдлийн урсгал холболтын төхөөрөмжтэй хэвлэгчид хэсэгт ийм төрлийн хэвлэгчийг хэрхэн суулгах талаар зааварласан байгаа.

### 10.4.3.1. Алсад буй компьютерт залгагдсан хэвлэгчид

LPD-д өөрт нь өөр газар LPD (эсвэл LPD-тэй зохимжтой) ажиллаж байгаа машин руу хэвлэх ажлыг илгээж чаддаг ажиллагаа байдаг. Энэ шинж чанарыг ашиглаж та өөр компьютерт хэвлэгчээ залгаад түүнд өөр газраас хандаж болох боломж нээж өгч байна. Энэ арга нь мөн LPD дамжуулах протоколыг ойлгодог хэвлэгчид дээр бас ажиллана .

Иймэрхүү алсаас хэвлэх ажиллагааг гүйцэтгэхийн тулд эхлээд хэвлэгчээ *хэвлэгчийн эзэн* болгож нэг машинд Хэвлэгч суулгах энгийн явц хэсэгт заасны дагуу залгах хэрэгтэй. Шаардлагатай бол Хэвлэгчийн нэмэлт тохиргоо хэсэгт заасан тохиргоог хийж болно. Хэвлэгчээ хэвлэж байгааг нь магадлаад LPD тохиргоог зөвшөөрсөн эсэхийг шалгах хэрэгтэй. Мөн *remote host* дотор *local host* нь LPD үйлчилгээг хэрэглэх эрхтэй эсэхийг шалгаарай (Алсад буй хостод хэвлэх хязгаарлалт хийх хэсэгт харна уу).

Хэрэв та сүлжээний карттай бөгөөд LPD протокол дэмждэг хэвлэгч хэрэглэж байгаа бол дээр өгүүлснээр *хэвлэгчийн эзэн* болон *хэвлэгчийн нэр* нь уг тохируулж буй хэвлэгчийн нэр байх

ёстой. Хэвлэгчтэй хамт ирсэн үйлдвэрлэгчийнх нь гарын авлагаас хэвлэгчийнх нь тухай эсвэл сүлжээний төхөөрөмжийнх нь тухай мэдээллийг хараарай.

Хэрэв та Hewlett Packard Laserjet хэвлэгч хэрэглэж байгаа бол `text` нэртэй хэвлэгч нь автоматаар LF-г CRLF руу хувиргадаг болохоор hpif скрипт хэрэглэх шаардлагагүй.

Дараа нь хэвлэгч рүү хандах гэж буй нөгөө машины /etc/printcap файл дотор дараах зүйлсийг оруулж өгдөг:

1. Хүссэн нэрээ оруулна. Алсад буй компьютерийн нэрэнд давхар нэр өгч оруулан амарчилж болох юм.
2. `lp` тохиргоог хоосон орхино. (`:lp=:`).
3. Түр хадгалах сан үүсгэж түүний байршлыг `sd` хэсэгт зааж өгдөг. LPD нь алсад буй хэвлэгч рүү ажлуудыг илгээхээсээ өмнө энэ санд хадгалдаг.
4. Хэвлэгчийн эзний нэрийг `rm` хэсэгт бичиж өгнө.
5. *Хэвлэгчийн эзэнд* холбогдсон хэвлэгчийн нэрийг `rp` хэсэгт оруулна.

Ингээд болох нь тэр. Та хувиргах шүүлтүүр, цаасны хэмжээ гэх мэтийг /etc/printcap файл дотор оруулах шаардлагагүй.

Энд нэг жишээ татъя. `rose` нэртэй машин `bamboo` болон `rattan` гэсэн хоёр хэвлэгчтэй. Бид одоо `orchid` машиныг уг хэвлэгчид рүү хандаж болдог болгож тохируулцгаая. Энд ( Толгой хуудсыг зөвшөөрөх хэсгээс дахин авав) `orchid` машины /etc/printcap файл хэрхэн харагдаж байгааг харуулав. Уг машинд өмнө нь `teak` нэртэй машин тохируулагдсан байгаа бөгөөд бид `rose` машины хоёр хэвлэгчийг нэмэх болно:

```
#
#  /etc/printcap for host orchid - added (remote) printers on rose
#

#
#  teak is local; it is connected directly to orchid:
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0:sd=/var/spool/lpd/teak:mx#0:\
        :if=/usr/local/libexec/ifhp:\
        :vf=/usr/local/libexec/vfhp:\
        :of=/usr/local/libexec/ofhp:

#
#  rattan is connected to rose; send jobs for rattan to rose:
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
    :lp=:rm=rose:rp=rattan:sd=/var/spool/lpd/rattan:

#
#  bamboo is connected to rose as well:
```

```
#
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :lp=:rm=rose:rp=bamboo:sd=/var/spool/lpd/bamboo:
```

Дараа нь бид orchid дээр түр хадгалах сан нэмэх хэрэгтэй:

```
# mkdir -p /var/spool/lpd/rattan /var/spool/lpd/bamboo
# chmod 770 /var/spool/lpd/rattan /var/spool/lpd/bamboo
# chown daemon:daemon /var/spool/lpd/rattan /var/spool/lpd/bamboo
```

Одоо orchid дээрх хэрэглэгчид rattan болон bamboo хэвлэгчдийг хэрэглэж чадна. Жишээ нь orchid дээрх хэрэглэгч дараах тушаал өгвөл:

```
% lpr -P bamboo -d sushi-review.dvi
```

orchid дээрх LPD систем хэвлэх ажлуудыг /var/spool/lpd/bamboo гэсэн түр хадгалагч руу хуулна. Энэ хэвлэгдэх ажил нь DVI файл байсныг та анзаарна уу. rose машинд bamboo түр хадгалах сан үүсэнгүүт хоёр машины LPD-ууд хоорондоо уг файлыг rose машин уруу дамжуулдаг. Дараа нь уг файл rose-н дараалалд очиж зогсоод хэвлэгдэх хүртлээ хүлээгдэнэ. Энэ файл DVI-с PostScript® уруу (bamboo нь PostScript® хэвлэгч учир) rose дээр хувиргагдана.

#### 10.4.3.2. Сүлжээгээр өгөгдлийг урсгал хэлбэрээр дамжуулдаг холболттой хэвлэгчид

Ихэнх тохиолдолд, хэвлэгчдэд зориулж сүлжээний төхөөрөмж авах үед танд хоёр сонголт бий: түр хадгалагчтай (илүү үнэтэй), эсвэл цуваа юм уу зэрэгцээ холболтоор холбогдсон мэт өгөгдлийг хэвлэгч рүү шууд илгээдэг (хямд үнэтэй). Энэ хэсэгт хямд үнэтэйг нь авч хэлэлцэнэ. Үнэтэйг нь Алсад буй компьютерт холбогдсон хэвлэгчид хэсгээс хараарай.

/etc/printcap файлд ямар цуваа юм уу эсвэл ямар зэрэгцээ холболт хэрэглэхийг зааж өгдөг бөгөөд (хэрэв та цуваа холболтоор холбосон бол ) ямар давтамжаар илгээх, ямар урсгал хянагч ашиглах, илгээх завсарлагаа, шинэ мөр тэмдгийг хэрхэн хувиргах гэх мэтчилэн маш олон зүйлийг тохируулж болно. Хэрэв хэвлэгч TCP/IP порт юм уу эсвэл өөр сүлжээний порт чагнаж байгаа хэвлэгчийг энд зааж өгөх боломжгүй.

Сүлжээнд холбогдсон хэвлэгч рүү илгээхийн тулд текст болон хувиргах шүүлтүүрээр дуудагдаж болдог холболтын програм та бичих хэрэгтэй. Энд нэг жишээ үзүүлье. netprint нэртэй скрипт нь бүх өгөгдлийг ердийн оролтоос аваад сүлжээнд холбогдсон хэвлэгч рүү илгээх болно. Хэвлэгч холбогдсон компьютерийн нэрийг нь эхний утгаар авах бөгөөд хоёр дахь утгаараа чагнаж буй портын дугаарыг netprint-д өгнө. Тэмдэглэж хэлэхэд, энэ нь зөвхөн нэг чиглэлтэй холболт (FreeBSD-с хэвлэгч рүү) бөгөөд ихэнх сүлжээний хэвлэгчид хоёр тийш чиглэсэн холболтыг дэмждэг болохоор тэр чанарыг нь та мөн ашиглаж болох юм (хэвлэгчийн төлөв байдлыг мэдэх, бүртгэл хийх гэх мэт).

```
#!/usr/bin/perl
#
#  netprint - Text filter for printer attached to network
```

```
#  Installed in /usr/local/libexec/netprint
#
$#ARGV eq 1 || die "Usage: $0 <printer-hostname> <port-number>";

$printer_host = $ARGV[0];
$printer_port = $ARGV[1];

require 'sys/socket.ph';

($ignore, $ignore, $protocol) = getprotobyname('tcp');
($ignore, $ignore, $ignore, $ignore, $address)
    = gethostbyname($printer_host);

$sockaddr = pack('S n a4 x8', &AF_INET, $printer_port, $address);

socket(PRINTER, &PF_INET, &SOCK_STREAM, $protocol)
    || die "Can't create TCP/IP stream socket: $!";
connect(PRINTER, $sockaddr) || die "Can't contact $printer_host: $!";
while (<STDIN>) { print PRINTER; }
exit 0;
```

Дараа нь бид энэ скриптийг янз бүрийн шүүлтүүртэй хамт хэрэглэж болно. Сүлжээнд Diablo 750-N туузан хэвлэгч холбогдсон байна гэж авч үзье. Хэвлэгч хэвлэгдэх өгөгдлийг 5100 гэсэн портын дугаар дээр хүлээж авна. Хэвлэгчийн холбогдсон компьютерийн нэр нь `scrivener`. Одоо энэ хэвлэгчид дараах текст шүүлтүүр хэрэглэе:

```
#!/bin/sh
#
#  diablo-if-net - Text filter for Diablo printer `scrivener' listening
#  on port 5100.   Installed in /usr/local/libexec/diablo-if-net
#
exec /usr/libexec/lpr/lpf "$@" | /usr/local/libexec/netprint scrivener 5100
```

### 10.4.4. Хэвлэгч хэрэглэх хязгаарлалт

Энэ хэсэгт хэвлэгч хэрэглэлтийг хязгаарлах талаар өгүүлэх болно. LPD систем хэвлэгчийг гадаад сүлжээ юм уу дотоодоос хэн олон хувилах, хэр зэрэг том хэмжээний ажил хэвлэж болох, хэвлэгчийн дараалал хэр урт байх ёстой зэргийг зааж өгч болно.

#### 10.4.4.1. Олон хэвлэлтийг хязгаарлах

LPD нь хэрэглэгчдэд нэг файлыг олон хэвлэхийг амарчилж өгдөг. Хэрэглэгч нар `lpr -#5` (жишээ болгож) тушаалаар файлыг таван удаа хувилж чадна. Үүний сайн эсэх нь танаас хамаарна.

Хэрэв ингэж олон хувилах нь цаг болон цаасны дайсан юм байна гэж үзэж байгаа бол `-` сонголтыг `lpr(1)` дээр хорьж болох бөгөөд ингэхийн тулд `sc` сонголтыг `/etc/printcap` файл дотор оруулж өгнө. Хэрэв хэрэглэгч `-` сонголттой хэвлэх тушаал илгээвэл дараах мэдэгдлийг харах болно:

```
lpr: multiple copies are not allowed
```

Хэрэв та алсын хэвлэгч тохируулсан байвал (Алсын компьютерт холбогдсон хэвлэгчид хэсгээс харна уу) `sc` сонголтыг алсад буй компьютерийн /etc/printcap файл дотор хийж өгөхгүй бол хэрэглэгчид хүссэн тоогоороо хувилж чадсаар байх болно.

Энд нэг жишээ авцгаая. Дараах файл бол алсад буй `rose` компьютерийн /etc/printcap файл. Хэвлэгч `rattan` нь нэлээн өгөөмөр бөгөөд хэн хүссэнд нь хувилж өгдөг байхад лазер хэвлэгч `bamboo` нь нэлээн харамч болохоор өөртөө `sc` тохиргоог агуулж байна:

```
#
#  /etc/printcap for host rose - restrict multiple copies on bamboo
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan:\
        :lp=/dev/lpt0:\
        :if=/usr/local/libexec/if-simple:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:sc:\
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif:\
        :df=/usr/local/libexec/psdf:
```

Одоо `sc` тохиргоог мөн `orchid` компьютерийн /etc/printcap файл дотор оруулж өгөх хэрэгтэй (бид өөрсдөө энэ компьютер дээр байгаа болохоор `teak` хэвлэгч дээр олон хуулбар хийхийг хорих болно):

```
#
#  /etc/printcap for host orchid - no multiple copies for local
#  printer teak or remote printer bamboo
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0:sd=/var/spool/lpd/teak:mx#0:sc:\
        :if=/usr/local/libexec/ifhp:\
        :vf=/usr/local/libexec/vfhp:\
        :of=/usr/local/libexec/ofhp:

rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :lp=:rm=rose:rp=rattan:sd=/var/spool/lpd/rattan:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :lp=:rm=rose:rp=bamboo:sd=/var/spool/lpd/bamboo:sc:
```

`sc` тохиргоог хэрэглэснээр `lpr -#` тушаалын хэрэглээг хорих боловч lpr(1) тушаалын олон удаагийн хэрэглээ юм уу эсвэл нэг удаагийн хэвлэгдэх ажилд нэг файлыг дараах маягаар олон илгээхийг хорих боломжгүй:

```
% lpr forsale.sign forsale.sign forsale.sign forsale.sign forsale.sign
```

Үүнийг хорих олон арга бий бөгөөд та чөлөөтэй оролдон үзэх хэрэгтэй (үл тоомсорлох аргыг оруулаад).

#### 10.4.4.2. Хэвлэгч рүү хандах хандалтыг хязгаарлах

UNIX®-н бүлэглэл болон /etc/printcap файлын `rg` тохиргоог ашигласнаар ямар хэвлэгч рүү хэн хандаж юм хэвлэж болохыг та тохируулж болдог. Нэг тодорхой бүлэгт хэрэглэгчдээ хамруулж оруулаад тэр бүлгийнхээ нэрийг `rg` тохиргоонд бичиж өгөөд л болох нь тэр.

Хэрэв уг бүлэгт хамааараагүй хэрэглэгч (`root`-г оруулаад) тухайн хяналттай хэвлэгч рүү юм хэвлэхийг оролдох үед доорх мэдэгдлийг харуулах болно:

```
lpr: Not a member of the restricted group
```

`sc` (suppress multiple copies буюу олон хуулбарыг хорих) тохиргооны адилаар алсын компьютерууд таны хэвлэгч рүү хандахаар хэрэв та зөвшөөрсөн бол `rg`-г тохируулах хэрэгтэй болно. (Алсын компьютерт суугдсан хэвлэгчид хэсгээс харна уу)

Жишээ нь, ямар ч хэрэглэгч `rattan` хэвлэгч рүү хандаж чадахаар, харин `artists` бүлгийн хэрэглэгчид зөвхөн `bamboo` хэвлэгчийг хэрэглэж болно гэж үзье. Энд бидний танил `rose` компьютерийн /etc/printcap файлыг харуулж байна:

```
#
#  /etc/printcap for host rose - restricted group for bamboo
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan:\
        :lp=/dev/lpt0:\
        :if=/usr/local/libexec/if-simple:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:sc:rg=artists:\
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif:\
        :df=/usr/local/libexec/psdf:
```

/etc/printcap файлын ( `orchid` машины) тохиргоог орхие. Мэдээж `orchid` дээрх хэн гуай ч `bamboo` -г хэрэглэж чадахгүй. Магадгүй `orchid` дээрх онцгой хэрэглэгчдийг хэвлэгч рүү хандах эрх олгож болох юм. Үгүй ч байсан болох юм :)

Хэвлэгч бүрд зөвхөн нэг л хязгаарлах бүлэг байж болно.

#### 10.4.4.3. Хэвлэгдэхээр илгээгдэх файлын хэмжээг хязгаарлах

Хэрэв танд хэвлэгч хэрэглэх эрхтэй хэвлэгчид байгаа бол тэдгээрийн хэвлэх файлд нь хэмжээг нь хязгаарлах тохиргоо хийхийг хүсэж болох юм. Хэдийгээр файл систем дээрх хэвлэгчийн түр хадгалагчийн санд хангалттай зай байгаа ч гэсэн хэрэглэгч бүрд тохирсон тодорхой хэмжээг зааж өгөх шаардлага гарч магадгүй.

LPD нь `mx` тохиргоонд зааж өгсөн байтын хэмжээгээр хэвлэгдэх ажлын хэмжээг хязгаарлах боломж өгдөг. Хэмжих нэгж нь `BUFSIZ`-д зааж өгсөн багцын хэмжээ байдаг ба нэг багц нь 1024 байт байдаг билээ. Хэрэв та энэ тохиргоонд тэгийн тоо бичих юм бол файлын хэмжээнд хязгаар тавихгүй болно. Гэхдээ та ямар ч `mx` тохиргоо зааж өгөөгүй бол анхдагч хязгаарлалт нь 1000 багц байдаг.

Энэ хязгаарлалт нь хэвлэгдэх гэж буй *файлд* хамаатай болохоос хэвлэгдэх бүх ажлуудыг *хязгаарлана гэсэн үг биш*.

LPD нь хэвлэгдэх файл нь хязгаарласан хэмжээнээс их файл ирвэл татгалздаггүй харин хязгаар хүртэлх хэмжээг нь хэвлэх дараалалд оруулж өгдөг бөгөөд тэр хэсэг нь хэвлэгддэг. Үлдсэн хэсэг нь цуцлагдах болно. Иймэрхүү хязгаарлалт нь зөв буруу эсэх нь таны шийдэл билээ.

Бидний жишээ болсон `rattan` болон `bamboo` нар дээрээ хязгаарлалт хийж үзэцгээе. `artists`-н PostScript® файлууд их хэмжээтэй байх хандлагатай байдаг болохоор тэдгээрийг таван мегабайтаас хэтрэхгүй гэж хязгаарлая. Энгийн текст файлд харин ямар ч хязгаарлалт өгөхгүй байя:

```
#
#  /etc/printcap for host rose
#

#
#  No limit on job size:
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:mx#0:sd=/var/spool/lpd/rattan:\
        :lp=/dev/lpt0:\
        :if=/usr/local/libexec/if-simple:

#
#  Limit of five megabytes:
#
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:sc:rg=artists:mx#5000:\
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif:\
        :df=/usr/local/libexec/psdf:
```

Зөвхөн дотоод хэрэглэгчдэд ийм хязгаарлалт хэрэгжих болно гэдгийг дахин сануулъя. Хэрэв та алсад буй хэвлэгчид ийм хязгаарлалт хийвэл алсаас хандагдаж буй хэрэглэгчдэд энэ

хязгаарлалт хүчингүй. Үүнийг биелүүлэхийн тулд алсад буй машины /etc/printcap файл дотор `mx` тохиргоог зааж өгөх ёстой. Алсад буй компьютер дээр холбогдсон хэвлэгчид хэсгээс алсын хэвлэлт хийх талаар уншаарай.

Алсад буй хэвлэгчид хязгаарлалт тавих өөр нэг онцгой арга байдаг. Алсын хост дээр хэвлэх үйлдлийг хязгаарлах хэсгээс дэлгэрэнгүй уншина уу.

#### 10.4.4.4. Алсаас хост дээр хэвлэх үйлдлийг хязгаарлах

LPD нь алсын компьютераас илгээгдсэн хэвлэх ажлыг бас хянаж болдог боломж олгодог:

**Компьютерээр нь хязгаарлах**

Дотоод LPD руу ямар компьютер хандаж болохыг /etc/hosts.equiv болон /etc/hosts.lpd файлд зааж өгснөөр хянаж болдог. LPD нь саяны хоёр файлын аль нэгэнд нь хэвлэлт хүссэн компьютерийн нэр байгаа эсэхийг шалгадаг. Хэрэв нэр нь байхгүй бол LPD уг хүсэлтээс татгалздаг.

Уг файлын загвар бол тун энгийн. /etc/hosts.equiv файл нь мөн ruserok(3) протоколд хэрэглэгддэг бөгөөд rsh(1) болон rcp(1) зэрэг програмд нөлөөлдөг болохоор тун болгоомжтой хэрэглэхийг сануулж байна.

Жишээ нь, `rose` компьютер дээрх /etc/hosts.lpd файл:

```
orchid
violet
madrigal.fishbaum.de
```

Энэ нь тэгэхээр `rose` компьютер нь `orchid`, `violet`, болон `madrigal.fishbaum.de` зэрэг компьютераас хэвлэх хүсэлт хүлээж авч болно гэсэн үг. Хэрэв эдгээрээс өөр компьютерууд `rose`-н LPD руу хандахыг хүсвэл тэр хүсэлт нь татгалзагдах болно.

**Хэмжээгээр нь хязгаарлах**

Түр хадгалагч санд байрлаж болох файлуудын хэмжээ нь хэд хүртэл байж болохыг хянаж болдог. Дотоод хэвлэгчид зориулсан түр хадгалах сан дотор minfree нэртэй файл үүсгэх хэрэгтэй. Уг файл дотор алсаас ирсэн хэвлэгдэх ажлын хэмжээ нь хичнээн дискний багцаар (512 байт) байж болох дугаарыг бичиж өгдөг.

Ингэснээр алсаас хэрэглэх хэрэглэгчид таны дискийг дүүргэхээс сэргийлнэ. Та энэ файлыг ашигласнаар бас тодорхой дотоод хэрэглэгчдэд давуу эрхийн дараалал тогтоож болдог ба minfree файлд зааж өгсөн хэмжээнээс хэтэрсэн хэмжээгээр нь дараалал тогтоодог.

Жишээ болгож minfree файлыг `bamboo` хэвлэгчид зориулж нэмж үзье. Бид /etc/printcap файл дээр дадлага хийх бөгөөд хэвлэгчид зориулсан түр хадгалагч тохируулсан хэсгийг нь олж харцгаая. Ингээд `bamboo`-н оруулгыг харуулъя:

```
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo:sc:rg=artists:mx#5000:\
        :lp=/dev/ttyu5:ms#-parenb cs8 clocal crtscts:rw:mx#5000:\
```

```
        :if=/usr/local/libexec/psif:\
        :df=/usr/local/libexec/psdf:
```

Түр хадгалах санг `sd` тохиргооны хэсэгт зааж өгсөн байна. Одоо LPD-н алсаас илгээгдсэн хэвлэгдэх ажилд зориулж гурван мегабайт хэмжээг (6144 дискний багц) зааж өгцгөөе. :

```
# echo 6144 > /var/spool/lpd/bamboo/minfree
```

**Хэрэглэгчээр нь хязгаарлах**

Та ямар хэрэглэгч алсаас хэвлэх хүсэлт тавьж болохыг /etc/printcap файл дотор `rs` тохиргооны хэсэгт зааж өгдөг. Хэрэв дотоод хэвлэгчийн оруулгын хэсэгт `rs` гэж харагдвал LPD нь алсаас хэвлэх хүсэлт илгээсэн компьютерийн хэрэглэгч *хэрэв* дотоод компьютер дээрх нэвтрэх эрхтэй ижилхэн нэртэй хэрэглэгч байвал зөвшөөрөл өгч хэвлүүлнэ. Бусад тохиолдолд LPD нь тухай ажлыг хэвлэхээс татгалзана.

Энэ тохиргоо нь (жишээлбэл) нэг хэвлэгчийг хувааж хэрэглэдэг олон салбартай байгууллагад ашиглагдаж болох бөгөөд зарим хэрэглэгчид нь хэд хэдэн салбарт харьяалагддаг бол уг хэрэглэгчид эрх өгснөөр өөрийнхөө байгаа салбараас өөр салбарт буй хэвлэгчид хэвлэх ажил илгээх маягаар ашиглаж болох юм. Хэрэв та тэдгээр хэрэглэгчдийг *зөвхөн* таны хэвлэгчийг л хэрэглэх бөгөөд бусад зүйлсийг ашиглуулахгүй гэж бодож байгаа бол тэд нарт эхлэл сан байхгүй бөгөөд /usr/bin/false гэсэн хэрэглэгдэхгүй бүрхүүлтэй "хязгаарлагдмал" бүртгэл үүсгэх хэрэгтэй.

### 10.4.5. Хэвлэгчийн хэрэглээнд бүртгэл тооцох

За тэгэхээр та хэвлэлт болгонд мөнгө тооцох шаардлага байг. Цаас болон хэвлэх хорнуудыг мөнгөөр авдаг болохоор яагаад болохгүй гэж? Мөн дээрээс нь хэвлэгчийн эд анги байнга хөдөлж эвдрэх магадлалтай байдаг болохоор хэвлэгчээ тордох нь бас үнэтэй билээ. Та хэвлэгчээ арчилж хуудас болгондоо (эсвэл хуудасны хэсэг бүрд, гарчиг бүрд, эсвэл юу ч байсан яахав) мөнгө тооцохыг хүсэв. Тэгэхээр та яаж хэвлэгдсэн хуудас болгонд мөнгө тооцож чадах вэ?

Гэхдээ, жаахан таагүй мэдээ нь юу вэ гэхээр LPD түр хадгалагч гуай иймэрхүү талын үйлчилгээ хийхдээ тун тааруу юм. Иймэрхүү бүртгэл тооцоо нь хэвлэгчээ хэрхэн ашиглахаас их хамаардаг бөгөөд хэвлэгдэх загвар, мөн *таны* хэвлэгчээсээ мөнгө олборлох шаардлагаас их шалтгаална.

Бүртгэл тооцоо хийхийн тулд та хэвлэгчийнхээ текст шүүлтүүрийг өөрчлөх хэрэгтэй (энгийн текст ажлаас мөнгө тооцохын тулд) бөгөөд мөн хувиргагч шүүлтүүрүүдийг бас (мөн бусад файл төрлүүдээс) өөрчлөх шаардлагатай. Энгийн шүүлтүүр ашиглаад та бүртгэл тооцоо хийж чадахгүй. Шүүлтүүр хэсгийг харна уу.

Ерөнхийдөө бүртгэл тооцоо хийж болох хоёр арга бий:

- *Үечлэн бүртгэл тооцох* гэдэг бол тун амархан бөгөөд байнга хэрэглэгддэг арга. Хэн нэгэн хэвлэх үед хэрэглэгчийн нэр, компьютерийн нэр, мөн хэвлэсэн хуудасны дугаар зэргийг бүртгэл файлд хадгалдаг. Хэвлэгчдээ зориулсан нэгжийнх нь тооцоог та сараар, эсвэл жилээр, эсвэл өөрийнхөө заасан хугацаагаар хэрэглэгч бүрийн хувьд тохирсон жагсаалт

авч болдог. Ингэж бүртгэл тооцоо хийгээд уг файлыг цэвэрлэж дараагийн үеэр нь мөн тооцоо хийхээр бэлтгэнэ.

- *Цагаар бүртгэж тооцох* гэдэг нь жаахан хэцүү болохоор нэлээн бага хэрэглэгддэг. Энэ арганд хэрэглэгчдийг хэвлэж эхлэнгүүт тооцоо хийдэг шүүлтүүр байдаг. Дискний хэмжээ хязгаарлагчтай ижилхэн энэ бүртгэл тооцоо нь харьцангуй шулуухан гүйцэтгэгддэг. Та хэрэглэгчдийн бүртгэлд заагдсан хэмжээ дуусах дөхөхөөр хэвлүүлэхгүй болгож болохоос гадна хэрэглэгчдэд өөрсдийн "хэвлэх хувийг" шалгаж тохируулах боломжийг олгон тохируулж болох юм. Гэвч энэ арга нь хэрэглэгчийн бүртгэл болон тэдгээрийн хязгаарласан хэмжээг мөрдөхийн тулд зарим өгөгдлийн бааз руу хандах шаардлагатай байдаг.

Та шаардагдах шүүлтүүрүүдийг, мөн бүртгэж тооцоо хийх кодыг нь хангаж чадаж л байвал LPD түр хадгалагч систем нь өмнө тайлбарласан хоёр аргыг дэмждэг. Бүртгэж тооцох арганд бас онцгой тал байдаг. Жишээ нь та үечилж үү эсвэл цагаар тооцох уу гэдгээ сонгож болно. Та ямар мэдээллийг бүртгэж бичихээ сонгоно: хэрэглэгчийн нэр, компьютерийн нэр, хэвлэсэн төрөл, хэвлэгдсэн хуудасны тоо, загвар, хэр удаан хэрэглэсэн гэх мэт. Та шүүлтүүрүүдийг өөрчилж эдгээр мэдээллийг хадгалдаг болгох хэрэгтэй.

#### 10.4.5.1. Хэвлэлтэд тооцоо хийх түргэн бөгөөд бохир арга

FreeBSD нь бүртгэл тооцоо хийхэд хэрэглэгдэх хоёр үечлэн тооцоо хийдэг програмтай хамт ирдэг. Тэдгээр нь `lpf` текст шүүлтүүрүүд бөгөөд lpf: Текст шүүлтүүр хэсэгт тайлбарлагдсан байгаа. Нөгөөх нь pac(8) бөгөөд энэ нь хэвлэлтийн бүртгэл тооцоо хийх файлд буй оруулгуудыг цуглуулдаг програм.

Шүүлтүүрийн хэсэгт тайлбарласны дагуу (Шүүлтүүрүүд), LPD нь текст болон хувиргагч шүүлтүүрийг бүртгэл файлтай нь хамт шүүлтүүрийн тушаалын мөрнөөс эхлүүлдэг. Шүүлтүүр нь ингэж тушаалын мөрнөөс авсан утгаар хаана бүртгэл файлд оруулах вэ гэдгээ мэддэг. Энэ файлын нэр нь /etc/printcap файл доторх `af` тохиргоонд заагдсан байдаг ба хэрэв файлынх нь зам нь яг нарийвчилж заагдаагүй байгаа бол түр хадгалагчийн сангийн байрлалтай харьцангуй хэлбэрээр мөн зааж өгч болно.

LPD нь `lpf`-г хуудасны өргөн болон өндрийн утгатай нь (`pw` болон `pl` тохиргуулаас авч) эхлүүлдэг. `lpf` нь эдгээр утгыг ашиглаж хичнээн хуудас хэвлэгдэх вэ гэдгийг тодорхойлдог. Файлыг хэвлэгч рүү илгээсний дараа бүртгэл тооцооны файлд тооцоо хийх утга нь оруулагддаг. Энэ оруулга нь дараах маягтай байдаг:

```
2.00 rose:andy
3.00 rose:kelly
3.00 orchid:mary
5.00 orchid:mary
2.00 orchid:zhang
```

Та болж өгвөл хэвлэгч болгонд тус тусад нь бүртгэл тооцооны файл хөтлөх хэрэгтэй. Учир нь `lpf`-д ямар ч түгжиж зохицуулдаг шинж чанар байхгүй болохоор хоёр `lpf`-ууд нэг файлд хандах тохиолдолд мөргөлдөөн үүсэж болзошгүй. Тус тусад нь бүртгэл тооцооны файл хөтлөх амар арга бол /etc/printcap дотор `af=acct` гэсэн тохиргоог оруулах хэрэгтэй. Дараа нь хэвлэгчид зориулсан түр хадгалах сан бүрд acct гэсэн файл хөтлөгдөх болно.

Хэрэглэгчдээс хэвлэлтийн төлбөрийг шаардах үедээ pac(8) програмыг ажиллуулах хэрэгтэй. Тооцоо хийх хэвлэгчийнхээ түр хадгалах санд оронгуутаа pac гэсэн тушаал ажиллуулахад болно. Дараа нь доллараар тооцсон дараах маягийн үр дүнг харах болно:

```
  Login               pages/feet   runs    price
orchid:kelly                5.00    1   $  0.10
orchid:mary                31.00    3   $  0.62
orchid:zhang                9.00    1   $  0.18
rose:andy                   2.00    1   $  0.04
rose:kelly                177.00  104   $  3.54
rose:mary                  87.00   32   $  1.74
rose:root                  26.00   12   $  0.52

total                     337.00  154   $  6.74
```

pac(8)-тай хамт хавсруулж хэрэглэдэг сонголтуудыг жагсааж харуулъя:

**-Рхэвлэгч**

Энэ нь тооцоо бодох *хэвлэгч*ийг зааж өгнө. /etc/printcap файл доторх af сонголтод яг нарийвчилсан файлын зам зааж өгсөн үед л энэ нь хэрэгждэг.

**-c**

Хэрэглэгчдийн нэрээр биш үнээр нь эрэмбэлж харуулдаг.

**-m**

Тооцоо хийх файлаас компьютерийн нэрийг алгасаж харуулдаг. Ийм үед alpha компьютер дээрх smith нэртэй хэрэглэгч gamma компьютер дээрх smith нэртэй хэрэглэгчээс ялгардаггүй.

**-рүнэ**

/etc/printcap файлд буй pc тохируулгад зааж өгсөн үнийн оронд нэгж хуудас болгоныг *үнээр* үнэлж доллараар тооцдог бөгөөд эсвэл хоёр центээр ( анхдагч хэмжээ нь) тооцно. Та энэ *үнэ*-г бутархай тоон хэлбэрээр бас бичиж болно.

**-г**

Эрэмбэлэх дарааллыг эсрэгээр үйлдэнэ.

**-s**

Төлбөр тооцооны дүгнэсэн хуудас үүсгэж бүртгэл тооцооны файлыг богиносгодог.

***хэрэглэгч ...***

Зөвхөн зааж өгсөн *хэрэглэгчийн* бүртгэл тооцоог харуулна.

pac(8) боловсруулсан анхдагч тооцоонд янз бүрийн компьютераас хэвлэсэн хэрэглэгч бүрд хуудасных нь дугаарыг харуулдаг. Хэрэв танд компьютерийн нэр харуулах шаардлага байхгүй (хэрэглэгч янз бүрийн машинаас хэвлэсэн байж болно) гэж үзвэл pac -m гэсэн тушаалаар доорх маягаар үр дүнг харах болно:

```
  Login               pages/feet   runs    price
```

```
andy                    2.00    1   $  0.04
kelly                 182.00  105   $  3.64
mary                  118.00   35   $  2.36
root                   26.00   12   $  0.52
zhang                   9.00    1   $  0.18

total                 337.00  154   $  6.74
```

Долларын хэмжээг тооцохын тулд pac(8) нь /etc/printcap доторх `pc` тохиргооны утгыг ашигладаг (анхдагч утга нь хуудас болгонд 200, эсвэл 2 цент гэж байдаг). Энэ тохируулгад хуудас бүрд юм уу эсвэл алхам бүрд тооцох центүүдийг зааж өгдөг. Та энэ утгыг pac(8)-г ажиллуулахдаа `-p` гэсэн хавсралт сонголтоор дарж хэрэглэж болдог. `-p` сонголтод хэрэглэх утга нь центээр биш доллараар байх ёстой. Жишээ нь,

```
# pac -p1.50
```

гэснээр хуудас бүр нэг доллар тавин цент болж байна. Та энэ тохиргоог ашиглаж үнэхээр ашиг олж болох юм шүү.

Эцэст нь `pac -s` гэж тушааснаар дүгнэсэн тооцоог дүнгийн бүртгэл файл уруу хадгалах бөгөөд энэ файл нь хэвлэгчийн тооцооны файлын нэрийн ард нь `_sum` гэж залгагдсан нэртэй файл болж хадгалагдана. Дараа нь бүртгэлийн файлыг богиносгодог. Хэрэв та pac(8)-г дахин ажиллуулбал энэ нь дүгнэлт файлаас нийт утгыг уншиж аваад ердийн тооцооны файлаас авсан утган дээр нэмээд ерөнхий нийт нийлбэрийг харуулах болно.

#### 10.4.5.2. Хэвлэгдсэн хуудасны тоог яаж тооцох вэ?

Бүр алсаас бүртгэл хийлтийг зөв хийж гүйцэтгэхийн тулд ажил хэр их цаас ашигладгийг та тодорхойлж чаддаг байх хэрэгтэй. Энэ нь хэвлэгчийн бүртгэл хийлтийн үндсэн асуудал юм.

Цэвэр текст ажлуудын хувьд энэ асуудлыг шийдэх нь тийм ч хэцүү биш юм: та ажилд хичнээн мөр байгааг тоолж тэр тоогоо таны хэвлэгч нэг хуудсандаа хичнээн мөрийг дэмждэгтэй харьцуулах хэрэгтэй. Мөрүүдийг давхар хэвлэдэг файл дахь устгах тэмдэгтүүд эсвэл нэг буюу хэд хэдэн нэмэлт мөрүүд болдог урт логик мөрүүдийг бодолцохоо мартуузай.

Текст шүүлтүүр `lpf` (lpf: Текст шүүлтүүр хэсэгт танилцуулагдсан) нь бүртгэл хийхдээ эдгээр зүйлсийг бодолцдог. Бүртгэл хийх хэрэгтэй текст шүүлтүүр та бичиж байгаа бол `lpf`-ийн эх кодыг та магадгүй үзэхийг хүсэж болох юм.

Тэгэхээр та бусад файлын хэлбэршилтүүдтэй хэрхэн ажиллах вэ?

DVI-aac-LaserJet эсвэл DVI-aac-PostScript® руу хийх хөрвүүлэлтийн хувьд `dvilj` эсвэл `dvips` тушаалаас гарах оношлогооны гаралтыг та өөрийн шүүлтүүрээр оруулж хичнээн хуудаснууд хөрвүүлэгдсэнийг хайж олж болох юм. Бусад файлын хэлбэршилтүүд болон хөрвүүлэх програмуудын хувьд та эдгээртэй төстэй зүйлсийг хийж чадах ёстой.

Гэхдээ хэвлэгч нь тэдгээр бүх хуудаснуудыг жинхэнэдээ хэвлэхгүй байж болох сул тал энэ аргуудад ажиглагддаг. Жишээ нь цаас гацах, хор дуусах, эсвэл дэлбэрсэн хэр нь хэрэглэгчид

төлбөр ноогдуулсан хэвээр байж болох зэргийг дурдаж болох юм.

Тэгэхээр та юу хийж чадах вэ?

*Зөв* бүртгэл хийх цорын ганц *найдвартай* арга байдаг. Хичнээн цаас ашигладаг болохоо хэлж чаддаг хэвлэгч аваад цуваа шугамаар юм уу эсвэл сүлжээгээр залгах хэрэгтэй. Бараг бүх PostScript® хэвлэгчүүд үүнийг дэмждэг. Бусад загварууд бас ингэж чаддаг (жишээ нь сүлжээнд холбогдсон Imagen laser хэвлэгчүүд). Хэвлэгчүүдийг ажил бүрийг хэвлэснийхээ дараа хуудасныхаа хэрэглээг авч *зөвхөн* тэр утга дээрээ үндэслэн бүртгэлийн мэдээллээ бүртгүүлдэг болгохын тулд эдгээр хэвлэгчүүдийн хувьд шүүлтүүрүүдийг өөрчлөх хэрэгтэй. Мөр тоолох эсвэл алдаагүй файл шалгалт шаардлагагүй юм.

Мэдээж та үргэлж өгөөмөр байж бүх хэвлэлтийг үнэгүй болгож болох юм.

# 10.5. Хэвлэгчдийг ашиглах нь

Энэ хэсэгт таны FreeBSD дээр тохируулсан хэвлэгчийг хэрхэн хэрэглэх талаар өгүүлэх болно. Хэрэглэгчийн ашиглаж болох тушаалууд нь:

**lpr(1)**

Хэвлэх үйлдлийг гүйцэтгэдэг

**lpq(1)**

Хэвлэгчийн хэвлэх дарааллыг шалгадаг

**lprm(1)**

Хэвлэгчийн хэвлэх дарааллаас устгадаг

Энд мөн удирдаж зохицуулах тушаалууд байдаг. Тэр нь Хэвлэгчдийг зохицуулах хэсэгт өгүүлсэн lpc(8) тушаал юм.

lpr(1), lprm(1), мөн lpq(1) гэсэн эдгээр тушаалууд нь бүгдээрээ `-P хэвлэгчийн-нэр` гэсэн сонголт авдаг бөгөөд үүгээрээ /etc/printcap файлд буй хэвлэгч/дараалалтай ажилладаг. Энэ нь хэвлэгчийн ажлыг нэмэх, устгах эсвэл хэвлэгдэж байгаа ажлыг шалгах зэрэгт хэрэглэгддэг. Хэрэв та `-P` сонголт хэрэглэдэггүй бол `PRINTER` гэсэн орчны хувьсагчийн уттыг энэ тушаал хэрэглэдэг. Эцэст нь хэрэв танд `PRINTER` орчны хувьсагч зарлагдаагүй бол `lp` нэртэй хэвлэгчийг анхдагч хэвлэгч гэж ханддаг.

Энд *анхдагч хэвлэгч* гэдэг нь `PRINTER` орчны хувьсагч дотор буй хэвлэгч юм уу эсвэл уг орчны хувьсагчид хэвлэгч заагдаагүй байвал `lp` гэсэн нэртэй хэвлэгч байна гэж авч үзэхийг хэлж байгаа юм.

## 10.5.1. Хэвлэх ажиллагаа

Файлыг хэвлэхийн тулд:

```
% lpr filename ...
```

Энэ нь жагсааж өгсөн файлуудыг анхдагч хэвлэгч рүү илгээж хэвлэдэг. Хэрэв та ямар ч файл зааж өгөөгүй бол lpr(1) нь гарнаас оруулах утгуудыг хэвлэгч рүү илгээдэг. Жишээ нь дараах тушаал нь системийн чухал файлуудыг хэвлэнэ:

```
% lpr /etc/host.conf /etc/hosts.equiv
```

Хэвлэгчийг онцгойлж сонгохыг хүсвэл :

```
% lpr -P printer-name filename ...
```

Дараах жишээн дээр тухайн байгаа сангийн файлуудыг жагсааж `rattan` нэртэй хэвлэгч рүү илгээхийх харуулах болно:

```
% ls -l | lpr -P rattan
```

Ямар ч файлын жагсаалт өгөөгүй болохоор lpr(1) тушаал нь `ls -l` тушаалын гаралтыг өөрийнхөө оруулга гэж авч үзсэн байна.

lpr(1) тушаал нь файл хувиргалт хийх, олон хуулбарлаж хэвлэх гэх мэтчилэн маш олон төрлийн сонголт хүлээн авч хэвлэх үйлдлийнхээ загварыг өөрчилж чаддаг. Хэвлэх сонголтууд хэсгээс нэмэлт мэдээллүүдийг уншина уу.

### 10.5.2. Хэвлэж буй ажлыг шалгах

lpr(1) тушаалаар хэвлэгдэх зүйлс LPD түр хадгалагч дээр очдог бөгөөд тэдгээрийг ерөнхийд нь "хэвлэгдэх ажил" гэж нэрлэдэг. Хэвлэгч болгонд хэвлэгдэх ёстой ажлууд жагсаагдсан байдаг бөгөөд таны хэвлэхийг хүссэн юм тань бусад хэрэглэгчидтэй адилхан хэвлэгдэх дараалалд ээлжээ хүлээгээд зогсож байдаг. Хэвлэгч нь ирсэн зүйлсийг хэн-түрүүлж-ирнэ түүнийг-түрүүлж-хэвлэнэ гэсэн зарчим баримталж хэвлэдэг.

Анхдагч хэвлэгчийн дарааллыг харуулахын тулд lpq(1) тушаалыг бичдэг. Хэрэв хэвлэгчээр нь сонгож харахыг хүсвэл `-P` сонголт хэрэглэдэг. Жишээлбэл

```
% lpq -P bamboo
```

гэсэн тушаал нь `bamboo` нэртэй хэвлэгч дээр байгаа хэвлэх дарааллыг хардаг. Доорх хэсэгт `lpq` тушаалын гаралтыг харуулав:

```
bamboo is ready and printing
Rank   Owner    Job  Files                              Total Size
active kelly    9    /etc/host.conf, /etc/hosts.equiv   88 bytes
2nd    kelly    10   (standard input)                   1635 bytes
3rd    mary     11   ...                                78519 bytes
```

Энэ нь `bamboo` хэвлэгчид гурван файл жагсаагдан ээлжээ хүлээж байгааг харуулсан байна. Эхний ээлжит ажлыг kelly хэрэглэгч илгээсэн бөгөөд "хэвлэх дугаар нь" 9 байна. Хэвлэх ажил болгон давтагдаагүй өөрийн гэсэн дугаар авдаг. Та хэвлэх дарааллын энэ дугаарыг хэрэгсэхгүй байж болох ч гэсэн хэрэв хэвлэх ажлыг цуцлах хэрэг гарвал уг дугаарыг мэдэж байх нь танд хэрэгтэй билээ. Энэ талаар Хэвлэх ажлыг устгах хэсгээс харна уу.

Ес гэсэн дугаартай хэвлэх ажил нь хоёр ширхэг файлаас бүтсэн байна. lpr(1) тушаалд хэрэв олон файл өгсөн бол тэдгээрийг нэг ажилд тооцдог. Энэ ажил нь одоогийн идэвхитэй ажил ("Rank" нэртэй баганад `active` гэж бичсэн байгааг анхаарна уу) бөгөөд хэвлэгч яг одоо түүнийг хэвлэж байна гэсэн үг. Хоёр дахь ажил нь стандарт оруулгаас lpr(1) тушаал уруу илгээгдсэн ажил байна. Гурав дахь ажил нь `mary` хэрэглэгчээс илгээгдсэн бөгөөд нэлээн их хэмжээнийх юм байна. Файлын нэрийн урт нь их байгаа учраас lpq(1) нь түүнийг гурван цэгээр товчилж харуулсан байна.

lpq(1) тушаалын гаралтын хамгийн эхний мөр бас чухал утга илэрхийлдэг нь: хэвлэгч яг одоо юу хийж байгааг (ядаж хэвлэгчийн юу хийж байгааг LPD нөхөр бодож байгааг) харуулдаг.

lpq(1) тушаал нь мөн `-l` сонголтыг дэмждэг бөгөөд энэ нь уртасгасан жагсаалтыг харуулдаг юм. Доор `lpq -l` тушаалын гаралтыг харуулав:

```
waiting for bamboo to become ready (offline ?)
kelly: 1st                [job 009rose]
       /etc/host.conf                     73 bytes
       /etc/hosts.equiv                   15 bytes

kelly: 2nd                [job 010rose]
       (standard input)                   1635 bytes

mary: 3rd                                 [job 011rose]
      /home/orchid/mary/research/venus/alpha-regio/mapping 78519 bytes
```

### 10.5.3. Хэвлэх ажлыг устгах

Хэрэв та хэвлэе гэсэн бодлоо өөрчилбөл lprm(1) тушаалаар хэвлэгдэх ажлыг устгаж болно. Мөн та lprm(1) тушаалаар хэвлэгдэж байгаа ажлыг бас устгаж болдог боловч зарим хэсэг нь хэвлэгдэж магадгүй юм.

Анхдагч хэвлэгчээс ажил устгахын тулд эхлээд lpq(1) тушаал хэрэглэж дарааллынх нь дугаарыг нь мэддэг. Тэгээд дараа нь:

```
% lprm job-number
```

Хэвлэгдэх ажлыг хэвлэгчийн нэрийг нь онцгойлж хандан устгахын хүсвэл `-P` сонголт хэрэглэнэ. Дараах тушаал нь `bamboo` нэртэй хэвлэгчээс 10 дугаартай ажлыг устгана:

```
% lprm -P bamboo 10
```

lprm(1) тушаалын хэд хэдэн товчлол бий:

**lprm -**

Танд хамаатай бүх хэвлэгдэх ажлыг устгадаг (анхдагч хэвлэгчээс).

**lprm *хэрэглэгч***

*Хэрэглэгч*-д хамаатай бүх ажлыг устгана (анхдагч хэвлэгчээс). Супер хэрэглэгч мэдээж бүх хэрэглэгчдийн ажлыг устгаж болох бөгөөд хэрэв та биш бол зөвхөн өөрийнхөө л ажлыг устгана.

**lprm**

Ямар ч дарааллын дугаар, хэрэглэгчийн нэр, эсвэл - сонголт зэргийг оруулаагүй бол танд хамаатай яг одоо хэвлэгч дээр идэвхитэй байгаа ажлыг lprm(1) устгадаг. Супер хэрэглэгч бол яг одоо идэвхитэй байгаа хэний ч дараалал байсан гэсэн устгана.

Дээрх товчлолуудыг -P сонголт ашиглан хэвлэгчийн нэр зааж өгөн ямар нэгэн хэвлэгч рүү онцгойлон хандаж болдог. Жишээлбэл дараах тушаал нь rattan хэвлэгчээс тухайн хэрэглэгчийн бүх дарааллыг устгана:

```
% lprm -P rattan -
```

Хэрэв та сүлжээний орчинд ажиллаж байгаа бол lprm(1) тушаал нь өөр ижил нэртэй хэвлэгч өөр машинд залгаатай байсан ч зөвхөн илгээсэн компьютерийн л дарааллыг устгахад зөвшөөрдөг. Доорх жишээнд үүнийг тодруулав:

```
% lpr -P rattan myfile
% rlogin orchid
% lpq -P rattan
Rank   Owner      Job  Files                         Total Size
active seeyan     12    ...                          49123 bytes
2nd    kelly      13   myfile                        12 bytes
% lprm -P rattan 13
rose: Permission denied
% logout
% lprm -P rattan 13
dfA013rose dequeued
cfA013rose dequeued
```

## 10.5.4. Энгийн текстээс өөр зүйлс хэвлэх нь: Хэвлэх сонголтууд

lpr(1) тушаал нь текст загварчлах, зураг файлыг өөр хэлбэрт хувиргах, олон хуулбар хийх, хэвлэгдэж байгаа файлууд уруу хандах гэх мэтчилэн маш олон сонголттой. Энэ хэсэгт тэр

тухай зааварлах болно.

#### 10.5.4.1. Хэлбэршүүлэх болон хувиргах сонголтууд

Дараах lpr(1)-н сонголтууд нь хэвлэгдэх ажлын хэлбэршүүлэлтийг хянадаг. Эдгээр сонголтуудыг хэрэв таны хэвлэх зүйлс тань энгийн текст хэлбэртэй биш юм уу эсвэл текстээ pr(1) хэрэгсэл ашиглан өөрчилж хэвлэх үед хэрэглэнэ.

Жишээлбэл дараах тушаал нь fish-report.dvi нэртэй DVI төрлийн файлыг (TeX төрлийн бичилтийн системийн) `bamboo` хэвлэгч рүү явуулна:

```
% lpr -P bamboo -d fish-report.dvi
```

Энэ сонголт нь хэвлэгдэх гэж буй бүх файлд хэрэгждэг болохоор жишээ нь та DVI болон ditroff файлуудыг хамт хэвлэгч рүү илгээж болохгүй. Түүний оронд файлуудыг тусад нь тусдаа сонголттой илгээх хэрэгтэй.

`-p` болон `-T` сонголтуудаас бусад бүх сонголтуудад хэвлэгч дээр суугдсан хөрвүүлж хувиргадаг шүүгч ажиллагаа шаардагддаг. Жишээ нь `-d` сонголт DVI хөрвүүлж хувиргадаг шүүлт шаарддаг. Хувиргагч шүүлтүүрүүд хэсэгт тодорхой өгүүлсэн буй.

**`-c`**

cifplot файлуудыг хэвлэнэ.

**`-d`**

DVI файлуудыг хэвлэнэ.

**`-f`**

FORTRAN текст файлуудыг хэвлэнэ.

**`-g`**

plot төрлийн зураг график файлуудыг хэвлэнэ.

**`-i тоо`**

Гаралтыг *тоо*гоор зай авч хэвлэнэ. Хэрэв та *тоо* оруулахгүй тушаавал 8 гэсэн зайгаар хэвлэнэ. Энэ сонголт зөвхөн тодорхой хэдэн шүүлтүүртэй л ажилладаг.

`-i` сонголт болон тоо хоёрын завсар ямар ч зай байж болохгүй.

**`-l`**

Текст файлд буй зарим хянагч тэмдэгтүүдтэй нь шууд үсэгчлэн хэвлэнэ.

**`-n`**

ditroff (төхөөрөмж болгоноос хамааралгүй troff) өгөгдлийг хэвлэнэ.

**-p**

Хэвлэхээсээ өмнө энгийн текстийг pr(1) хэрэгслээр хэлбэршүүлдэг. Нэмэлт мэдээллийг pr(1) гарын авлагаас харна уу.

**`-T гарчиг`**

pr(1) хуудасны толгой хэсэгт файлын нэрийг байрлуулахын оронд зааж өгсөн *Гарчиг*-г хэвлэдэг. Энэ сонголт нь гэхдээ зөвхөн `-p` сонголт хэрэглэсэн үед л хэрэгждэг.

**`-t`**

troff өгөгдлийг хэвлэнэ.

**`-v`**

raster өгөгдлийг хэвлэнэ.

Жишээ: Дараах тушаал нь ls(1)-н гарын авлагыг аятайхан загварчлаад анхдагч хэвлэгчээр хэвлэнэ:

```
% zcat /usr/shared/man/man1/ls.1.gz | troff -t -man | lpr -t
```

zcat(1) тушаал нь ls(1) гарын авлагын эх шахсан файлыг нь задлаад GNU troff хэлбэрт оруулдаг troff(1) тушаал уруу дамжуулна. Уг хэлбэрт оруулсны дараа lpr(1) руу дамжигдах бөгөөд энэ нь хэвлэх ажлыг LPD түр хадгалагч уруу илгээдэг. Бид `-t` сонголт lpr(1) дээр хэрэглэсэн болохоор түр хадгалагч хэвлэгдэх үед GNU troff гаралтыг хэвлэгчийн ойлгох хэлбэрт хувиргаж хэвлэнэ.

#### 10.5.4.2. Хэвлэлтэд хандах сонголтууд

lpr(1) тушаалд өгсөн дараах сонголтууд LPD дээрх хэвлэгдэх ажлуудад онцгойлж ханддаг:

***-# хуулбарлах-тоо***

Зөвхөн нэг удаа хэвлэгдэх ажлуудыг *хуулбарлах-тоо*гоор хувилдаг. Систем хянаж зохицуулагч албатай хүн магадгүй хэвлэгчийн ингэж ахин дахин хуулбарлаж хэвлэхийн оронд хувилах фото хуулбар үүсгээд түүнийгээ шууд буулгах сонголт санал тандтавьж болох юм. Олон хуулбарлалтыг хязгаарлах хэсэгт тодорхой заасан байгаа.

Доорх жишээн дээр parser.c файлыг гурав, parser.h файлыг мөн гурав хувилж анхдагч хэвлэгч рүү хэвлэж байна:

```
% lpr -#3 parser.c parser.h
```

**-m**

Хэвлэх ажил дууссаны дараа захиа илгээнэ. Хэвлэгдэх ажил дууссаны дараа LPD систем таны бүртгэл рүү захиа явуулдаг. Захиан дотроо таны хэвлэх ажиллагаа амжилттай болсон эсвэл алдаа гарсан тухай мэдэгддэг бөгөөд алдаа гарсан байвал (голдуу) ямар алдаа гарсныг нь бичсэн байдаг.

**-s**

Түр хадгалагч сан уруу хуулахгүйгээр уг файлуудад тэмдэгт холбоос үүсгэдэг.

Хэрэв та том хэмжээний файл хэвлэх гэж байгаа бол энэ сонголт танд тун хэрэг болно. Ингэснээр түр хадгалагчид буй дискний хэмжээг хэмнэдэг ( таны том хэмжээний файл түр хадгалагчийн хэмжээнээс хэтэрч түүнтэй зэрэгцээ орших зай уруу илүүдэж гарах аюултай). LPD мөн хэвлэх файлыг хуулах гэж цаг зарцуулахгүй түргэн байх болно.

Нэг сул тал бий: LPD нь хэвлэгдэх файл уруу шууд хандаж байгаа болохоор хэвлэгдэж дуусахаас нааш та уг файлыг засварлаж чадахгүй байх болно.

Хэрэв та сүлжээгээр өөр хэвлэгч рүү хэвлэж байгаа бол LPD нь уг компьютераас алсад буй хэвлэгч рүү файлыг хуулдаг болохоор `-s` сонголт нь дотоод зайгаа хэмнэхээс биш алсад буй компьютерийн дискний зайг хэмнэдэггүй. Гэвч ийм байсан ч хэрэгтэй сонголт билээ.

**-r**

Түр хадгалагч уруу хуулсны дараа юм уу эсвэл `-s` сонголт хэрэглэж хэвлэсний дараа уг файлыг утсгана. Энэ сонголтыг тун болгоомжтой хэрэглэнэ үү!

#### 10.5.4.3. Хуудасны толгой хэсгийг загварчлах сонголтууд

lpr(1) тушаалд хэрэглэсэн эдгээр сонголтууд нь хуудасны толгой хэсэгт хэвлэгддэг текстийг зааж өгөхөд хэрэглэгддэг. Хэрэв хуудасны толгой хэсгийг өөр програм өөрчилнө гэж заасан байвал энэ сонголт хүчингүй болдог. Хуудасны толгой хэсэг хэсэгт энэ талаар тодорхой өгүүлсэн байгаа.

**-C *текст***

Толгой хэсэгт байдаг компьютерийн нэрийг зааж өгсөн *текст*ээр орлуулдаг. Ердийн үед бол уг хэвлэх ажлыг илгээсэн компьютерийн нэр байдаг.

**-J *текст***

Толгой хэсэгт байдаг хэвлэх ажлын нэрийг зааж өгсөн *текст*ээр орлуулдаг. Ердийн үед ажлын нэр нь хэвлэх файлын нэр байдаг ба хэрэв та консолийн стандарт оруулга хэрэглэсэн бол stdin гэж байдаг.

**-h**

Толгой хэсэггүй хэвлэнэ.

Хэрэв өөр програмаар толгой хэсэг нь бэлтгэгдсэн бол энэ сонголт зарим тохиолдолд хүчингүй болдог. Хуудасны толгой хэсэг хуудсанд энэ тухай тайлбарласан буй

### 10.5.5. Хэвлэгчдийг зохицуулах

Хэвлэгчдийг зохицуулагчийн зүгээс тэдгээрийг суулгаж шалгахад үүрэгтэй байдаг. lpc(8) тушаал ашиглан хэвлэгчтэйгээ та олон аргаар харилцаж чадна. lpc(8) тушаал ашиглан

- Хэвлэгчийг эхлүүлэх болон зогсоох
- Хэвлэгдэх дарааллыг хорих болон зөвшөөрөх
- Дараалалд буй хэвлэгдэх ажлын ээлж дарааг өөрчлөх

Эхлээд жаахан тодруулцгаая: Хэрэв хэвлэгч *зогссон* бол дараалалд буй ямар ч ажлыг хэвлэхгүй. Хэрэглэгчид хэвлэх ажил илгээж болох бөгөөд тийм ажлууд бүгд дараалалд нэмэгдэж хэвлэгчийг *эхлэх* хүртэл юм уу эсвэл дарааллыг цэвэрлэх хүртэл тэнд хадгалагдана.

Хэрэв дараалал *хоригдсон* бол ямар ч хэрэглэгч (`root` хэрэглэгчээс бусад) хэвлэгч рүү юм илгээж чадахгүй. Дараалал *зөвшөөрөгдсөн* үед л хэвлэх ажил хүлээж авдаг. Хэвлэгч *эхлэх* үедээ мөн хоригдсон дарааллыг хоосортол нь хэвлэнэ.

lpc(8) тушаалыг хэрэглэхийн тулд та `root` эрхтэй байх хэрэгтэй байдаг. Энгийн хэрэглэгчид бол lpc(8) тушаалыг хэвлэгчийн төлөв байдал юм уу эсвэл гацсан хэвлэгчийг эхлүүлэхэд хэрэглэнэ.

Энд lpc(8) тушаалын нэгтгэсэн жагсаалтыг харуулав. Ихэнх тушаалуудад онцгойлон хэвлэгч рүү хандахын тулд *хэвлэгчийн-нэр* нэмж өгдөг. Хэрэв та `all` гэж *хэвлэгчийн-нэр*-н оронд өгвөл /etc/printcap файл жагсаалттай буй бүх хэвлэгчийг илэрхийлэх болно.

`abort хэвлэгчийн-нэр`

Одоогийн хэвлэж байгаа ажлыг цуцлаад хэвлэгчийг зогсооно. Хэрэв дараалал хоригдоогүй бол хэрэглэгчид хэвлэх зүйлс илгээж байж болно.

`clean хэвлэгчийн-нэр`

Хэвлэгчийн түр хадгалагчид буй хуучин файлуудыг устгана. LPD уг файлыг ямар нэгэн шалтгааны улмаас устгаагүй байж болох юм. Энэ нь хэвлэх үед алдаа гарах, эсвэл зохион байгуулах зарим үйлдэл хийгдэх гэх мэт янз бүрийн шалтгаан байж болно. Энэ тушаал түр хадгалагч санд хамаарахгүй файлуудыг шалгаж олоод устгадаг.

`disable хэвлэгчийн-нэр`

Шинээр ирэх ажлуудад дарааллыг хорьно. Хэрэв хэвлэгч тухайн үед хэвлэж байвал дараалалд үлдсэн байгаа зүйлсүүдийг хэвлэдэг. Супер хэрэглэгч (`root`) дараалал хоригдсон байсан ч хэвлэх зүйл илгээдэг.

Энэ тушаал шинэ хэвлэгчийн суулгац юм уу шүүлтийг шалгахад их хэрэгтэй. Жишээ нь хэвлэх дарааллыг хориод `root` эрхээр хэвлэж туршиж болно. Бусад хэрэглэгчид дарааллыг `идэвхжих` хүртэл дараалал уруу ажил илгээж чадахгүй.

`down хэвлэгчийн-нэр мэдээлэл`

Хэвлэгчийг унтраана. `disable` тушаалыг `stop`-тай хамт хэрэглэсэнтэй ижил. Хэрэв хэрэглэгч хэвлэгчийн lpq(1) юм уу эсвэл `lpc status` тушаал хэрэглэн төлөв байдлыг мэдье гэсэн үед уг зааж өгсөн *мэдээлэл* харуулагддаг.

`enable хэвлэгчийн-нэр`

Хэвлэгчийн дарааллыг зөвшөөрдөг. Хэрэглэгч хэвлэгч рүү юм илгээж болох боловч хэвлэгч эхлэхээс нааш хэвлэгдэхгүй.

`help тушаал`

*тушаал*ын хэрэглэх тусламжийг харуулдаг. Хэрэв *тушаал* бичилгүйгээр тушаавал боломжтой бүх тушаалын ерөнхий мэдээллийг харуулна.

`restart хэвлэгчийн-нэр`

Хэвлэгчийг эхлүүлдэг. Ердийн хэрэглэгчид энэ тушаалыг хэрэв ямар нэгэн тохиолдлоор LPD гацсан үед хэвлэгчийг эхлүүлэхэд хэрэглэж болдог боловч `stop` юм уу `down` тушаал хэрэглэн ажиллаж байгаа хэвлэгчийг зогсоож чаддаггүй. `restart` тушаал нь `abort` тушаалыг `start` тушаалтай хэрэглэсэнтэй ижилхэн.

`start хэвлэгчийн-нэр`

Хэвлэгчийг эхлүүлнэ. Хэвлэгч дараалалд байгаа ажлуудыг хэвлэж эхэлдэг.

`stop хэвлэгчийн-нэр`

Хэвлэгчийг зогсооно. Хэвлэгч яг хэвлэж байсан ажлаа дуусгаад дараалалд байгаа ажлуудыг хэвлэлгүйгээр зогсоно. Хэвлэгч зогссон байсан ч хэрэглэгчид хэвлүүлэх ажлаа дараалалд илгээж болно.

`topq хэвлэгчийн-нэр ажил-эсвэл-хэрэглэгчийн-нэр`

*хэвлэгчийн-нэр* нэртэй хэвлэгчийн дараалалд байгаа ажлуудын хамгийн дээд хэсэгт нь зааж өгсөн *хэрэглэгчийн-нэр* хэрэглэгчийн *ажлыг* оруулдаг. Энэ тушаалыг `all` гэж *хэвлэгчийн-нэр*-н оронд хэрэглэх боломжгүй.

`up хэвлэгчийн-нэр`

`down` тушаалын эсрэгээр уг хэвлэгчийг шинээр эхлүүлдэг. `start` тушаалыг `enable`-тай хэрэглэсэнтэй ижилхэн.

lpc(8) нь дээрх тушаалуудыг тушаал бичих мөрнөөс хүлээж авдаг. Хэрэв та ямар ч тушаал оруулаагүй бол lpc(8) нь `exit` юм уу `quit`, эсвэл end-of-file тэмдэгт илгээх хүртэл гарнаас тушаал оруулж ажиллах харилцах горимд шилждэг.

## 10.6. Хэвлэгчийн стандарт түр хадгалагчидтай ижил хадгалагчид

Хэрэв та энэ гарын авлагыг эхнээс нь уншиж байгаа бол FreeBSD-тэй цуг ирдэг түр хадгалах систем болох LPD-ийн талаар бүх л мэдэж болох зүйлсийг одоо сурсан байх ёстой. Та үүний олонхи дутагдалтай талуудыг үнэлж чадах байх. Энэ нь дараах асуултанд хүргэдэг: "Өөр ямар (FreeBSD-тэй ажилладаг) түр хадгалах системүүд байдаг вэ?"

**LPRng**

LPRng буюу "LPR: the Next Generation" гэгддэг дараа үеийн систем нь PLP-ийг дахин бичсэн хувилбар юм. Патрик Пауэл болон Жастин Мэйсон (PLP-ийн гол арчлагч) нар нийлэн LPRng-г хийсэн юм. LPRng-ийн гол сайт бол http://www.lprng.org/ юм.

**CUPS**

CUPS буюу Common UNIX Printing System нь UNIX® дээр тулгуурласан үйлдлийн системүүдэд зориулсан зөөгдөж болох хэвлэх давхаргын боломжийг олгодог. Энэ нь бүх

UNIX® үйлдвэрлэгчид болон хэрэглэгчдэд зориулж стандарт хэвлэх шийдлийг дэмжихийн тулд Easy Software Products-аас хөгжүүлэгдсэн юм.

CUPS нь Internet Printing Protocol (IPP) буюу Интернэтийн Хэвлэх Протоколыг ашиглаж хэвлэх ажлууд болон дарааллуудыг удирдах үндсээ болгодог. Line Printer Daemon (LPD) буюу Шугамын Хэвлэгчийн Дэмон, Server Message Block (SMB) буюу Серверийн Мэдэгдлийн Блок, болон AppSocket (JetDirect гэгддэг) протоколууд нь багасгасан ажиллагаатайгаар бас дэмжигдсэн байдаг. CUPS нь амьдралд байх UNIX® дээрх хэвлэлтийг дэмжихийн тулд сүлжээний хэвлэгч олох (browsing) болон PostScript Printer Description (PPD) буюу PostScript Хэвлэгчийн Тайлбар дээр суурилсан хэвлэх тохируулгуудыг нэмдэг.

CUPS-ийн гол сайт нь http://www.cups.org/ юм.

**HPLIP**

HPLIP буюу HP Linux® Imaging and Printing систем нь HP өөрийнхөө төхөөрөмжүүдэд зориулан хөгжүүлсэн хэвлэлт, скан хийлт, болон факсын боломжуудыг дэмжсэн програм хангамжийн цуглуулга юм. Энэ програм хангамжийн цуглуулга нь өөрийн зарим хэвлэх боломжууддаа зориулж ар талдаа CUPS хэвлэх системийг хэрэглэдэг.

HPLIP-ийн гол сайт нь http://hplipopensource.com/hplip-web/index.html юм.

# 10.7. Алдааг олж засварлах

lptest(1) програмын тусламжтайгаар энгийн шалгах үйлдэл хийж дуусахад та зөв хэвлэгдсэн хүссэн үр дүнд биш харин дараах алдаануудтай тулгарч болзошгүй юм:

**Хэвлэж болж байна, гэхдээ хэсэг хугацааны дараа юм уу эсвэл анхнаасаа хуудасны тал хэсгийг хэвлээд зогсох.**

Хэвлэгч дээд хэсгийг нь хэвлээд хэсэг хугацааны дараа юу ч хийхгүй болох. Ийм үед та хэвлэгч дээрх PRINT REMAINING эсвэл FORM FEED товч дарж үлдсэн хэсгийг нь хэвлэх эсэхийг харах хэрэгтэй.

Хэрэв хэвлэгч өөр хэвлэх юм байгаа эсэхийг хүлээгээд байгаа тийм нөхцөлд орвол хэвлэгч рүү FORM FEED тэмдгийг илгээснээр (эсвэл ямар нэгэн юм) уг асуудал шийдэгдэж болох юм. Өөрийнхөө түр хадгалагч дээр орж ирсэн өгөгдлийг шууд хэвлэдэг хэвлэгчдэд иймэрхүү арга хэрэглэхэд хангалттай билээ. Хэрэв өмнөх хуудас нь хуудасны тал хүртэл байгаад дараагийн хуудас нь шинээр хуудаснаас эхлэх байсан болоод дууссан хуудасны тал хэсгээс хэвлэхгүй байгаад байх нөхцөл үүссэн байж болзошгүйг шалгахад мөн илүүдэхгүй.

/usr/local/libexec/if-simple бүрхүүлийн скриптэд оруулсан дараах оруулга нь хэвлэх үйлдлийн дараа хэвлэгч рүү хуудас дууссан тэмдэгт илгээнэ:

```
#!/bin/sh
#
# if-simple - Simple text input filter for lpd
# Installed in /usr/local/libexec/if-simple
```

```
#
# Simply copies stdin to stdout.  Ignores all filter arguments.
# Writes a form feed character (\f) after printing job.

/bin/cat && printf "\f" && exit 0
exit 2
```

**Гэхдээ энэ нь "хачирхалтай нөлөө" үүсгэж болох юм.**

Та дараах зүйлсийг хуудсан дээрээ харах болно:

```
!"#$%&'()*+,-./01234
               "#$%&'()*+,-./012345
                              #$%&'()*+,-./0123456
```

Энэ нь таныг шинэ мөр илэрхийлдэг тэмдэгт хөрвүүлэгдээгүй улмаас өнөөх *хачирхалтай нөлөөний* хохирогч нь болсон байна гэдгийг илэрхийлж байна. UNIX® төрлийн үйлдлийн системүүд мөр дууссаныг илэрхийлэхийн тулд ASCII-н 10 гэсэн дугаарыг буюу line feed (LF) гэсэн нэг л тэмдэгт хэрэглэдэг. Харин MS-DOS®, OS/2®, болон бусад үйлдлийн системүүд ASCII-н 10 *ба* ASCII-н 13 (CR буюу шинэ мөр эхлэх) гэсэн хоёр тэмдгийг хамтад нь хэрэглэдэг. Ихэнх хэвлэгч нар шинэ мөрийг илэрхийлэхдээ MS-DOS®-н шийдлийг хэрэглэдэг.

Хэрэв та FreeBSD-с хэвлэхээр бол таны текстүүд зөвхөн мөр дууссан тэмдэгт л агуулсан байгаа. Мөр дууссан тэмдэгтийг хэвлэгч хүлээж авангуутаа хуудсаа нэг мөрөөр дээшлүүлсэн мөртлөө яг тухайн байрлалдаа дараагийн тэмдэгт ирэхийг хүлээж зогсдог. Энэ үед л шинэ мөр гэсэн тэмдэгт ирж байж хэвлэгчийн толгой зүүн хэсэг рүүгээ гүйдэг билээ.

FreeBSD нь дараах маягаар хэвлэхийг хэвлэгчээс хүсдэг:

| | |
|---|---|
| Хэвлэгч CR тэмдэг хүлээж авлаа | Хэвлэгч CR гэж хэвлэв |
| Хэвлэгч LF тэмдэгт хүлээж авлаа | Хэвлэгч CR + LF гэж хэвлэх |

Үүнийг залруулах хэдэн арга бий:

- Хэвлэгчийнхээ тохируулгын цонхыг ашиглан эдгээр тэмдэгтийг өөрөөр хөрвүүлж ойлгохоор тохируулах. Иймэрхүү тохиргоо байдаг эсэхийг хэвлэгчийнхээ гарын авлагад харах хэрэгтэй.

Хэрэв та системээ шинээр ачаалахдаа FreeBSD-ээс өөр давхар суусан үйлдлийн систем рүү орвол тухайн үйлдлийн системдээ тохируулж CR болон LF тэмдэгтийн хөрвүүлэх тохиргоог *дахин хийх* хэрэг гарч магадгүй юм.

- FreeBSD-н цуваа холболтын таниулагч програм нь автоматаар LF тэмдэгтийг CR+LF тэмдэгт рүү хувиргадаг. Мэдээж энэ нь *зөвхөн* цуваа холболтоор холбогдсон хэвлэгч дээр ажиллана. Энэ шинж чанарыг идэвхжүүлэхийн тулд `ms#` хэрэгслийг ашиглан

`onlcr` горимыг хэвлэгчийн /etc/printcap файл дотор оруулж өгөх хэрэгтэй.

- Хэвлэгчийн алгасаж болдог *escape code* тэмдэгт илгээж хэвлэгчийг LF тэмдэгт илгээгдсэн гэж ойлгуулж болох юм. Хэвлэгчийн гарын авлагаас ийм тэмдэгт дэмждэг эсэхийг харах хэрэгтэй. Хэрэв та тохирох тийм тэмдэгт олсон бол текст шүүгч програмыг өөрчилж эхлээд тэр тэмдэгтийг илгээгээд дараа нь хэвлэх зүйлсийг илгээхээр засах хэрэгтэй.

  Энд Hewlett-Packard PCL хэвлэгчийн ойлгодог алгасаж болдог тэмдэгтийн текст шүүгч жишээг татаж үзүүлэв. Энэ шүүгч нь LF тэмдэгтийг LF ба CR гэж хөрвүүлээд хэвлэгч рүү хэвлэх ажлыг илгээж дараа нь мөр дууссан тэмдэгтийг илгээж хэвлэх ажил дууссаныг мэдэгддэг. Энэ нь бараг бүх Hewlett Packard хэвлэгчдэд тохирдог.

```
#!/bin/sh
#
# hpif - Simple text input filter for lpd for HP-PCL based printers
# Installed in /usr/local/libexec/hpif
#
# Simply copies stdin to stdout.  Ignores all filter arguments.
# Tells printer to treat LF as CR+LF.  Ejects the page when done.

printf "\033&k2G" && cat && printf "\033&l0H" && exit 0
exit 2
```

  Энд `orchid` нэртэй компьютерийн /etc/printcap файлын жишээг харуулж байна. Энд `teak` нэртэй Hewlett Packard LaserJet 3Si хэвлэгчийг зэрэгцээ холболтоор холбосон байгаа. Энд өмнө үзүүлсэн скриптийг текст шүүгчээ болгож хэрэглэсэн байгаа:

```
#
#  /etc/printcap for host orchid
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0:sh:sd=/var/spool/lpd/teak:mx#0:\
        :if=/usr/local/libexec/hpif:
```

**Энэ нь мөрүүдийг давхарлан хэвлэж байх.**

Хэвлэгч мөрөө дээшлүүлсэнгүй. Бүх мөрүүд дээд мөрөнд давхарлагдаж хэвлэгдэв.

Энэ хүндрэл нь өнөөх хачирхалтай нөлөөний "эсрэг" нөлөөлөл бөгөөд арай бага тохиолддог. FreeBSD-н мөрийн төгсгөл илэрхийлсэн LF тэмдэгт нь CR тэмдэгтээр солигдоод мөрөө шинээр ахиулалгүй хуудасны эхэнд очиж дахин хэвлэчихсэн байна.

Хэвлэгчийн тохируулгын цонх ашиглан LF болон CR тэмдэгтүүдийг дараах маягаар хөрвүүлэхээр тохируулах хэрэгтэй:

| **Хэвлэгчийн хүлээж авалт** | **Хэвлэгчийн хэвлэлт** |
|---|---|
| CR | CR |

| Хэвлэгчийн хүлээж авалт | Хэвлэгчийн хэвлэлт |
|---|---|
| LF | CR + LF |

**Хэвлэгч тэмдэгт гээх үзэгдэл.**

Хэвлэгч хэвлэх үедээ мөрөнд зарим тэмдэгтүүдийг хэвлэхгүй болдог. Энэ хүндрэл нь хэвлэгч ажиллахгүй болох эсвэл маш их тэмдэгт гээж эхлэх зэргээс илүү дор үзэгдэл юм.

Хүндрэлийн шалтгаан нь цуваа холболтоор компьютерийн илгээсэн хурдтай хэвлэгчийн хэвлэх хурд таарахгүй байгаагаас үүсдэг (ийм хүндрэл зэрэгцээ холбосон хэвлэгчдэд тулгардаггүй). Энэ хүндрэлийг давах хоёр арга бий:

- Хэрэв хэвлэгч нь XON/XOFF гэсэн урсгал зохицуулагчийг дэмждэг бол `ms#` шинж чанарыг ашиглан `ixon` горимыг FreeBSD дээр хэрэглэхээр тааруулах хэрэгтэй.
- Хэрэв хэвлэгч нь Request to Send (Илгээхийг Хүсэх) / Clear to Send (Илгээхдээ цэвэрлэх) hardware handshake буюу тоног төхөөрөмжийн тохиролцоог (ихэвчлэн `RTS/CTS` гэгддэг) дэмждэг бол `ms#` боломжид `crtscts` горимыг зааж өгөх хэрэгтэй. Тоног төхөөрөмжийн урсгал зохицуулагчид тааруулж цуваа холболтоо зөв холбосон эсэхээ магадлах хэрэгтэй.

**Хог хэвлэх үзэгдэл.**

Хэвлэгч хүссэн текстийг хэвлэхийн оронд хог хэвлээд эхлэв.

Энэ нь цуваа холболт буруу холбогдсоноос үүсдэг үзэгдэл. Дамжуулах хурдны давтамжийн `br` шинж чанарыг шалгаад `ms#` шинж чанараар өгөгдөл тэгшилж илгээнэ гэх зэрэг /etc/printcap файлд байгаа тохиргоотой тохирч буй эсэхийг шалгах хэрэгтэй.

**Юу ч болохгүй байх.**

Хэрэв юу ч болохгүй байвал хүндрэл нь хэвлэгчид биш FreeBSD-д байж болох юм. /etc/printcap файлд бүртгэл файл хэрэглэнэ гэсэн (`lf`) шинж чанарыг нэмэх хэрэгтэй. Жишээ нь энд `rattan` оруулганд `lf` шинж чанар нэмж байна:

```
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan:\
        :lp=/dev/lpt0:\
        :if=/usr/local/libexec/if-simple:\
        :lf=/var/log/rattan.log
```

Тэгээд дахиад хэвлэх гээд оролдоод үз. Гарч болзошгүй алдааны мэдэгдэл байгаа эсэхийг бүртгэлийн файлаас (бидний жишээн дээр /var/log/rattan.log) шалгах хэрэгтэй. Мэдэгдлүүдээс хамаарч асуудлыг засахыг оролдох хэрэгтэй.

Хэрэв та `lf` шинж чанар зааж өгөөгүй бол, LPD нь /dev/console файлыг анхдагч гэж авч үзнэ.

# Chapter 11. Линуксийн Хоёртын Нийлэмж

## 11.1. Ерөнхий агуулга

FreeBSD нь Линуксийг оруулаад хэд хэдэн бусад UNIX® төст үйлдлийн системүүдтэй хоёртын нийлэмжтэй байдаг. Та магадгүй FreeBSD яагаад Линукс хоёртын файлуудыг ажиллуулж чаддаг байх хэрэгтэй вэ? гэж өөрөөсөө асууж болох юм. Энэ асуултын хариулт их энгийн. Линукс нь тооцоолох ертөнц дэх хамгийн сүүлийн үеийн "халуун зүйл" болохоор олон компаниуд болон хөгжүүлэгчид зөвхөн Линуксд зориулж хөгжүүлдэг. Тэгэхээр бидний FreeBSD хэрэглэгчид эдгээр компаниуд болон хөгжүүлэгчдээс өөрсдийн програмууддаа зориулж FreeBSD-ийн хувилбарууд бичихийг шалгаахаас өөр аргагүйд хүргэдэг. Гол асуудал нь эдгээр компаниудын ихэнх нь өөрсдийнх нь бүтээгдэхүүн бас FreeBSD дээрх хувилбартай байсан бол хэр олон хүмүүс ашиглахыг яг үнэндээ бодож байгаагүй бөгөөд ихэнх нь зөвхөн Линуксд зориулж хөгжүүлэлт хийсээр байдаг. Тэгэхээр FreeBSD хэрэглэгчид юу хийх вэ? Яг энэ үед Линуксийн хоёртын нийлэмж хэрэг болдог билээ.

Товчхондоо нийлэмж нь FreeBSD хэрэглэгчдэд Линуксийн бүх програмуудын ойролцоогоор 90%-ийг засваргүйгээр ажиллуулах боломжийг олгодог. Эдгээрт StarOffice™, getenv(3)-ийн Линуксийн хувилбар, Adobe® Acrobat®, RealPlayer®, Oracle®, WordPerfect®, Doom, Quake зэрэг олон програмууд орно. Зарим тохиолдолд Линуксийн хоёртын файлууд нь Линукс дээр байгаагаас илүү хурдан FreeBSD дээр ажилладаг тохиолдол гарсан байна.

Гэхдээ зарим нэг Линуксд зөвхөн зориулагдсан үйлдлийн системийн боломжууд FreeBSD дээр дэмжигдээгүй байдаг. Хэрэв Линуксийн хоёртын файлууд нь виртуал 8086 горимыг идэвхжүүлэх зэрэг зөвхөн i386™-д зориулсан дуудлагуудыг ихээр ашиглаж байгаа үед ажилладаггүй.

Энэ бүлгийг уншиж дууссаны дараа та дараах зүйлсийг мэдэх болно:

- Өөрийн систем дээр Линуксийн хоёртын нийлэмжийг хэрхэн идэвхжүүлэх талаар.
- Линуксийн нэмэлт хуваалцсан сангуудыг хэрхэн суулгах талаар.
- Өөрийн FreeBSD систем дээр Линуксийн програмуудыг хэрхэн суулгах талаар.
- FreeBSD дээрх Линуксийн нийлэмжийн шийдлийн нарийн учруудын талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- Гуравдагчдын бүтээсэн програм хангамжуудыг хэрхэн суулгах талаар мэдэх (Програм суулгах. Багцууд болон портууд).

## 11.2. Суулгалт

Линуксийн хоёртын нийлэмж анхдагчаар идэвхтэй байдаггүй. Энэ ажиллагааг хамгийн хялбараар идэвхжүүлэхийн тулд `линукс` KLD обьектийг ("Kernel LoaDable обьект") дуудах явдал юм. Та энэ модулийг дараах тушаалыг `root` эрхээр ажиллуулж дуудаж болно:

```
# kldload linux
```

Хэрэв та Линуксийн нийлэмжийг үргэлж идэвхжүүлэхийг хүсэж байгаа бол дараах мөрийг /etc/rc.conf файлд нэмэх хэрэгтэй:

```
linux_enable="YES"
```

KLD дуудагдсан эсэхийг шалгахын тулд kldstat(8) тушаалыг ашиглаж болно:

```
% kldstat
Id Refs Address    Size     Name
 1    2 0xc0100000 16bdb8   kernel
 7    1 0xc24db000 d000     linux.ko
```

Хэрэв ямар нэгэн шалтгаанаас болоод та KLD-г хүсэхгүй эсвэл дуудаж чадахгүй байгаа бол `options COMPAT_LINUX` тохируулгыг өөрийн цөмийн тохиргооны файлдаа нэмж Линуксийн хоёртын нийлэмжийг статикаар холбож болох юм. Дараа нь FreeBSD цөмийг тохируулах нь хэсэгт тайлбарласны дагуу өөрийн шинэ цөмийг суулгах хэрэгтэй.

## 11.2.1. Линуксийн ажиллах үеийн сангуудыг суулгах нь

linux_base порт юм уу эсвэл тэдгээрийг гараар суулгах гэсэн хоёр аргын аль нэгээр үүнийг хийж болох юм.

### 11.2.1.1. linux_base портыг суулгаж ашиглах

Ажиллах үеийн сангуудыг суулгахад ашиглах хамгийн хялбар арга нь энэ юм. Энэ нь ямар нэгэн портыг Портын цуглуулгаас суулгаж байгаатай адил юм. Ердөө л доор дурдсаныг хийх хэрэгтэй:

```
# cd /usr/ports/emulators/linux_base-fc10
# make install distclean
```

FreeBSD 8.0-с өмнөх FreeBSD системүүд дээр та emulators/linux_base-f10-ийн оронд emulators/linux_base-fc4 портыг ашиглах ёстой.

Та одоо ажилладаг Линуксийн хоёртын нийлэмжтэй болсон байх ёстой. Зарим програмууд системийн сангуудын буруу бага хувилбаруудын талаар гомдоллож болох юм. Гэхдээ энэ нь ерөнхийдөө асуудал биш байдаг.

Төрөл бүрийн Линукс тархацуудын янз бүрийн хувилбаруудад тохирох emulators/linux_base портын олон хувилбарууд байж болох юм. Таны суулгахыг хүсэж байгаа Линуксийн програмуудын шаардлагыг аль болох ойр хангаж байгаа тийм портыг та суулгах шаардлагатай.

#### 11.2.1.2. Сангуудыг гараар суулгах

Хэрэв та "портын" цуглуулга суулгаагүй бол сангуудыг гараар суулгаж болно. Програмын хамаардаг Линуксийн хуваалцсан сангууд болон ажиллах үеийн холбогч (linker) танд хэрэгтэй болно. Та бас "сүүдэр root" сан буюу /compat/linux санг өөрийн FreeBSD систем дээр Линуксийн сангуудад зориулж үүсгэх хэрэгтэй болно. FreeBSD дээр ажиллаж байгаа Линуксийн програмуудын онгойлгосон дурын хуваалцсан сангууд энэ модыг эхлээд хайдаг. Тэгэхээр хэрэв Линуксийн програм жишээ нь /lib/libc.so-г дуудвал FreeBSD эхлээд /compat/linux/lib/libc.so-г онгойлгохыг оролдох бөгөөд хэрэв байхгүй бол /lib/libc.so-г оролдох болно. Хуваалцсан сангууд нь Линуксийн `ld.so` тушаалаар гарч байгаа замуудад биш харин сүүдэр мод /compat/linux/lib-т суулгагдах ёстой.

Ерөнхийдөө таныг Линуксийн програм өөрийн FreeBSD системд суулгах эхний цөөхөн хэдэн удаад зөвхөн Линуксийн хоёртын файлуудын хамаардаг хуваалцсан сангуудыг хайх хэрэгтэй болно. Хэсэг хугацааны дараа та өөрийн систем дээрээ ямар нэгэн илүү ажиллагаагүйгээр шинээр импорт хийгдсэн Линуксийн хоёртын файлуудыг ажиллуулж чадах Линуксийн хуваалцсан сангуудын хангалттай олонлогтой болсон байх болно.

#### 11.2.1.3. Нэмэлт хуваалцсан сангуудыг хэрхэн суулгах вэ

linux_base портыг суулгасан ч гэсэн таны програм хуваалцсан сангууд байхгүй байна гээд гомдоллоод байвал яах вэ? Линуксийн хоёртын файлуудад ямар хуваалцсан сангууд хэрэгтэйг яаж мэдэх вэ, тэдгээрийг хаанаас авах вэ? Үндсэндээ 2 боломж байдаг (эдгээр зааврууд дагаж байхдаа та өөрийн FreeBSD систем дээрээ `root` байх хэрэгтэй болно).

Линукс системд хандах боломжтой бол програмд ямар хуваалцсан сангууд хэрэгтэйг мэдээд тэдгээрийг өөрийн FreeBSD систем уруу хуулах хэрэгтэй. Доорх жишээг харна уу:

Та FTP ашиглан Doom гэсэн Линуксийн хоёртын файлыг авч өөрөө хандаж болох Линукс систем дээрээ байрлуулсан гэж үзье. Та тэгээд үүний шаарддаг хуваалцсан сангуудыг `ldd linuxdoom` тушаал ашиглан иймэрхүүгээр шалгаж болно:

```
% ldd linuxdoom
libXt.so.3 (DLL Jump 3.1) => /usr/X11/lib/libXt.so.3.1.0
libX11.so.3 (DLL Jump 3.1) => /usr/X11/lib/libX11.so.3.1.0
libc.so.4 (DLL Jump 4.5pl26) => /lib/libc.so.4.6.29
```

Та сүүлийн баганад байгаа бүх файлуудыг авч эхний баганад байгаа нэрсийг тэдгээр уруу заасан симболын холбоосууд болгон /compat/linux дотор байрлуулах хэрэгтэй. Энэ нь та өөрийн FreeBSD системдээ эцсийн эцэст эдгээр файлуудтай болно гэсэн үг юм:

```
/compat/linux/usr/X11/lib/libXt.so.3.1.0
/compat/linux/usr/X11/lib/libXt.so.3 -> libXt.so.3.1.0
/compat/linux/usr/X11/lib/libX11.so.3.1.0
/compat/linux/usr/X11/lib/libX11.so.3 -> libX11.so.3.1.0
/compat/linux/lib/libc.so.4.6.29
/compat/linux/lib/libc.so.4 -> libc.so.4.6.29
```

Хэрэв та `ldd` тушаалын гаралтын эхний баганатай Линуксийн хуваалцсан сангийн гол залруулалтын дугаар таарсан тийм Линуксийн хуваалцсан сантай байгаа бол сүүлийн баганад байгаа файлуудыг өөрийн систем уруу хуулах шаардлагагүй бөгөөд танд байгаа чинь ажиллах ёстойг санаарай. Хэрэв шинэ хувилбар бол хуваалцсан санг ямар ч байсан хуулахыг зөвлөдөг. Та шинэ уруу заасан симболын холбоос үүсгэснийхээ дараа хуучинг устгаж болно. Тэгэхээр хэрэв эдгээр сангууд таны өөрийн системд байгаа:

```
/compat/linux/lib/libc.so.4.6.27
/compat/linux/lib/libc.so.4 -> libc.so.4.6.27
```

бөгөөд шинэ сан нь `ldd` тушаалын гаралтын дагуу сүүлийн хувилбарыг шаардана гэж байгааг мэдвэл:

```
libc.so.4 (DLL Jump 4.5pl26) -> libc.so.4.6.29
```

Хэрэв төгсөж байгаа тоон дээр зөвхөн нэг юм уу эсвэл хоёр хувилбар хуучин байгаа бол /lib/libc.so.4.6.29-г бас хуулж санаагаа зовоогоод хэрэггүй бөгөөд програм нь нэлээн хуучин хувилбартай зүгээр ажиллах ёстой юм. Гэхдээ хэрэв та хүсэж байгаа бол libc.so-г ямар ч байсан гэсэн солихоор шийдэж болох бөгөөд энэ нь таныг доорхтой үлдээх болно:

```
/compat/linux/lib/libc.so.4.6.29
/compat/linux/lib/libc.so.4 -> libc.so.4.6.29
```

Симболын холбоосын арга нь *зөвхөн* Линукс хоёртын файлуудад хэрэгтэй байдаг. FreeBSD-ийн ажиллах үеийн холбогч нь таарах гол залруулалтын дугааруудыг хайх талаар өөрөө санаа тавьдаг болохоор та санаа зовох хэрэггүй юм.

### 11.2.2. Линуксийн ELF хоёртын файлуудыг суулгах

ELF хоёртын файлууд нь заримдаа "branding" буюу "тамгалах" нэмэлт алхмыг шаарддаг. Хэрэв та тамгалаагүй ELF хоёртын файлыг ажиллуулахыг оролдвол доор дурдсантай төстэй алдааг хүлээн авах болно:

```
% ./my-linux-elf-binary
ELF binary type not known
Abort
```

FreeBSD цөмд FreeBSD ELF хоёртын файлыг Линуксийн хоёртын файлаас ялгахад туслахын тулд brandelf(1) хэрэгслийг ашиглана.

```
% brandelf -t Linux my-linux-elf-binary
```

Өнөөдөр GNU хэрэгслийн цуглуулга нь ELF хоёртын файлуудад автоматаар тохирох тамгалалтын мэдээллийг байрлуулдаг. Тэгэхээр энэ алхам нь ирээдүйд бараг л хэрэггүй болох ёстой юм.

### 11.2.3. RPM дээр тулгуурласан линуксын програм суулгах

FreeBSD нь өөрийн багцын өгөгдлийн сантай бөгөөд бүх портыг (Linux®-ийн портуудыг бас) хянахад ашиглагддаг. Linux®-ийн RPM санг ашигладаггүй (дэмждэггүй).

Гэхдээ хэрэв та Linux®-ийн RPM програм суулгах шаардлагатай бол үүнийг доорх аргаар хийж болно:

```
# cd /compat/linux
# rpm2cpio -q < /path/to/linux.archive.rpm | cpio -id
```

Дараа нь ELF хоёртын файлуудыг (сангуудыг биш!) brandelf хийх хэрэгтэй. Та суулгаснаа буцаж цэвэрхэн арилгаж чадахгүй боловч энэ нь тест хийхэд чинь хэрэг болж болох юм.

### 11.2.4. Хостын нэр танигчийг тохируулах

Хэрэв DNS ажиллахгүй байвал эсвэл та ийм мэдэгдэл хүлээн авбал:

```
resolv+: "bind" is an invalid keyword resolv+:
"hosts" is an invalid keyword
```

Та дараах мөрүүдийг агуулсан /compat/linux/etc/host.conf файлыг тохируулах хэрэгтэй болно:

```
order hosts, bind
multi on
```

Энд байгаа order нь /etc/hosts-г эхлээд хайж дараа нь DNS-ийг хайна гэж зааж байна. /compat/linux/etc/host.conf суугаагүй үед Линуксийн програмууд FreeBSD-ийн /etc/host.conf-г олж нийцгүй FreeBSD-ийн бичлэг байна гэж гомдоллох болно. Хэрэв та /etc/resolv.conf файл ашиглан нэрийн серверийг тохируулаагүй бол `bind` гэсэн үгийг устгах хэрэгтэй.

## 11.3. Mathematica® суулгах нь

Энэ баримт нь Mathematica® 5.X-ийн Линукс хувилбарыг FreeBSD систем уруу суулгах процессийг тайлбарлах болно.

Mathematica® эсвэл Mathematica® for Students-ийн Линуксийн хувилбарыг http://www.wolfram.com/ дэх Wolfram-аас захиалж болно.

### 11.3.1. Mathematica® суулгагчийг ажиллуулах

Эхлээд Mathematica®-ийн Линукс хоёртын файлууд Линуксийн ABI-г ашиглана гэдгийг та FreeBSD-д хэлж өгөх хэрэгтэй. Ингэж хийх хамгийн хялбар арга бол бүх тамгалаагүй хоёртын файлуудын хувьд анхдагч ELF тамгалалтыг Линукс болгон дараах тушаалаар тохируулах явдал юм:

```
# sysctl kern.fallback_elf_brand=3
```

Энэ нь тамгалаагүй ELF хоёртын файлууд Линуксийн ABI-г ашиглана гэж бодоход FreeBSD-г хүргэх бөгөөд та суулгагчийг CDROM-оос шууд ажиллуулах боломжтой болох ёстой.

Одоо MathInstaller файлыг өөрийн хатуу хөтөч уруу хуулах хэрэгтэй:

```
# mount /cdrom
# cp /cdrom/Unix/Installers/Linux/MathInstaller /localdir/
```

бөгөөд энэ файл дахь эхний мөрний `/bin/sh`-г `/compat/linux/bin/sh` гэж солих хэрэгтэй. Энэ нь суулгагч sh(1)-ийн Линуксийн хувилбарыг ажиллуулахаар болгож байгаа юм. Дараа нь `Linux)` гэсэн бүгдийг `FreeBSD)` гэдгээр текст засварлагчаар юм уу эсвэл доорх дараагийн хэсэгт байгаа скриптээр солих хэрэгтэй. Үйлдлийн системийг тодорхойлохын тулд `uname -s` тушаалыг дууддаг Mathematica® суулгагчид FreeBSD-г Линукс төст үйлдлийн систем гэж үзэхийг энэ нь хэлж өгнө. `MathInstaller`-г ажиллуулахад одоо Mathematica®-г суулгах болно.

### 11.3.2. Mathematica®-ийн ажиллуулах боломжтой файлуудыг өөрчлөх

Суулгах явцад Mathematica®-ийн үүсгэсэн бүрхүүлийн скриптүүдийг ашиглахаасаа өмнө өөрчлөх хэрэгтэй. Хэрэв та Mathematica®-ийн ажиллуулах боломжтой файлуудыг байрлуулах санд /usr/local/bin-г сонгосон бол та энэ санд math, mathematica, Mathematica, болон MathKernel гэж нэрлэгдсэн файлууд уруу заагдсан симболын холбоосуудыг олох болно. Эдгээр бүрт `Linux)` гэдгийг `FreeBSD)` гэдгээр текст засварлагчаар юм уу эсвэл доорх бүрхүүлийн скриптээр солих хэрэгтэй:

```
#!/bin/sh
cd /usr/local/bin
for i in math mathematica Mathematica MathKernel
  do sed 's/Linux)/FreeBSD)/g' $i > $i.tmp
  sed 's/\/bin\/sh/\/compat\/linux\/bin\/sh/g' $i.tmp > $i
  rm $i.tmp
  chmod a+x $i
done
```

### 11.3.3. Өөрийн Mathematica® нууц үгийг олж авах

Mathematica®-г эхний удаа эхлүүлэхэд танаас нууц үг асуух болно. Хэрэв та Wolfram-аас нууц үгээ авч амжаагүй байгаа бол өөрийн "машины ID"-г авахын тулд суулгацын санд

mathinfo-г ажиллуулах хэрэгтэй. Энэ машины ID нь таны эхний Ethernet картны MAC хаяг дээр тулгуурласан болохоор Mathematica®-ийн өөрийн хуулбарыг та өөр машинууд дээр ажиллуулж чадахгүй юм.

Wolfram-д цахим захидлаар ч юм уу эсвэл утас, факсаар бүртгүүлэх үедээ та "машины ID"-аа тэдэнд өгөх бөгөөд тэд бүлэг тоонуудаас тогтох харгалзах нууц үг бүхий хариу өгөх болно.

### 11.3.4. Mathematica® нүүрэн хэсгийг сүлжээгээр ажиллуулах

Mathematica® нь тэмдэгтүүдийг үзүүлэхийн тулд аль ч стандарт үсгийн маягуудын олонлогуудад байдаггүй тусгай үсгийн маягуудыг ашигладаг (интегралууд, нийлбэрүүд, Грек үсгүүд гэх мэт.). X протокол нь эдгээр үсгийн маягуудыг *өөр дээрээ* локалаар суулгахыг шаарддаг. Энэ нь CDROM юм уу эсвэл Mathematica® суулгагдсан хостоос эдгээр үсгийн маягуудын хуулбарыг өөрийн локал машин уруу хуулна гэсэн үг юм. Эдгээр үсгийн маягууд нь CDROM-ийн /cdrom/Unix/Files/SystemFiles/Fonts сан, эсвэл таны хатуу хөтчийн /usr/local/mathematica/SystemFiles/Fonts санд ихэвчлэн хадгалагддаг. Үсгийн жинхэнэ маягууд нь Type1 болон X дэд сангуудад байрладаг. Доор тайлбарласан тэдгээрийг ашиглах хэд хэдэн аргууд байдаг.

Эхний арга нь тэдгээрийг /usr/X11R6/lib/X11/fonts дахь байгаа үсгийн маягийн сангуудын аль нэг уруу хуулах явдал юм. Энэ нь үсгийн маягийн нэрсийг нэмж, эхний мөрөн дээр үсгийн маягуудын тоог өөрчилж fonts.dir файлыг засварлахыг шаардана. Мөн өөрөөр та тэдгээрийг хуулсан санд mkfontdir(1) тушаалыг ажиллуулж бас чадах ёстой юм.

Үүнийг хийх хоёр дахь арга нь сангуудыг /usr/X11R6/lib/X11/fonts уруу хуулах явдал юм:

```
# cd /usr/X11R6/lib/X11/fonts
# mkdir X
# mkdir MathType1
# cd /cdrom/Unix/Files/SystemFiles/Fonts
# cp X/* /usr/X11R6/lib/X11/fonts/X
# cp Type1/* /usr/X11R6/lib/X11/fonts/MathType1
# cd /usr/X11R6/lib/X11/fonts/X
# mkfontdir
# cd ../MathType1
# mkfontdir
```

Одоо шинэ үсгийн маягийн сангуудаа өөрийн үсгийн маягийн замдаа нэмнэ:

```
# xset fp+ /usr/X11R6/lib/X11/fonts/X
# xset fp+ /usr/X11R6/lib/X11/fonts/MathType1
# xset fp rehash
```

Хэрэв та Xorg серверийг ашиглаж байгаа бол эдгээр үсгийн маягийн сангуудыг өөрийн xorg.conf файлдаа нэмэн автоматаар дуудагдахаар тохируулж болно.

Хэрэв танд /usr/X11R6/lib/X11/fonts/Type1 гэгдсэн сан *байхгүй* байгаа бол та дээрх жишээн дээрх MathType1 сангийн нэрийг Type1 болгон өөрчилж болно.

# 11.4. Maple™ суулгах нь

Maple™ нь Mathematica®-тай төстэй арилжааны зориулалтын математикийн програм юм. Та энэ програмыг http://www.maplesoft.com/-с худалдан авч дараа нь тэндээ лицензийн файл авахын тулд бүртгүүлэх хэрэгтэй. Энэ програм хангамжийг FreeBSD дээр суулгахын тулд эдгээр хялбар алхмуудыг дагана уу.

1. INSTALL бүрхүүлийн скриптийг бүтээгдэхүүний тархацаас ажиллуулна. Суулгац програмаас хүсэхэд "RedHat" тохируулгыг сонгоорой. Ердийн суулгацын сан нь /usr/local/maple байж болох юм.
2. Хэрэв та тэгж хийгээгүй бол Maple Waterloo Software (http://register.maplesoft.com/) компаниас Maple™-д зориулж лиценз захиж түүнийгээ /usr/local/maple/license/license.dat сан уруу хуулах хэрэгтэй.
3. Maple™-тай цуг ирдэг INSTALL_LIC суулгацын бүрхүүлийн скриптийг ажиллуулж FLEXlm лиценз менежерийг суулгах хэрэгтэй. Лицензийн серверт зориулж өөрийн машиныхаа анхдагч хостын нэрийг зааж өгнө.
4. /usr/local/maple/bin/maple.system.type файлыг доор дурдсанаар нөхөөс хийх хэрэгтэй:

```
   ----- snip ------------------
*** maple.system.type.orig      Sun Jul  8 16:35:33 2001
--- maple.system.type   Sun Jul  8 16:35:51 2001
***************
*** 72,77 ****
--- 72,78 ----
          # the IBM RS/6000 AIX case
          MAPLE_BIN="bin.IBM_RISC_UNIX"
          ;;
+     "FreeBSD"|\
      "Linux")
          # the Linux/x86 case
        # We have two Linux implementations, one for Red Hat and
   ----- snip end of patch -----
```

   `"FreeBSD"|\` мөрийн дараа ямар ч хоосон зай байх ёсгүйг санаарай.

   Энэхүү нөхөөс нь "FreeBSD"-г Линукс системийн төрөл маягаар танихыг Maple™-д заадаг. bin/maple бүрхүүлийн скрипт нь bin/maple.system.type бүрхүүлийн скриптийг дуудаж тэр нь эргээд үйлдлийн системийн нэрийг олохын тулд `uname -a` тушаалыг дууддаг. OS-ийн нэрээс хамаараад энэ нь аль хоёртын файлуудаа ашиглахаа олох болно.

5. Лицензийн серверийг эхлүүлнэ.

   /usr/local/etc/rc.d/lmgrd.sh гэж суулгагдсан дараах скрипт нь `lmgrd`-г эхлүүлэх хялбар арга юм:

```
   ----- snip ------------

#! /bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/X11R6/bi
n
PATH=${PATH}:/usr/local/maple/bin:/usr/local/maple/FLEXlm/UNIX/LINUX
export PATH

LICENSE_FILE=/usr/local/maple/license/license.dat
LOG=/var/log/lmgrd.log

case "$1" in
start)
    lmgrd -c ${LICENSE_FILE} 2>> ${LOG} 1>&2
    echo -n " lmgrd"
    ;;
stop)
    lmgrd -c ${LICENSE_FILE} -x lmdown 2>> ${LOG} 1>&2
    ;;
*)
    echo "Usage: `basename $0` {start|stop}" 1>&2
    exit 64
    ;;
esac

exit 0
   ----- snip ------------
```

6. Maple™-ийг тестээр эхлүүлнэ:

```
% cd /usr/local/maple/bin
% ./xmaple
```

Одоо ажиллаж байх ёстой. Maplesoft уруу та FreeBSD-ийн эх хувилбарыг хүсэж байгаагаа бичихээ мартуузай!

### 11.4.1. Түгээмэл асуудлууд

- FLEXlm лицензийн менежертэй ажиллахад төвөгтэй байж болох юм. Энэ тухай нэмэлт баримтыг http://www.globetrotter.com/ хаягаас олж болно.
- `lmgrd` нь лицензийн файлыг хэтэрхий голж шилдэг, бас хэрэв ямар нэгэн асуудал байвал core файл үүсгэх дуртайгаараа алдартай програм юм. Зөв лицензийн файл иймэрхүү харагдах ёстой:

```
# ====================================================
# License File for UNIX Installations ("Pointer File")
```

```
# ============================================================
SERVER chillig ANY
#USE_SERVER
VENDOR maplelmg

FEATURE Maple maplelmg 2000.0831 permanent 1 XXXXXXXXXXXX \
         PLATFORMS=i86_r ISSUER="Waterloo Maple Inc." \
         ISSUED=11-may-2000 NOTICE=" Technische Universitat Wien" \
         SN=XXXXXXXXX
```

Энд сериал дугаар болон түлхүүрийг 'X' болгосон байна. `chillig` нь хостын нэр юм.

Лицензийн файлыг засварлахдаа "FEATURE" мөрийг (энэ нь лицензийн түлхүүрээр хамгаалагдсан байдаг) оролдохгүй бол ямар ч асуудал гарахгүй ажиллах болно.

## 11.5. MATLAB® суулгах нь

Энэ баримт нь MATLAB® version 6.5-ийн Линуксийн хувилбарыг FreeBSD систем уруу суулгах процессийг тайлбарлах болно. Java Virtual Machine™-г тооцохгүй юм бол энэ нь маш сайн ажилладаг (Java™-ийн ажиллах үеийн орчныг холбох-г үзнэ үү).

MATLAB®-ийн Линуксийн хувилбарыг http://www.mathworks.com дэх MathWorks-с шууд захиалж болно. Лицензийн файлыг авах юм уу эсвэл хэрхэн үүсгэх зааврыг бас авахаа мартуузай. Тэнд байхдаа тэдний програм хангамжийн FreeBSD хувилбарыг хүсэж байгаагаа мэдэгдээрэй.

### 11.5.1. MATLAB® суулгах нь

MATLAB®-г суулгахын тулд доор дурдсаныг хийнэ:

1. Суулгацын CD-г хийж холбоно. Суулгацын скриптийн зөвлөснөөр `root` болно. Суулгах скриптийг эхлүүлэхийн тулд доор дурдсаныг бичнэ:

   ```
   # /compat/linux/bin/sh /cdrom/install
   ```

   Суулгагч нь график орчных юм. Хэрэв та дэлгэц онгойлгож чадахгүй байна гэсэн алдаанууд авбал `setenv HOME ~USER` гэж бичих хэрэгтэй бөгөөд энд байгаа *USER* нь таны su(1) хийсэн хэрэглэгч юм.

2. MATLAB®-ийн root санг асуухад доор дурдсаныг бичнэ: `/compat/linux/usr/local/matlab`.

   Суулгах процессийн үлдсэн хэсэгт хялбараар бичдэг байхын тулд өөрийн бүрхүүлийн хүлээх мөрөн дээр үүнийг бичнэ: `set MATLAB=/compat/linux/usr/local/matlab`

3. MATLAB® лицензийг авч байхдаа зааварласны дагуу лицензийн файлыг засварлана.

Өөрийн дуртай засварлагчаа ашиглан та энэ файлыг урьдчилан бэлдэж, суулгагч танаас үүнийг засварлахыг асуухаас өмнө $MATLAB/license.dat уруу хуулна.

4. Суулгах процессийг гүйцээнэ.

Энд хүрэхэд таны MATLAB® суулгалт дууссан байна. Үүнийг таны FreeBSD систем уруу холбохын тулд дараах алхмууд нь "хийж" өгдөг.

### 11.5.2. Лицензийн Менежерийн эхлүүлэлт

1. Лицензийн менежерийн скриптүүдэд зориулж симболын холбоосуудыг үүсгэнэ:

```
# ln -s $MATLAB/etc/lmboot /usr/local/etc/lmboot_TMW
# ln -s $MATLAB/etc/lmdown /usr/local/etc/lmdown_TMW
```

2. /usr/local/etc/rc.d/flexlm.sh-т эхлүүлэх файлыг үүсгэнэ. Доорх жишээ нь түгээгдсэн $MATLAB/etc/rc.lm.glnx86-ийн өөрчлөгдсөн хувилбар юм. Өөрчлөлтүүд нь файлын байрлалууд болон Линуксийн эмуляц доор ажиллах лицензийн менежерийн эхлүүлэлт байна.

```
#!/bin/sh
case "$1" in
  start)
        if [ -f /usr/local/etc/lmboot_TMW ]; then
              /compat/linux/bin/sh /usr/local/etc/lmboot_TMW -u username &&
echo 'MATLAB_lmgrd'
        fi
        ;;
  stop)
    if [ -f /usr/local/etc/lmdown_TMW ]; then
            /compat/linux/bin/sh /usr/local/etc/lmdown_TMW  > /dev/null 2>&1
    fi
        ;;
  *)
    echo "Usage: $0 {start|stop}"
    exit 1
    ;;
esac

exit 0
```

Файлыг ажиллуулахаар болгох ёстой:

```
# chmod +x /usr/local/etc/rc.d/flexlm.sh
```

Та дээрх *username*-г өөрийн систем дээрх зөв хэрэглэгчээр солих ёстой (`root` биш).

3. Лицензийн менежерийг дараах тушаалаар эхлүүлнэ:

```
# /usr/local/etc/rc.d/flexlm.sh start
```

### 11.5.3. Java™-ийн ажиллах үеийн орчныг холбох

Java™ Runtime Environment (JRE) буюу Ажиллах үеийн орчны холбоосыг FreeBSD дээр ажиллаж байгаа уруу зааж өөрчлөх хэрэгтэй:

```
# cd $MATLAB/sys/java/jre/glnx86/
# unlink jre; ln -s ./jre1.1.8 ./jre
```

### 11.5.4. MATLAB®-ийн эхлүүлэх скриптийг үүсгэх

1. Дараах эхлүүлэх скриптийг /usr/local/bin/matlab-д байрлуулна:

```
#!/bin/sh
/compat/linux/bin/sh /compat/linux/usr/local/matlab/bin/matlab "$@"
```

2. Тэгээд `chmod +x /usr/local/bin/matlab` тушаалыг бичнэ.

Таны emulators/linux_base хувилбараас шалтгаалаад энэ скриптийг ажиллуулахад алдаа гарч болох юм. Үүнээс сэргийлэхийн тулд /compat/linux/usr/local/matlab/bin/matlab файлыг засварлаж доорх:

```
if [ `expr "$lscmd" : '.*->.*'` -ne 0 ]; then
```

(13.0.1 хувилбар дээр энэ нь 410-р мөрөнд байна) мөрийг ингэж өөрчлөх хэрэгтэй:

```
if test -L $newbase; then
```

### 11.5.5. MATLAB®-ийн зогсоох скриптийг үүсгэх

MATLAB®-ийн зөв дуусдаггүй асуудлыг шийдэхэд дараах зүйлсийг хийх хэрэгтэй.

1. $MATLAB/toolbox/local/finish.m файл үүсгэж түүнд ганц мөр нэмнэ:

```
! $MATLAB/bin/finish.sh
```

`$MATLAB` нь үг юм.

Тэр сан дотроо гарахаас өмнө өөрийн ажлын талбарыг хадгалах боломжийг олгох finishsav.m болон finishdlg.m файлуудыг та олох болно. Хэрэв та тэдгээрийн аль нэгийг ашиглаж байгаа бол дээрх мөрийг `save` тушаалын дараа нэн даруй оруулах хэрэгтэй.

2. доор дурдсаныг агуулсан $MATLAB/bin/finish.sh файлыг үүсгэнэ:

```
#!/compat/linux/bin/sh
(sleep 5; killall -1 matlab_helper) &
exit 0
```

3. Файлыг ажиллах боломжтой болгоно:

```
# chmod +x $MATLAB/bin/finish.sh
```

### 11.5.6. MATLAB®-ийг ашиглах

Энд хүрэхэд та `matlab` гэж бичин үүнийг ашиглаж эхлэхэд бэлэн болсон байна.

## 11.6. Oracle® суулгах нь

### 11.6.1. Оршил

Энэ баримт нь Линуксд зориулсан Oracle® 8.0.5 болон Oracle® 8.0.5.1 Enterprise Edition-г FreeBSD машин уруу суулгах процессийг тайлбарлах болно.

### 11.6.2. Линуксийн орчныг суулгах

Та emulators/linux_base болон devel/linux_devtools программууд портын цуглуулгаас суусан эсэхийг шалгаарай. Та эдгээр портын хувьд асуудалтай байгаа бол багцууд юм уу эсвэл портын цуглуулгад байгаа хуучин хувилбаруудыг ашиглах хэрэгтэй болж болох юм.

Хэрэв та ухаалаг агентийг ажиллуулахыг хүсэж байгаа бол Red Hat Tcl багц: tcl-8.0.3-

20.i386.rpm-г бас суулгах хэрэгтэй болно. Албан ёсны RPM порттой (archivers/rpm) багцуудыг суулгах ерөнхий тушаал бол:

```
# rpm -i --ignoreos --root /compat/linux --dbpath /var/lib/rpm package
```

*package*-ийн суулгалт ямар ч алдаа үүсгэх ёсгүй.

## 11.6.3. Oracle® орчныг үүсгэх

Oracle®-г суулгахаасаа өмнө та тохирох орчныг тохируулах хэрэгтэй. Энэ баримт нь Oracle®-г суулгах зааварт тайлбарласныг биш Линуксд зориулсан Oracle®-г FreeBSD дээр ажиллуулахын тулд *яг* юу хийхийг зөвхөн тайлбарладаг.

### 11.6.3.1. Цөмийг тааруулах

Oracle®-г суулгах зааварт тайлбарласнаар хуваалцсан санах ойн хамгийн их хэмжээг тохируулах хэрэгтэй. FreeBSD дээр SHMMAX-г битгий ашиглаарай. SHMMAX нь ердөө л SHMMAXPGS болон PGSIZE-с тооцоологдон гаргагддаг. Тийм болохоор SHMMAXPGS-г тодорхойл. Бусад бүх тохируулгууд зааварт тайлбарласны дагуу ашиглагдаж болно. Жишээ нь:

```
options SHMMAXPGS=10000
options SHMMNI=100
options SHMSEG=10
options SEMMNS=200
options SEMMNI=70
options SEMMSL=61
```

Эдгээр тохируулгуудыг өөрийн Oracle®-ийн хэрэглээнд тааруулж тохируулах хэрэгтэй.

Мөн цөмийн тохиргооны файлдаа дараах тохируулгуудтай байгаа эсэхээ бас шалгаарай:

```
options SYSVSHM #SysV shared memory
options SYSVSEM #SysV semaphores
options SYSVMSG #SysV interprocess communication
```

### 11.6.3.2. Oracle® бүртгэл

Өөр бусад бүртгэл үүсгэдэг шигээ oracle бүртгэл үүсгэнэ. oracle бүртгэл нь Линуксийн бүрхүүл өгөх шаардлагатай цорын ганц тусгай бүртгэл юм. /etc/shells файлд /compat/linux/bin/bash-г нэмээд oracle хэрэглэгчийн бүрхүүлийг /compat/linux/bin/bash гэж тохируулна.

### 11.6.3.3. Орчин

ORACLE_HOME болон ORACLE_SID зэрэг ердийн Oracle® хувьсагчуудаас гадна та дараах орчны хувьсагчуудыг тохируулах ёстой:

| Хувьсагч | Утга |
| --- | --- |
| LD_LIBRARY_PATH | $ORACLE_HOME/lib |
| CLASSPATH | $ORACLE_HOME/jdbc/lib/classes111.zip |
| PATH | /compat/linux/bin /compat/linux/sbin /compat/linux/usr/bin /compat/linux/usr/sbin /bin /sbin /usr/bin /usr/sbin /usr/local/bin $ORACLE_HOME/bin |

Бүх орчны хувьсагчуудыг .profile-д тохируулахыг зөвлөдөг. Гүйцэд жишээ бол дараах байна:

```
ORACLE_BASE=/oracle; export ORACLE_BASE
ORACLE_HOME=/oracle; export ORACLE_HOME
LD_LIBRARY_PATH=$ORACLE_HOME/lib
export LD_LIBRARY_PATH
ORACLE_SID=ORCL; export ORACLE_SID
ORACLE_TERM=386x; export ORACLE_TERM
CLASSPATH=$ORACLE_HOME/jdbc/lib/classes111.zip
export CLASSPATH
PATH=/compat/linux/bin:/compat/linux/sbin:/compat/linux/usr/bin
PATH=$PATH:/compat/linux/usr/sbin:/bin:/sbin:/usr/bin:/usr/sbin
PATH=$PATH:/usr/local/bin:$ORACLE_HOME/bin
export PATH
```

### 11.6.4. Oracle®-ийг суулгах

Линуксийн эмуляторын хайхрамжгүй бүрэн бүтэн бус байдлаас болоод суулгагчийг эхлүүлэхээсээ өмнө та .oracle санг /var/tmp дотор үүсгэх хэрэгтэй болдог. Үүнийг `oracle` хэрэглэгч эзэмшихээр болгох хэрэгтэй. Та Oracle®-г ямар ч асуудалгүйгээр суулгаж чадах ёстой. Хэрэв танд асуудлууд учирвал эхлээд өөрийн Oracle® тархац ба/эсвэл тохиргоог шалгах хэрэгтэй! Oracle®-г суулгасныхаа дараа та дараах хоёр дэд хэсэгт тайлбарласан нөхөөсүүдийг хийж өгөх хэрэгтэй.

Байнга учирдаг асуудал бол TCP протоколын хувиргагч зөв суулгагдаагүй байдаг явдал юм. Үүнээс болоод та ямар ч TCP сонсогч эхлүүлж чадахгүй. Дараах үйлдлүүд нь энэ асуудлыг шийдэхэд тусалдаг:

```
# cd $ORACLE_HOME/network/lib
# make -f ins_network.mk ntcontab.o
# cd $ORACLE_HOME/lib
# ar r libnetwork.a ntcontab.o
# cd $ORACLE_HOME/network/lib
# make -f ins_network.mk install
```

root.sh-г дахин ажиллуулахаа битгий мартаарай!

#### 11.6.4.1. root.sh-д нөхөөс хийх

Oracle®-г суулгаж байхад `root` эрхээр хийгдэх шаардлагатай зарим үйлдлүүд root.sh гэж нэрлэгдсэн бүрхүүлийн скриптэд бичигдсэн байдаг. Энэ скрипт нь orainst санд бичигдсэн байдаг. `chown` тушаалын зөв байрлалыг ашиглахаар болгож дараах нөхөөсийг root.sh-д хийх хэрэгтэй, эсвэл өөрөөр скриптийг Линуксийн эх бүрхүүл дээр ажиллуулах хэрэгтэй.

```
*** orainst/root.sh.orig Tue Oct 6 21:57:33 1998
--- orainst/root.sh Mon Dec 28 15:58:53 1998
***************
*** 31,37 ****
# This is the default value for CHOWN
# It will redefined later in this script for those ports
# which have it conditionally defined in ss_install.h
! CHOWN=/bin/chown
#
# Define variables to be used in this script
--- 31,37 ----
# This is the default value for CHOWN
# It will redefined later in this script for those ports
# which have it conditionally defined in ss_install.h
! CHOWN=/usr/sbin/chown
#
# Define variables to be used in this script
```

CD-ээс Oracle®-г та суулгахгүй үед root.sh-д зориулсан эхийг нөхөж болно. Үүнийг rthd.sh гэдэг бөгөөд эх модны orainst санд байрладаг.

#### 11.6.4.2. genclntsh-д нөхөөс хийх

`genclntsh` скрипт нь хуваалцсан ганц клиентийн санг үүсгэхэд ашиглагддаг. Үзүүлбэрүүдийг бүтээж байхад үүнийг ашигладаг. `PATH`-н тодорхойлолтыг тайлбар болгохын тулд дараах нөхөөсийг хийх хэрэгтэй:

```
*** bin/genclntsh.orig Wed Sep 30 07:37:19 1998
--- bin/genclntsh Tue Dec 22 15:36:49 1998
***************
*** 32,38 ****
#
# Explicit path to ensure that we're using the correct commands
#PATH=/usr/bin:/usr/ccs/bin export PATH
! PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin export PATH
#
# each product MUST provide a $PRODUCT/admin/shrept.lst
--- 32,38 ----
#
# Explicit path to ensure that we're using the correct commands
#PATH=/usr/bin:/usr/ccs/bin export PATH
! #PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin export PATH
```

```
#
# each product MUST provide a $PRODUCT/admin/shrept.lst
```

### 11.6.5. Oracle®-г ажиллуулах

Зааврууд дагасны дараа та Oracle®-г Линукс дээр ажиллуулж байгаа юм шиг ажиллуулж чадах ёстой.

## 11.7. Нэмэлт сэдвүүд

Хэрэв та Линуксийн хоёртын нийлэмж хэрхэн ажилладгийг мэдэхийг хүсэж байгаа бол энэ хэсэг нь таны уншихыг хүсэж байгаа хэсэг юм. Доор бичигдсэн зүйлийн ихэнх нь Тэрри Лэмбэрт tlambert@primenet.com-ийн FreeBSD chat захидлын жагсаалт руу бичсэн цахим захидал (Message ID: <199906020108.SAA07001@usr09.primenet.com>) дээр тулгуурласан байгаа.

### 11.7.1. Хэрхэн ажилладаг вэ?

FreeBSD нь "execution class loader" буюу ажиллуулах ангилал дуудагч гэгддэг хийсвэрлэлттэй байдаг. Энэ нь execve(2) системийн дуудлага уруу хийгдэх шаантаг юм.

Юу болдог вэ гэхээр FreeBSD нь ямар нэгэн бүрхүүлийн тайлбарлагчид эсвэл бүрхүүлийн скриптүүдийг ажиллуулахын тулд `#!` дуудагч уруу ордог нэг дуудагчийн оронд дуудагчдын жагсаалттай байдаг.

Уламжлалаар бол UNIX® тавцангийн дуудагч нь хоёртын файлыг системд мэдэгдэж байгаа эсэхийг мэдэхийн тулд шидэт тоог (ерөнхийдөө файлын эхний 4 эсвэл 8 байт) шалгадаг бөгөөд хэрэв мэдэгдэж байвал хоёртын дуудагчийг ажиллуулдаг ганц дуудагч юм.

Хэрэв энэ нь системд зориулагдсан хоёртын төрөл биш бол execve(2) дуудлага амжилгүй болон буцаж бүрхүүл үүнийг бүрхүүлийн тушаалууд маягаар ажиллуулж эхлэхийг оролддог.

"Тухайн үеийн бүрхүүл ямар байгаа" түүнийг анхдагч гэж таамагладаг.

Дараа нь эхний хоёр тэмдэгтүүдийг шалгах засварыг (hack) sh(1)-д зориулж хийсэн бөгөөд хэрэв тэдгээр нь `:\n` бол энэ нь csh(1) бүрхүүлийг ажиллуулдаг (энэ засварыг SCO анхлан хийсэн гэж бид итгэдэг).

Төгсгөлийн дараах дараагийн хоосон зайны дараа байдаг тэмдэгтүүдийн дараа /bin/sh уруу буцдаг тэмдэгтүүд байдаг болохоор одоо FreeBSD юу хийдэг вэ гэхээр дуудагчдын жагсаалтаар тайлбарлагчдын талаар мэддэг ерөнхий `#!` дуудагчтай явдаг.

Линуксийн ABI дэмжлэгийн хувьд FreeBSD нь шидэт тоог ELF хоёртын файл гэж хардаг (одоогоор энэ нь FreeBSD, Solaris™, Линукс болон ELF дүрсний төрөл бүхий бусад OS-ийг хооронд нь ялгадаггүй).

ELF дуудагч нь тусгай *тамга* хайдаг бөгөөд энэ нь ELF дүрс дэх тайлбар хэсэг юм. Энэ нь SVR4/Solaris™ ELF хоёртын файлуудад байдаггүй.

Линукс хоёртын файлууд нь ажиллахын тулд тэдгээр нь brandelf(1)-ээр Линукс гэж *тамгалагдах* ёстой байдаг:

```
# brandelf -t Linux file
```

Үүнийг хийсний дараа ELF дуудагч нь файлд Линукс тамгыг харах болно.

ELF дуудагч Линукс тамгыг харах үед дуудагч proc бүтэц дэх заагчийг сольдог. Энэ заагчаар бүх системийн дуудлагууд индекслэгддэг (уламжлалт UNIX® систем дээр энэ нь системийн дуудлагуудыг агуулах sysent[] бүтцийн массив байх юм). Үүнээс гадна процесс нь дохионы трамплиний кодонд зориулсан занга векторыг тусгайлан зохицуулахад болон Линуксийн цөмийн модулиар зохицуулагддаг бусад хэд хэдэн (жижиг) засваруудад зориулагдаж тэмдэглэгддэг.

Линукс системийн дуудлагын вектор нь бусад зүйлүүдээс гадна цөмийн модульд хаягууд нь байдаг sysent[] оруулгуудын жагсаалтыг агуулдаг.

Системийн дуудлага Линуксийн хоёртын файлаар дуудагдахад занга код системийн дуудлагын функц заагчийн хаягийг proc бүтцээс авч FreeBSD-ийн биш Линуксийн системийн дуудлагын оруулгын цэгүүдийг авдаг.

Мөн Линукс горим нь хайлтыг динамикаар *дахин эхлүүлдэг*; энэ нь файлын системийн холболтууд дахь union тохируулга (unionfs файлын системийн төрөл *биш*!) хийдэг тэр зүйл юм. Эхлээд /compat/linux/original-path сан дахь файлыг хайх бөгөөд *дараа нь* хэрэв энэ нь амжилттүй болвол хайлт /original-path санд хийгддэг. Энэ нь бусад хоёртын файлуудыг шаарддаг хоёртын файлуудыг ажиллаж чадахаар (өөрөөр хэлбэл Линуксийн хэрэгслийн цуглуулга бүгдээрээ Линуксийн ABI дэмжлэгийн доор ажиллаж чаддаг байхаар) болгодог. Линуксийн хоёртын файлууд нь тохирох Линуксийн хоёртын файлууд байхгүй бол FreeBSD-ийн хоёртын файлуудыг дуудаж ажиллуулж бас чадна гэсэн үг бөгөөд Линуксийн хоёртын файлуудыг Линукс дээр ажиллахгүй байгааг нь хэлж чадахааргүй болгохын тулд та uname(1) тушаалыг /compat/linux санд байрлуулж болно гэсэн үг юм.

Үндсэндээ Линуксийн цөм FreeBSD цөмд байдаг; цөмийн үзүүлдэг бүх үйлчилгээнүүдийг хийдэг, доор нь орших төрөл бүрийн функцууд нь FreeBSD-ийн системийн дуудлагын хүснэгтийн оруулгууд болон Линуксийн системийн дуудлагын хүснэгтийн оруулгуудтай ижил байдаг: файлын системийн үйлдлүүд, виртуал санах ойн үйлдлүүд, дохио хүргэлт, System V IPC, гэх мэт... Цорын ганц ялгаа нь FreeBSD-ийн хоёртын файлууд FreeBSD-ийн *цавуу* функцуудыг, Линуксийн хоёртын файлууд Линуксийн *цавуу* функцуудыг авдаг явдал юм (ихэнх хуучин OS-үүд зөвхөн өөрсдийн *цавуу* функцуудтай байсан: дуудлага хийж байгаа процессийн proc бүтэц дэх динамикаар эхлүүлэгдсэн заагчаар хийгдсэн функцуудын хаягуудын оронд статик глобал sysent[] бүтцийн массив дахь функцуудын хаягуудтай байсан).

Аль нь эх FreeBSD ABI вэ? Энэ нь хамаагүй юм. Үндсэндээ цорын ганц ялгаа нь (зөвхөн одоогоор; эдгээр нь ирээдүйн хувилбаруудад амархан өөрчлөгдөж болох бөгөөд магадгүй үүний дараагаар хийгдэх байх) FreeBSD-ийн *цавуу* функцууд нь цөмд статикаар холбогдсон байдаг бөгөөд Линуксийн *цавуу* функцууд нь статикаар холбогдож эсвэл тэдгээрт цөмийн модулийн тусламжтайгаар хандаж болдог явдал юм.

Тиймээ, гэхдээ энэ нь жинхэнэ эмуляц мөн үү? Үгүй ээ. Энэ нь ABI шийдэл болохоос эмуляц биш юм. Ямар ч эмулятор (эсвэл дүр үзүүлэгч (simulator) (дараагийн асуултыг асуулгахгүйн тулд)) оролцоогүй.

Тэгэхээр яагаад энэ нь заримдаа "Линукс эмуляц" гэгддэг юм бэ? FreeBSD-г худалдахад хэцүү болгохын тулд! Тиймээ, юу болж байгааг тайлбарлаж байгаагаас өөр зүйлгүй байх тэр үед уламжлалт шийдэл хийгдсэн болохоор тэр юм; хэрэв та кодыг эмхэтгээгүй эсвэл модулийг дуудаагүй бол Линуксийн хоёртын файлуудыг FreeBSD ажиллуулдаг гэж хэлэх нь үнэн биш бөгөөд юу дуудагдсаныг тайлбарлах үг хэрэгтэй байсан болохоор -"Линукс эмулятор" гэсэн үг гарсан юм.

# Part III: Системийн Удирдлага

FreeBSD гарын авлагын үлдсэн бүлгүүд нь FreeBSD системийн удирдлагын бүх ойлголтуудыг хамрах болно. Бүлэг бүрийн эхэнд, уг бүлгийг уншсанаар таны юу сурч авахыг тайлбарлах бөгөөд уншиж эхлэхээс өмнө юуг мэдсэн байх ёстойг бас хэлж өгөх болно.

Эдгээр бүлгүүд нь танд мэдээлэл хэрэгтэй үед уншигдахаар зориулагдсан юм. Та ямар нэгэн тодорхой дарааллаар унших хэрэггүй бөгөөд FreeBSD ашиглаж эхлэхээсээ өмнө бүгдийг нь унших шаардлагагүй болно.

# Chapter 12. Тохиргоо ба Тааруулалт

## 12.1. Ерөнхий агуулга

FreeBSD-ийн хамгийн чухал зүйлүүдийн нэг нь системийн тохиргоо юм. Зөв системийн тохиргоо нь ирээдүйн шинэчлэлтүүдийн үед толгойн өвчин гаргахгүй байхад тусална. Энэ бүлэг FreeBSD системийг тааруулахад хэрэглэгддэг зарим нэг параметрүүд болон тохиргооны процессийн талаар илүү тайлбарлах болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Файлын системүүд болон хуваалтуудтай хэрхэн үр ашигтай ажиллах талаар.
- rc.conf тохиргоо болон /usr/local/etc/rc.d эхлэлийн системүүдийн үндсүүд.
- Сүлжээний картыг хэрхэн тохиргоо болон тест хийх талаар.
- Сүлжээний төхөөрөмж дээрээ виртуал хостууд хэрхэн тохируулах талаар.
- /etc дэх төрөл бүрийн тохиргооны файлыг хэрхэн ашиглах талаар.
- `sysctl` хувьсагчуудыг ашиглан FreeBSD-ийг хэрхэн тааруулах талаар.
- Дискний хурдан ажиллагааг хэрхэн тааруулах болон цөмийн хязгаарууд ыг хэрхэн өөрчлөх талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- UNIX® болон FreeBSD-ийн үндсийг ойлгох (Юниксийн үндэс).
- Цөмийн тохиргоо/хөрвүүлэлтийн үндсүүдийн талаар ойлголттой байх (FreeBSD цөмийг тохируулах нь).

## 12.2. Эхний Тохиргоо

### 12.2.1. Хуваалтын байрлал

#### 12.2.1.1. Үндсэн Хуваалтууд

bsdlabel(8) болон sysinstall(8) ашиглан файлын системүүдийг байрлуулахдаа хатуу хөтлөгчүүд өгөгдлийг дотоод замуудаас илүү гаднах замуудаас хурдан шилжүүлдгийг санаарай. Тиймээс жижиг, байнга ханддаг файлын системүүд хөтлөгчийн гадна тал уруу ойрхон байх ёстой бөгөөд /usr зэрэг том хуваалтууд дискийн дотор тал уруу байх хэрэгтэй. Хуваалтуудыг иймэрхүү дарааллаар байрлуулах нь зөв юм: root, swap, /var, /usr.

/var хуваалтын хэмжээ төлөвлөсөн машины хэрэглээг тусгадаг. /var файлын систем нь шуудангийн хайрцгууд, бүртгэлийн файлууд, болон принтерийн spool агуулдаг. Шуудангийн хайрцгууд болон бүртгэлийн файлууд хичнээн хэрэглэгч байгаа болон ямар хугацаанд бүртгэлийн файлууд байхаас хамаараад төсөөлөшгүй хэмжээнд хүртэл ихсэж болдог. Ихэнх хэрэглэгчдийн хувьд /var-д нэг гигабайт сул зай байхад хангалттай байдаг.

/var/tmp-д ихээхэн хэмжээний дискийн зай шаардагддаг цөөхөн тохиолдол

> байдаг. Шинэ програм хангамжийг pkg_add(1) ашиглан суулгахад багцлах хэрэгслүүд багцын түр зуурын хуулбарыг /var/tmp-д задалдаг. /var/tmp-д хангалттай дискийн чөлөөтэй зай байхгүй бол Firefox, OpenOffice эсвэл LibreOffice зэрэг томоохон програм хангамжийн багцуудыг суулгахад төвөгтэй байж болох юм.

/usr хуваалт ports(7) цуглуулга (байлгахыг зөвлөдөг), болон эх код (заавал биш) зэрэг системийг дэмжихэд шаардлагатай ихэнх файлуудыг агуулдаг. Портууд болон үндсэн системийн эхүүдийг суулгалтын үед сонгох боломжтой боловч бид энэ хуваалтад хамгийн багаар бодоход 2 гигабайт байхыг зөвлөдөг.

Хуваалтын хэмжээг сонгохдоо зайн шаардлагыг бодох хэрэгтэй. Нэг хуваалт нь бараг л ашиглагдахгүй байхад нөгөө нь зайгүй болж байх нь асуудал юм.

> sysinstall(8)-ийн `Auto-defaults` хуваалтын хэмжээг өгөгч нь заримдаа /var болон / хуваалтуудад боломжоос бага хэмжээг сонгодгийг зарим хэрэглэгчид олсон байна. Хуваалтыг ухаалгаар харамгүй хийгээрэй.

#### 12.2.1.2. Swap Хуваалт

Swap хуваалтын хэмжээ системийн санах ойг (RAM) хоёр дахин авсан хэмжээтэй байх ёстой. Жишээлбэл машин 128 мегабайт санах ойтой бол swap файл 256 мегабайт байх ёстой. Бага санах ойтой системүүд их swap-тай бол илүү хурдан ажиллаж болох юм. 256 мегабайтаас бага swap-ийг хэрэглэхийг зөвлөдөггүй бөгөөд санах ойн өргөтгөл хэрэгтэй. Цөмийн VM хуудаслах алгоритмууд нь багаар бодоход гол санах ойг хоёр дахин авсантай тэнцэх swap хуваалттай байх үед хамгийн хурдан ажиллахаар тааруулагдсан байдаг. Хэтэрхий бага swap тохируулах нь VM хуудас скан хийх кодыг үр ашиггүйтэлд хүргэж илүү санах ой хожим нэмэхэд асуудал үүсгэж болох юм.

Олон SCSI дискнүүд бүхий (эсвэл олон IDE дискнүүд өөр өөр хянагчууд дээр ажиллаж байгаа) томоохон системүүдэд swap-ийг хөтлөгч болгон дээр (4 хөтлөгч хүртэл) тохируулахыг зөвлөдөг. Swap хуваалтууд нь ойролцоогоор адилхан хэмжээний байх шаардлагатай. Цөм дурын хэмжээтэй ажиллаж чадах боловч дотоод өгөгдлийн бүтцүүд хамгийн том swap хуваалтыг 4 дахин авсантай адил хэмжээгээр томрох боломжтой. Swap хуваалтуудыг ойролцоогоор адил хэмжээтэй байлгах нь swap зайг дискнүүдийн дагуу оновчтойгоор судал үүсгэх боломжийг цөмд олгодог. Swap их ашиглагддаггүй байсан ч гэсэн том swap хэмжээ байж болно. Хүчээр дахин ачаалагдах үед дагаж хаагдсан програмаас өгөгдлийг сэргээх нь амархан байж болох юм.

#### 12.2.1.3. Яагаад Хуваах хэрэгтэй гэж?

Зарим хэрэглэгчид ганц том хуваалт байхад болно гэж боддог, гэхдээ энэ нь яагаад буруу болох хэд хэдэн шалтгаан бий. Нэгдүгээрт хуваалт болгон өөр өөр ажиллагааны шинж чанаруудтай бөгөөд тэдгээрийг тусгаарласнаар файлын системийг тэдгээрт тааруулах боломжийг олгодог. Жишээ нь root болон /usr хуваалтууд байнга бичигдэхээсээ илүү ихэвчлэн уншигддаг. Харин уншилт болон бичилт /var болон /var/tmp-д байнга хийгддэг.

Системийг зөв хуваснаар ачаалалтай хуваалтуудад хийсэн жижиг бичилтээр гарсан хэсэглэлт илүүдэж байнга уншигддаг хуваалтууд уруу хальдаггүй. Бичилт-ачаалсан

хуваалтуудыг дискний ирмэг уруу байрлуулах нь бичилт ихэвчлэн хийгддэг хуваалтууд дахь I/O ажиллагааг хурдасгадаг. Том хуваалтуудад I/O-н хурдан ажиллагаа хэрэгтэй байж болох ч тэдгээрийг дискний ирмэг уруу илүүтэй ойртуулах нь /var-ийг ирмэг уруу шилжүүлснээс илүү мэдэгдэхүйц хурдан ажиллагаанд хүргэхгүй. Эцэст нь найдвартай байдлыг бодох ёстой. Ихэвчлэн уншигддаг, жижиг, цэвэрхэн root хуваалт хэцүү сүйрэл болоход сэргэх боломж нь хамаагүй илүү байна.

## 12.3. Гол Тохиргоо

Системийн тохиргооны мэдээлэл /etc/rc.conf дотор байдаг. Энэ файл нь өргөн хүрээний, зарчмын хувьд системийг эхлэх үед системийг тохируулахад ашиглагддаг тохиргооны мэдээллүүдээс тогтоно. Үүний нэр нь шууд утгыг тодорхойлно; энэ нь rc* файлуудад зориулсан тохиргооны мэдээлэл юм.

Администратор /etc/defaults/rc.conf-ийн анхдагч утгуудыг rc.conf файлд өөрчилж оруулах хэрэгтэй. Анхдагчуудын файл /etc уруу хуулагдах ёсгүй - энэ нь жишээ биш анхдагч утгуудыг агуулдаг. Бүх системийн холбогдолтой өөрчлөлтүүд rc.conf файлд өөрт нь хийгдэх ёстой.

Удирдлагын нэмэлт ачааллыг байнга бага байлгахын тулд сайт дагуух тохиргоог системийн тусгайлсан тохиргооноос тусгаарлах хэд хэдэн стратеги кластер хийгдсэн програмуудад байж болох юм. Тухайн системийн тохиргоог /etc/rc.conf.local файлд байрлуулах нь зүйтэй. Жишээ нь:

- /etc/rc.conf:

```
sshd_enable="YES"
keyrate="fast"
defaultrouter="10.1.1.254"
```

- /etc/rc.conf.local:

```
hostname="node1.example.org"
ifconfig_fxp0="inet 10.1.1.1/8"
```

Дараа нь rc.conf файл систем болгонд `rsync` эсвэл адил програмаар түгээгдэж болох бөгөөд харин rc.conf.local файл нь өөр өөр хэвээр байх болно.

sysinstall(8) эсвэл `make world` ашиглан системийг шинэчлэхэд rc.conf файлыг дарж бичихгүй, тэгэхээр системийн тохиргооны мэдээлэл хаягдахгүй.

/etc/rc.conf тохиргооны файлыг sh(1)-р уншуулдаг. Энэ нь системийн операторуудад уг файлд тодорхой хэмжээний логик нэмэх боломжийг олгодог бөгөөд ингэснээр илүү нарийн төвөгтэй тохиргооны хувилбарууд үүсгэхэд тусалдаг. Энэ талаар дэлгэрэнгүйг rc.conf(5)-с үзнэ үү.

## 12.4. Програмын Тохиргоо

Ерөнхийдөө суулгасан програмууд нь өөрийн дүрэм гэх мэт онцлогтой өөр өөрийн тохиргооны файлуудтай байдаг. Эдгээр файлуудыг багц удирдах хэрэгслүүдээр амархан олж удирдаж болохоор үндсэн системээс тусад нь байлгах нь чухал юм.

Ерөнхийдөө эдгээр файлууд нь /usr/local/etc дотор суулгагддаг. Програм их олон тооны тохиргооны файлуудтай тохиолдолд тэдгээрийг агуулж дэд сан үүсгэгдэнэ.

Ихэнхдээ порт эсвэл багц суухад жишээ тохиргооны файлууд бас суудаг. Эдгээр нь ихэнхдээ .default дагавраар танигддаг. Хэрэв програмын хувьд тохиргооны файлууд байхгүй байвал тэдгээрийг .default файлуудыг хуулж үүсгэнэ.

Жишээ нь /usr/local/etc/apache санд байгаа файлуудыг үзье:

```
-rw-r--r--  1 root  wheel   2184 May 20  1998 access.conf
-rw-r--r--  1 root  wheel   2184 May 20  1998 access.conf.default
-rw-r--r--  1 root  wheel   9555 May 20  1998 httpd.conf
-rw-r--r--  1 root  wheel   9555 May 20  1998 httpd.conf.default
-rw-r--r--  1 root  wheel  12205 May 20  1998 magic
-rw-r--r--  1 root  wheel  12205 May 20  1998 magic.default
-rw-r--r--  1 root  wheel   2700 May 20  1998 mime.types
-rw-r--r--  1 root  wheel   2700 May 20  1998 mime.types.default
-rw-r--r--  1 root  wheel   7980 May 20  1998 srm.conf
-rw-r--r--  1 root  wheel   7933 May 20  1998 srm.conf.default
```

Файлын хэмжээнүүд нь зөвхөн srm.conf файл өөрчлөгдсөнийг харуулж байна. Apache портын дараагийн шинэчлэл энэ өөрчлөгдсөн файлыг дарж хуулахгүй.

## 12.5. Үйлчилгээнүүдийг эхлүүлэх нь

Олон хэрэглэгчид Портуудын Цуглуулгаас гаравдагч програм хангамжуудыг FreeBSD дээр суулгахаар сонгодог. Ихэнх тохиолдолд програм хангамжийг систем ачаалахад эхлүүлэхээр тохируулах шаардлагатай байж болох юм. mail/postfix эсвэл www/apache22 зэрэг үйлчилгээнүүд нь системийг ачаалахад эхлүүлж болох програм хангамжийн багцуудын зөвхөн хоёрхон жишээ юм. Энэ хэсэгт гаравдагч програм хангамжийг ажиллуулах процедурын талаар тайлбарлах болно.

FreeBSD дээр cron(8) зэрэг ихэнх үйлчилгээнүүд системийн эхлүүлэх скриптүүдийн тусламжтай эхэлдэг. Эдгээр скриптүүд FreeBSD эсвэл үйлдвэрлэгчийн хувилбараас хамааран өөр өөр байна; гэхдээ хамгийн чухал авч үзэх зүйл нь тэдгээрийн эхлэх тохиргоог энгийн эхлүүлэх скриптүүдээр хийх боломжтой явдал юм.

### 12.5.1. Програмын Өргөтгөсөн Тохиргоо

Одоогийн FreeBSD-ийн rc.d-г агуулдаг нь програмын эхлүүлэх тохиргоог илүү хялбар, боломжтой болгосон. rc.d хэсэгт хэлэлцсэн түлхүүр үгүүдийг ашиглан програмууд жишээ нь DNS зэрэг зарим үйлчилгээнүүдийн дараа ажиллахаар тохируулагдаж болно; эхлүүлэх

скриптүүдэд хатуугаар бичигдсэн тугуудын оронд rc.conf-оор нэмэлт тугуудыг өгөхийг зөвшөөрч болох гэх мэт. Үндсэн скрипт дараах байдлаар харагдаж болно:

```
#!/bin/sh
#
# PROVIDE: utility
# REQUIRE: DAEMON
# KEYWORD: shutdown

. /etc/rc.subr

name="utility"
rcvar=utility_enable
command="/usr/local/sbin/utility"

load_rc_config $name

#
# DO NOT CHANGE THESE DEFAULT VALUES HERE
# SET THEM IN THE /etc/rc.conf FILE
#
utility_enable=${utility_enable-"NO"}
pidfile=${utility_pidfile-"/var/run/utility.pid"}

run_rc_command "$1"
```

Энэ скрипт нь өгөгдсөн utility-г `DAEMON` псевдо үйлчилгээний дараа ажиллуулахаар тохируулагдсан. Мөн PID, эсвэл процессийн ID файлыг заах болон дагах аргыг бас хангадаг.

Энэ програм дараах мөрийг /etc/rc.conf файлд оруулж болно:

```
utility_enable="YES"
```

Энэхүү арга нь тушаалын мөрийн нэмэлт өгөгдлүүдийг илүү хялбараар удирдах боломжийг зөвшөөрдөг бөгөөд /etc/rc.subr дахь анхдагч функцуудыг оруулах, rcorder(8) хэрэгсэлтэй нийцтэй байх, болон rc.conf файлын тусламжтай хялбараар тохиргоо хийх боломжийг бас хангадаг.

### 12.5.2. Үйлчилгээнүүдийг эхлүүлэхийн тулд үйлчилгээнүүдийг ашиглах нь

POP3 сервер дэмонууд, IMAP зэрэг бусад үйлчилгээнүүд inetd(8) ашиглан эхэлж болдог. Энэ нь Портуудын Цуглуулгаас /etc/inetd.conf файлд нэмэгдэх мөр бүхий эсвэл одоогийн байгаа мөрүүдийн нэгнээс тайлбарыг болиулж идэвхжүүлдэг үйлчилгээний хэрэгслийг суулгаснаар хэрэгждэг. inetd болон түүний тохиргоотой ажиллах талаар inetd хэсэгт гүнзгий тайлбарласан байгаа болно.

Зарим тохиолдолд cron(8) ашиглан системийн үйлчилгээнүүдийг эхлүүлэх нь илүү ашигтай

байж болох юм. Энэ арга нь хэд хэдэн давуу талуудтай бөгөөд учир нь `cron` эдгээр процессуудыг crontab-н файлын эзэмшигчийн эрхээр ажиллуулдаг. Энэ нь ердийн хэрэглэгчдэд зарим програмуудыг эхлүүлж ажиллагааг хангах боломжийг олгодог.

`cron` хэрэгсэл `@reboot` гэсэн бусдад байхгүй боломжийг олгодог бөгөөд цаг хугацааг заах хэсэгт ашиглагдах боломжтой. Энэ нь системийг эхлүүлэх явцад cron(8) эхлэх үед тухайн ажлыг ажиллуулдаг.

## 12.6. `cron` хэрэгслийг тохируулах нь

FreeBSD-ийн хамгийн ашигтай хэрэгслүүдийн нэг нь cron(8) юм. `cron` хэрэгсэл ард ажилладаг бөгөөд /etc/crontab файлыг байнга шалгаж байдаг. `cron` хэрэгсэл /var/cron/tabs сангаас шинэ crontab файлуудыг бас шалгадаг. Эдгээр crontab файлууд нь тусгай функцуудыг агуулдаг бөгөөд эдгээрийг `cron` тодорхой хугацаанд ажиллуулах ёстой байдаг.

`cron` хэрэгсэл системийн crontab болон хэрэглэгчийн crontab гэсэн хоёр төрлийн тохиргооны файлыг ашигладаг. Энэ хоёр хэлбэршилтийн зөвхөн ялгаа нь зургаа дахь талбараас хойш юм. Системийн crontab дээр `cron` тушаал зургаа дахь талбар дээр зааж өгсөн хэрэглэгчээр тушаалыг ажиллуулна. Хэрэглэгчийн crontab дээр crontab үүсгэсэн хэрэглэгчээр бүх тушаалыг ажиллуулах ба зургаа дахь талбар нь хамгийн сүүлийн талбар юм; энэ нь аюулгүй байдлын нэг чухал боломж юм.

Хэрэглэгчийн crontab-ууд нь хэрэглэгчдэд `root` эрхийн шаардлагагүйгээр бодлогуудыг цагийн хуваариар ажиллуулах боломж олгодог. Хэрэглэгчийн crontab дахь тушаалууд нь crontab-ийг эзэмшиж байгаа хэрэглэгчийн эрхээр ажилладаг.

`root` хэрэглэгч бас бусад хэрэглэгчийн нэгэн адил хэрэглэгчийн crontab-тай байж болно. `root` хэрэглэгчийн crontab нь /etc/crontab-аас (системийн crontab) тусдаа байна. Яагаад гэвэл системийн crontab нь заасан тушаалуудыг root эрхээр ажиллуулдаг учраас `root` хэрэглэгчийн хувьд ихэнхдээ хэрэглэгчийн crontab шаардлагагүй байдаг.

Системийн crontab /etc/crontab файлыг харцгаая:

```
# /etc/crontab - root's crontab for FreeBSD
#
# $FreeBSD: src/etc/crontab,v 1.32 2002/11/22 16:13:39 tom Exp $
#①
#
SHELL=/bin/sh
PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin ②
HOME=/var/log
#
#
#minute hour    mday    month   wday    who command ③
#
#
```

```
*/5 *   *   *   *   root    /usr/libexec/atrun ④
```

① FreeBSD-ийн ихэнх тохиргооны файлуудын адил `#` тэмдэгтээр эхэлсэн мөрүүд тайлбар юм. Тайлбарыг хүсэж байгаа үйлдэл нь юу болох яагаад хийгдэж байгааг сануулах зорилгоор файлд тавьж болдог. Тайлбаруудыг тушаал байгаа мөрд хийж болохгүй бөгөөд ингэсэн тохиолдолд тушаалын хэсэг мэтээр ойлгогдоно; тэдгээр нь шинэ мөрөнд байх ёстой. Хоосон мөрүүдийг тооцохгүй.

② Эхлээд орчин тодорхойлогдох шаардлагатай. Тэнцүүгийн (`=`) тэмдэг орчны тохиргоог тодорхойлоход ашиглагддаг бөгөөд энэ жишээн дээр `SHELL`,`PATH`, болон `HOME` тохируулгуудад ашиглагдаж байна. Хэрэв бүрхүүлийн мөрийг орхисон бол `cron` анхдагч болох `sh`-ийг ашигладаг. Хэрэв `PATH` хувьсагчийг орхисон бол ямар ч анхдагч ашиглагдахгүй бөгөөд файлын байрлалууд абсолют байх хэрэгтэй. Хэрэв `HOME` мөрийг орхисон бол `cron` ажиллуулж байгаа хэрэглэгчийн гэрийн санг ашигладаг.

③ Энэ мөр нь нийт долоон талбарыг тодорхойлдог. Энд жагсаагдсан утгууд нь `minute`, `hour`, `mday`, `month`, `wday`, `who`, болон `command` юм. Эдгээрийг нэрээс нь харахад ойлгомжтой. `minute` нь тушаал ажиллах минутаар илэрхийлэгдсэн хугацаа. `hour` нь `minute`-ын адил тохируулга бөгөөд цагаар илэрхийлэгддэг. `mday` нь сарын өдрийг заана. `month` нь `hour` болон `minute`-тай адил бөгөөд сарыг зааж өгнө. `wday` тохируулга нь долоо хоногийн өдрийг заана. Эдгээр бүх талбарууд нь тоон утга байх ёстой бөгөөд хорин дөрвөн цагийг дагадаг. `who` талбар нь тусгай бөгөөд зөвхөн /etc/crontab файлд байдаг. Энэ талбар нь аль хэрэглэгчийн эрхээр тушаал ажиллахыг заадаг. Сүүлийн талбар нь ажиллуулах тушаалд зориулагдсан байна.

④ Энэ сүүлийн мөр нь дээр дурдсан утгуудыг тодорхойлдог. Энд бид хэд хэдэн `тэмдэгтүүд дараалсан/5` гэсэн жагсаалт байгааг анзаарах хэрэгтэй. Эдгээр `*` тэмдэгтүүд нь "эхний-эцсийн" гэсэн үг бөгөөд *үргэлж* гэж ойлгогдож болно. Тэгвэл энэ мөрөөс үзэхэд `atrun` тушаал нь `root` эрхээр 5 минут тутам аль өдөр сар байгаагаас үл хамааран ажиллана. `atrun` тушаалын талаар дэлгэрэнгүй мэдээллийг atrun(8) гарын авлагаас үзнэ үү.Тушаалууд тэдгээрт өгч болох дурын тооны тугуудтай байж болно; гэхдээ олон мөр болон уртассан тушаалууд урагшаа ташуу "\" үргэлжлүүлэх тэмдэгтээр хуваагдсан байх ёстой.

Энэ нь crontab файл болгоны хувьд үндсэн тохиргоо байна, гэхдээ нэг зүйл нь үүнээс өөр байна. Хэрэглэгчийг заадаг зургаа дахь талбар нь зөвхөн системийн /etc/crontab файлд байна. Энэ талбарыг хэрэглэгчийн crontab файлуудын хувьд орхих хэрэгтэй.

### 12.6.1. Crontab суулгах нь

Та энд тайлбарласан процедурыг ашиглан системийн crontab /etc/crontab -ийг засаж болон суулгах хэрэггүй. Зүгээр л өөрийн дуртай засварлагчийг ашигла: `cron` хэрэгсэл файл өөрчлөгдсөнийг мэдээд тэр даруй шинэчлэгдсэн хувилбарыг ашиглаж эхэлнэ. Дэлгэрэнгүй мэдээллийг Энэ БХА-ын оруулгаас үзнэ үү.

Хэрэглэгчийн бичсэн шинэ crontab файлыг суулгахын тулд эхлээд өөрийн дуртай засварлагчийг ашиглаад зөв хэлбэршилттэй файл үүсгээд дараа нь `crontab` хэрэгслийг ашигла. Хамгийн их ашиглагддаг тушаал бол:

```
% crontab crontab-file
```

Энэ жишээн дээрх crontab-file нь урд нь үүсгэгдсэн crontab-ийн файлын нэр юм.

Суулгасан crontab файлуудыг үзүүлдэг тохируулга бас байдаг: `-l` тохируулгыг `crontab` уруу өгч ажиллуулаад гарах үр дүнг хараарай.

Өөрийн crontab файлыг загвар ашиглалгүйгээр эхнээс нь эхлүүлэхийг хүссэн хэрэглэгчдэд зориулсан `crontab -e` тохируулга байдаг. Энэ нь сонгосон засварлагчийг хоосон файлтай ажиллуулдаг. Файл хадгалагдсаны дараа автоматаар `crontab` тушаалаар суулгагддаг.

Хэрэглэгчийн crontab-ийг бүр мөсөн устгахыг хүсвэл `crontab`-ийг `-r` тохируулгатай ашиглаарай.

## 12.7. FreeBSD дээр rc(8) ашиглах нь

2002 онд FreeBSD системийг эхлүүлэхэд зориулж NetBSD-ийн rc.d системийг оруулсан. Хэрэглэгчид /etc/rc.d сан доторх файлуудыг анзаарах хэрэгтэй. Эдгээр файлуудын ихэнх нь `start`, `stop`, болон `restart` тохируулгуудаар хянагддаг үндсэн үйлчилгээнүүд байдаг. Жишээ нь sshd(8) нь дараах тушаалаар дахин эхлэж болно:

```
# /etc/rc.d/sshd restart
```

Энэ процедур нь бусад үйлчилгээнүүдийн адил юм. Мэдээж үйлчилгээнүүд ихэнхдээ автоматаар rc.conf(5)-д зааснаар ачаалах үед эхэлдэг. Жишээ нь Сүлжээний Хаяг Хөрвүүлэх дэмонг эхлэх үед ажиллуулахаар нээх нь амархан бөгөөд /etc/rc.conf-д дараах мөрийг нэмдэг:

```
natd_enable="YES"
```

Хэрэв `natd_enable="NO"` мөр аль хэдийн байвал `NO`-ийг `YES` болгож өөрчлөөрэй. rc скриптүүд өөр бусад хамааралтай үйлчилгээнүүдийг дараагийн дахин ачаалалтын үеэр доор тайлбарласны дагуу автоматаар ачаалдаг.

rc.d систем нь үндсэндээ системийн эхлэх/унтрах үеэр үйлчилгээнүүдийг эхлүүлэх/зогсоох зорилготой бөгөөд стандарт `start`, `stop` болон `restart` тохируулгууд нь зөвхөн /etc/rc.conf-ийн харгалзах хувьсагчууд заагдсан үед өөрийн үйлдлийг гүйцэтгэдэг. Жишээ нь дээр дурдсан `sshd restart` тушаал нь /etc/rc.conf-д `sshd_enable` хувьсагч `YES` гэсэн тохиолдолд зөвхөн ажиллана. /etc/rc.conf-д байгаа тохируулгаас үл хамааран үйлчилгээг `start`, `stop` эсвэл `restart` хийхийн тулд тушаалууд "one" угтвартай байх шаардлагатай. Жишээ нь `sshd`-г /etc/rc.conf дахь тохиргооноос үл хамааран дахин эхлүүлэхдээ дараах тушаалыг ашиглана:

```
# /etc/rc.d/sshd onerestart
```

Тохирох rc.d скриптийг `rcvar` тохируулгатай ажиллуулж /etc/rc.conf-д үйлчилгээ нээгдсэн эсэхийг амархан шалгадаг. Тиймээс администратор `sshd`-г /etc/rc.conf-д нээгдсэн эсэхийг

дараах тушаалыг ажиллуулж шалгаж болно:

```
# /etc/rc.d/sshd rcvar
# sshd
$sshd_enable=YES
```

Хоёр дахь мөр (`# sshd`) нь `root` консолынх биш `sshd` тушаалын гаргасан үр дүн юм.

Үйлчилгээг ажиллах байгаа эсэхийг шалгах `status` тохируулга байдаг. Жишээ нь `sshd` эхэлсэн эсэхийг шалгахдаа:

```
# /etc/rc.d/sshd status
sshd is running as pid 433.
```

Зарим тохиолдолд үйлчилгээг `reload` хийх бас боломжтой байдаг. Энэ нь үйлчилгээг өөрийн тохиргооны файлуудыг дахин уншихыг зааж үйлчилгээ уруу дохио шидэхийг оролддог. Ихэнх тохиолдолд энэ нь үйлчилгээ уруу `SIGHUP` дохио шиднэ гэсэн үг юм. Үйлчилгээ болгонд энэ боломжийн дэмжлэг байдаггүй.

rc.d систем нь зөвхөн сүлжээний үйлчилгээнд ашиглагдаад зогсохгүй мөн системийн эхлүүлэлтэд бас ихээхэн хувь нэмэр оруулдаг. Жишээ нь bgfsck файлыг авч үзье. Энэ скрипт ажиллахад дараах мэдээллийг хэвлэж гаргана:

```
Starting background file system checks in 60 seconds.
```

Тиймээс энэ файлыг зөвхөн системийг эхлүүлэх үед файлын системийн арын шалгалтыг хийхэд хэрэглэдэг.

Системийн олон үйлчилгээнүүд зөв ажиллахын тулд бусад үйлчилгээнүүдээс хамаардаг. Жишээ нь NIS болон бусад RPC дээр тулгуурласан үйлчилгээнүүд `rpcbind` (portmapper) үйлчилгээ ажиллахаас нааш амжилттай ажилладаггүй. Үүнийг шийдэхийн тулд хамаарлуудын тухай болон бусад мета-өгөгдлийн тухай мэдээллийг эхлүүлэх скрипт бүрийн дээд хэсэгт тайлбараар оруулсан байдаг. rcorder(8) програм хамаарлуудыг хангаж системийн үйлчилгээнүүдийг ямар дарааллаар ажиллуулах ёстойг тогтоохын тулд эдгээр тайлбаруудыг уншдаг.

Дараах үгнүүдийг бүх эхлүүлэх скриптэд оруулах ёстой (Эдгээр нь эхлүүлэх скриптийг "идэвхжүүлэх"эд rc.subr(8)-д шаардлагатай байдаг):

- `PROVIDE`: Энэ файлын хангаж байгаа үйлчилгээнүүдийг заана.

Дараах үгнүүдийг эхлүүлэх скрипт бүрийн эхэнд оруулж болно. Эдгээр нь заавал шаардлагатай биш боловч rcorder(8)-д тус дөхөм болох ашигтай байдаг:

- `REQUIRE`: Энэ үйлчилгээнд шаардлагатай үйлчилгээнүүдийг жагсаана. Энэ файл заагдсан

үйлчилгээнүүдийн *дараа* ажиллана.

- BEFORE: Энэ үйлчилгээнээс хамааралтай үйлчилгээнүүдийг жагсаана. Энэ файл заагдсан үйлчилгээнүүдийн *өмнө* ажиллана.

Эдгээр түлхүүр үгнүүдийг эхлүүлэх скрипт болгонд болгоомжтойгоор тохируулж өгснөөр бусад зарим UNIX® үйлдлийн системүүд шиг "ажиллах түвшингүүдтэй (runlevels)" зууралдалгүйгээр скриптүүдийн эхлэх дарааллыг маш сайн хянах боломжийг администраторт бий болгох юм.

rc.d системийн талаар нэмэлт мэдээллийг rc(8) болон rc.subr(8) гарын авлагын хуудаснуудаас олж болно. Хэрэв та өөрийн rc.d скриптүүд бичих эсвэл байгаагаа сайжруулахыг сонирхож байгаа бол танд бас энэ нийтлэл хэрэгтэй байж болох юм.

# 12.8. Сүлжээний интерфэйс картууд суулгах нь

Өнөөдөр бид сүлжээний холболтгүй компьютерийн талаар бодох ч аргагүй болсон билээ. Сүлжээний картыг нэмж тохируулах нь FreeBSD-ийн дурын администраторын ердийн ажил болдог.

## 12.8.1. Тохирох драйверийг олох нь

Эхлэхээсээ өмнө та өөрт байгаа картынхаа загвар, түүнд ашигласан бичил схем болон PCI эсвэл ISA картын аль нь эсэхийг мэдэх шаардлагатай. FreeBSD өргөн төрлийн PCI болон ISA картуудыг дэмждэг. Таны карт таны ашиглах хувилбар дээр дэмжигдсэн эсэхийг Тоног Төхөөрөмжийн Нийцтэй Байдлын Жагсаалтаас шалгаарай.

Таны карт дэмжигдсэнийг мэдсэний дараа та өөрийн картанд тохирох драйвераа тодорхойлох хэрэгтэй. /usr/src/sys/conf/NOTES болон /usr/src/sys/arch/conf/NOTES нь сүлжээний интерфэйс драйверуудын жагсаалтыг дэмжигдсэн бичил схем/картуудын тухай зарим мэдээллийн хамтаар танд өгөх болно. Хэрэв та аль драйвер нь зөв эсэхэд эргэлзэж байгаа бол драйверийн гарын авлагын хуудсыг уншаарай. Гарын авлагын хуудас нь дэмжигдсэн тоног төхөөрөмж болон бүр учирч болзошгүй асуудлуудын тухай дэлгэрэнгүй мэдээллийг өгдөг.

Хэрэв та ердийн карттай бол ихэнхдээ драйверийг хичээнгүйлэн хайх шаардлагагүй юм. Ердийн сүлжээний картуудад зориулсан драйверууд нь GENERIC цөмд байдаг, тэгэхээр таны карт ачаалах явцад иймэрхүү харагдах ёстой:

```
dc0: <82c169 PNIC 10/100BaseTX> port 0xa000-0xa0ff mem 0xd3800000-0xd38
000ff irq 15 at device 11.0 on pci0
miibus0: <MII bus> on dc0
bmtphy0: <BCM5201 10/100baseTX PHY> PHY 1 on miibus0
bmtphy0:  10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
dc0: Ethernet address: 00:a0:cc:da:da:da
dc0: [ITHREAD]
dc1: <82c169 PNIC 10/100BaseTX> port 0x9800-0x98ff mem 0xd3000000-0xd30
000ff irq 11 at device 12.0 on pci0
miibus1: <MII bus> on dc1
```

```
bmtphy1: <BCM5201 10/100baseTX PHY> PHY 1 on miibus1
bmtphy1:  10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
dc1: Ethernet address: 00:a0:cc:da:da:db
dc1: [ITHREAD]
```

Энэ жишээн дээр систем дээр байгаа хоёр карт dc(4) драйверийг ашиглаж байгааг бид харж байна.

Хэрэв таны NIC-д (Network Interface Card буюу Сүлжээний Интерфэйс Карт) зориулсан драйвер GENERIC-д байхгүй бол та өөрийн NIC-г ашиглахын тулд тохирох драйверийг ачаалах хэрэгтэй. Ингэхийн тулд хоёр аргын аль нэгийг ашиглана:

- Хамгийн амархан арга нь ердөө л өөрийн сүлжээний картанд зориулсан цөмийн модулийг kldload(8) ашиглан эсвэл тохирох мөрийг /boot/loader.conf-д нэмж ачаалах үед автоматаар ачаалах юм. Бүх NIC драйверууд модуль хэлбэрээр байдаггүй; модулиуд нь байдаггүй төхөөрөмжүүдийн дурдаж болох жишээнүүд гэвэл ISA картууд юм.
- Өөр нэг арга нь та өөрийн картын дэмжлэгийг цөмд оруулан статикаар хөрвүүлж болох юм. Өөрийн цөмийн тохиргооны файлд юу нэмэх ёстойг мэдэхийн тулд /usr/src/sys/conf/NOTES, /usr/src/sys/arch/conf/NOTES болон драйверийн гарын авлагын хуудсыг шалгаарай. Цөмийг дахин хөрвүүлэх талаар дэлгэрэнгүй мэдээллийг FreeBSD цөмийг тохируулах нь-с үзнэ үү. Хэрэв таны картыг таны цөм (GENERIC) ачаалах явцад илрүүлсэн бол та шинэ цөм бүтээх шаардлагагүй.

#### 12.8.1.1. Windows®-ийн NDIS драйверуудыг ашиглах нь

Харамсалтай нь өөрийн драйверуудад зориулсан схемүүдийг нээлттэй эхийн хүрээнийхэнд өгдөггүй, тийм мэдээллийг худалдааны нууц гэж үздэг олон үйлдвэрлэгчид байсаар байна. Ингэснээр FreeBSD болон өөр үйлдлийн системүүдийн хөгжүүлэгчдэд хоёр сонголт үлдсэн: буцаах инженерчлэлийн хүнд хэцүү, урт хугацааны процессийг туулж драйверуудыг хөгжүүлэх эсвэл Microsoft® Windows® тавцангуудад байдаг хоёртын хэлбэрийн драйверуудыг ашиглах арга замууд юм. FreeBSD-тэй холбогдсон зэрэг ихэнх хөгжүүлэгчид сүүлийн хандлагыг авч ашигладаг.

Билл Полын (wpaul) оруулсан хувь нэмрийн ачаар Сүлжээний Драйверийн Интерфэйсийн Тодорхойлолтын (NDIS) "эх (native)" дэмжлэг ордог болсон. FreeBSD NDISulator (өөрөөр Чөтгөр Төсөл) Windows® хоёртын драйверийг аваад ерөнхийдөө түүнийг Windows® дээр ажиллаж байгаа мэтээр хуурдаг. ndis(4) драйвер нь Windows® хоёртын файл ашиглаж байгаа учраас энэ нь зөвхөн i386™ болон amd64 системүүд дээр ажилладаг. PCI, CardBus, PCMCIA (PC-Card), болон USB төхөөрөмжүүдийг дэмждэг.

NDISulator ашиглахын тулд 3 зүйл хэрэгтэй:

1. Цөмийн эхүүд
2. Windows® XP драйверийн хоёртын файл (.SYS өргөтгөл)
3. Windows® XP драйверийн тохиргооны файл (.INF өргөтгөл)

Та өөрийн картад зориулсан файлуудыг олоорой. Ерөнхийдөө тэдгээрийг хавсаргасан CD-үүд эсвэл үйлдвэрлэгчүүдийн вэб хуудаснаас олж болно. Дараах жишээнүүдэд бид

W32DRIVER.SYS болон W32DRIVER.INF файлуудыг ашиглах болно.

Драйверын битийн урт FreeBSD-ийн хувилбарынхтай таарсан байх ёстой. FreeBSD/i386-н хувьд Windows® 32-бит драйвер ашиглана. FreeBSD/amd64-н хувьд Windows® 64-бит драйвер хэрэгтэй.

Дараагийн алхамд драйверийн хоёртын файлыг цөмийн ачаалж болох модуль болгон хөрвүүлнэ. `root` эрхээр ndisgen(8)-г хэрэглэнэ:

```
# ndisgen /path/to/W32DRIVER.INF /path/to/W32DRIVER.SYS
```

ndisgen(8) хэрэгсэл нь интерактив бөгөөд шаардлагатай нэмэлт мэдээллийг асуудаг. Одоо байгаа санд цөмийн шинэ модуль үүсгэнэ. kldload(8) ашиглан шинэ модулийг ачаална:

```
# kldload ./W32DRIVER_SYS.ko
```

Үүсгэгдсэн цөмийн модулиас гадна та ndis.ko болон if_ndis.ko модулиудыг ачаалах хэрэгтэй. Энэ нь таныг ndis(4)-ээс хамаарсан дурын модулийг ачаалах үед автоматаар хийгдэх ёстой. Хэрэв та тэдгээрийг гараар ачаалахыг хүсвэл дараах тушаалыг ашиглаарай:

```
# kldload ndis
# kldload if_ndis
```

Эхний тушаал нь NDIS минипорт драйвер дуттуйлагчийг ачаалах бөгөөд хоёр дахь нь яг сүлжээний интерфэйсийг ачаална.

Одоо dmesg(8)-ийг шалгаж ачаалахад алдаа байгаа эсэхийг үзэх хэрэгтэй. Бүгд сайн болж өнгөрсөн бол та дараах үр дүнг харах ёстой:

```
ndis0: <Wireless-G PCI Adapter> mem 0xf4100000-0xf4101fff irq 3 at device 8.0 on pci1
ndis0: NDIS API version: 5.0
ndis0: Ethernet address: 0a:b1:2c:d3:4e:f5
ndis0: 11b rates: 1Mbps 2Mbps 5.5Mbps 11Mbps
ndis0: 11g rates: 6Mbps 9Mbps 12Mbps 18Mbps 36Mbps 48Mbps 54Mbps
```

Эндээс эхлээд та ndis0 төхөөрөмжид өөр бусад сүлжээний интерфэйсийн (өөрөөр хэлбэл dc0) нэгэн адилаар хандах боломжтой болох юм.

Та бусад модулиудтай адилаар NDIS модулиудыг ачаалах явцад ачаалахаар системийг тохируулж болно. Эхлээд үүсгэгдсэн модуль W32DRIVER_SYS.ko-г /boot/modules уруу хуулах хэрэгтэй. Тэгээд дараах мөрийг /boot/loader.conf-д нэмнэ:

```
W32DRIVER_SYS_load="YES"
```

### 12.8.2. Сүлжээний карт тохируулах нь

Сүлжээний картанд зориулсан зөв драйвер ачаалагдсаны дараа картыг тохируулах шаардлагатай. Бусад олон зүйлсийн адил сүлжээний карт нь sysinstall програмаар суулгах явцад тохируулагдаж болно.

Таны системийн сүлжээний интерфэйсүүдэд зориулсан тохиргоог харуулахын тулд дараах тушаалыг ажиллуулна:

```
% ifconfig
dc0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80008<VLAN_MTU,LINKSTATE>
        ether 00:a0:cc:da:da:da
        inet 192.168.1.3 netmask 0xffffff00 broadcast 192.168.1.255
        media: Ethernet autoselect (100baseTX <full-duplex>)
        status: active
dc1: flags=8802<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80008<VLAN_MTU,LINKSTATE>
        ether 00:a0:cc:da:da:db
        inet 10.0.0.1 netmask 0xffffff00 broadcast 10.0.0.255
        media: Ethernet 10baseT/UTP
        status: no carrier
plip0: flags=8810<POINTOPOINT,SIMPLEX,MULTICAST> metric 0 mtu 1500
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
        options=3<RXCSUM,TXCSUM>
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x4
        inet6 ::1 prefixlen 128
        inet 127.0.0.1 netmask 0xff000000
        nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
```

Энэ жишээн дээр дараах төхөөрөмжүүдийг харуулсан:

- dc0: Эхний Ethernet интерфэйс
- dc1: Хоёрдугаар Ethernet интерфэйс
- plip0: Параллел порт интерфэйс (хэрэв параллел порт машин дээр байгаа бол)
- lo0: Буцаж эргэх төхөөрөмж

FreeBSD нь драйверийн нэр дээр цөмийн ачаалах явцад картууд ямар дарааллаар илрүүлэгдсэн тэр дарааллын тоог нэмж сүлжээний картыг нэрлэдэг. Жишээ нь sis2 нь систем дээрх sis(4) драйвер ашиглаж байгаа 3 дахь сүлжээний карт байж болох юм.

Энэ жишээн дээр dc0 төхөөрөмж босон ажиллаж байна. Түлхүүр индикаторууд нь:

1. `UP` нь картын тохиргоо хийгдэж бэлэн болсныг илэрхийлнэ.
2. Карт нь Интернэт (`inet`) хаягтай (энэ тохиолдолд `192.168.1.3`).
3. Энэ нь зөв дэд сүлжээний багтай (`netmask`; `0xffffff00` нь `255.255.255.0` адил).
4. Энэ нь зөв нийтэд цацах хаягтай (энэ тохиолдолд `192.168.1.255`).

5. Картны MAC (`ether`) хаяг нь `00:a0:cc:da:da:da` байна.
6. Физик зөөгчийн сонголт нь автомат сонголтын горим дээр байна (`media: Ethernet autoselect (100baseTX <full-duplex>)`). dc1 нь `10baseT/UTP` зөөгчтэй ажиллахаар тохируулагдсан байгааг бид харж болно. Байж болох зөөгчийн төрлүүдийн тухай дэлгэрэнгүй мэдээллийн талаар өөрийнх нь гарын авлагын хуудсанд хандаж үзнэ үү.
7. Холболтын (`status`) төлөв нь `active` буюу идэвхтэй байна, өөрөөр хэлбэл дамжуулагч илэрсэн байна. dc1-ийн хувьд бид `status: no carrier` буюу дамжуулагч байхгүйг харж болно. Энэ нь Ethernet кабель картанд залгагдаагүй байх үед хэвийн байна.

Хэрэв ifconfig(8)-ийн үр дүн дараах маягтай төстэй байвал:

```
dc0: flags=8843<BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80008<VLAN_MTU,LINKSTATE>
        ether 00:a0:cc:da:da:da
        media: Ethernet autoselect (100baseTX <full-duplex>)
        status: active
```

Энэ нь карт тохируулагдаагүйг илэрхийлнэ.

Картаа тохируулахын тулд танд `root` зөвшөөрлүүд хэрэгтэй. Сүлжээний картын тохируулгууд тушаалын мөрөөс ifconfig(8)-р хийгдэх боломжтой, гэхдээ та системийг дахин ачаалсан болгоныхоо дараа үүнийг хийх хэрэгтэй болно. /etc/rc.conf файл нь сүлжээний картын тохиргоог нэмэх газар юм.

/etc/rc.conf-ийг өөрийн дуртай засварлагч дээр нээгээрэй. Систем дээрх сүлжээний карт бүрийн хувьд мөр нэмэх хэрэгтэй, манай жишээн дээр бид эдгээр мөрүүдийг нэмсэн:

```
ifconfig_dc0="inet 192.168.1.3 netmask 255.255.255.0"
ifconfig_dc1="inet 10.0.0.1 netmask 255.255.255.0 media 10baseT/UTP"
```

Та dc0, dc1 болон бусдуудыг өөрийн картуудад зориулсан төхөөрөмжөөр өөрчлөх болон хаягуудыг зөвөөр солих хэрэгтэй. Зөвшөөрөгдсөн тохируулгуудын талаар дэлгэрэнгүйг картын драйвер болон ifconfig(8)-ийн гарын авлагын хуудаснуудаас, бас rc.conf(5) гарын авлагын хуудаснаас /etc/rc.conf-ийн синтаксын тухай дэлгэрэнгүй мэдээллийг унших хэрэгтэй.

Хэрэв та суулгах явцад сүлжээг тохируулсан бол сүлжээний карт(ууд)ын талаар зарим мөрүүд аль хэдийн байж болох юм. Мөрүүд нэмэхээсээ өмнө /etc/rc.conf-ийг дахин шалгаарай.

Мөн та LAN дахь төрөл бүрийн машинуудын нэрүүд болон IP хаягууд /etc/hosts файлд байхгүй бол тэдгээрийг нэмж засварлах шаардлагатай. Дэлгэрэнгүй мэдээллийн талаар hosts(5) болон /usr/shared/examples/etc/hosts файлд хандана уу.

Хэрэв энэ машинаар Интернэтэд холболт хийхээр төлөвлөсөн бол та гараараа анхдагч гарц болон нэрийн серверийг бас тохируулж өгөх ёстой:

```
# echo 'defaultrouter="your_default_router"' >> /etc/rc.conf
# echo 'nameserver your_DNS_server' >> /etc/resolv.conf
```

### 12.8.3. Тест хийх болон алдааг олж засварлах нь

/etc/rc.conf-д хэрэгцээтэй өөрчлөлтүүдийг хийснийхээ дараа та системээ дахин ачаалах шаардлагатай. Ингэснээр интерфэйс(үүд)эд хийгдэх өөрчлөлт(үүд)ийг зөвшөөрөх бөгөөд ямар нэг тохиргооны алдаагүйгээр систем ачаалж байгаа эсэхийг шалгадаг. Мөн өөрөөр та сүлжээний системээ дахин дуудаж болно:

```
# /etc/rc.d/netif restart
```

Хэрэв анхдагч гарцыг /etc/rc.conf файлд зааж өгсөн бол энэ тушаалыг ашиглана:

```
# /etc/rc.d/routing restart
```

Сүлжээний систем дахин дуудагдсаны дараа та сүлжээний интерфэйсүүдээ тест хийх хэрэгтэй.

#### 12.8.3.1. Ethernet карт тест хийх нь

Ethernet карт зөв тохируулагдсаныг шалгахдаа та 2 зүйлийг оролдох хэрэгтэй. Эхлээд интерфэйс уруу өөр уруу нь ping хийгээд дараа нь LAN дахь өөр машин уруу ping хийх хэрэгтэй.

Эхлээд локал интерфэйсийг тест хийнэ:

```
% ping -c5 192.168.1.3
PING 192.168.1.3 (192.168.1.3): 56 data bytes
64 bytes from 192.168.1.3: icmp_seq=0 ttl=64 time=0.082 ms
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=0.074 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=0.076 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=0.108 ms
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=0.076 ms

--- 192.168.1.3 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.074/0.083/0.108/0.013 ms
```

Одоо бид LAN дахь өөр машин уруу ping хийх хэрэгтэй:

```
% ping -c5 192.168.1.2
PING 192.168.1.2 (192.168.1.2): 56 data bytes
```

```
64 bytes from 192.168.1.2: icmp_seq=0 ttl=64 time=0.726 ms
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.766 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.700 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.747 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.704 ms

--- 192.168.1.2 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.700/0.729/0.766/0.025 ms
```

Хэрэв та /etc/hosts файлыг тохируулсан бол `192.168.1.2`-ийн оронд машины нэрийг бас ашиглаж болох болох юм.

#### 12.8.3.2. Алдааг олж засварлах нь

Тоног төхөөрөмж болон програм хангамжийн тохиргоонуудын алдааг олж засварлах нь үргэлж зовлон байдаг бөгөөд зовлонг энгийн зүйлүүдийг эхлээд шалгаснаар багасгах боломжтой. Таны сүлжээний кабель холбогдсон уу? Сүлжээний үйлчилгээнүүдээ зөв тохируулсан уу? Галт ханаа зөв тохируулсан уу? Таны хэрэглэж байгаа картыг FreeBSD дэмждэг үү? Алдааны тайланг явуулахаасаа өмнө тоног төхөөрөмжийн тэмдэглэлийг заавал шалгах хэрэгтэй. Өөрийн FreeBSD-ийн хувилбарыг хамгийн сүүлийн STABLE хувилбар уруу шинэчлээрэй. Захидлын жагсаалтын архивууд шалгах буюу эсвэл Интернетээс хайгаарай.

Хэрэв карт ажилласан мөртлөө ажиллагаа муу бол tuning(7) гарын авлагын хуудсыг унших нь зүйтэй юм. Мөн буруу сүлжээний тохиргоонууд удаан холболтын шалтгаан болдог учир та сүлжээний тохиргоог бас шалгаж болох юм.

Зарим хэрэглэгчид ганц хоёр `device timeout` мэдээлэлтэй тулгарч болох бөгөөд энэ нь зарим картуудын хувьд хэвийн юм. Хэрэв энэ нь үргэлжлээд эсвэл шаналгаатай болоод эхэлбэл уг төхөөрөмж өөр бусад төхөөрөмжтэй зөрчилдөж байгаа эсэхийг та магадгүй шалгахыг хүсэх байх. Кабелийн холболтуудыг дахин шалгаарай. Магадгүй танд өөр нэг карт хэрэгтэй байж болох юм.

Хэрэглэгчид зарим үед цөөн `watchdog timeout` гэсэн алдаанууд хардаг. Ийм үед эхлээд хийх юм нь сүлжээний кабелийг шалгана. Олон картууд Bus Mastering дэмждэг PCI оролтыг шаарддаг. Зарим нэг эх хавтангуудад үүнийг зөвхөн нэг PCI оролт зөвшөөрдөг (ихэнхдээ 0-р оролт). Энэ нь асуудал байж болох эсэхийг сүлжээний карт болон эх хавтангийн баримтаас шалгаарай.

Систем пакетийг зорьсон газар нь чиглүүлж чадахгүй тохиолдолд `No route to host` мэдээллүүд гардаг. Энэ нь анхдагч чиглүүлэлт заагаагүй тохиолдолд эсвэл кабель салгагдсан бол гардаг. `netstat -rn` тушаалын үр дүнг үзээд таны хүрэхийг оролдож байгаа тэр хост уруу чинь зөв чиглүүлэлт байгаа эсэхийг шалгаарай. Хэрэв байхгүй бол Сүлжээний нэмэлт ойлголтууд-г уншаарай.

`ping: sendto: Permission denied` алдааны мэдээллүүд нь буруу тохируулсан галт ханаас ихэвчлэн болдог. Хэрэв `ipfw` нь цөмд идэвхжсэн бөгөөд ямар ч дүрэм тодорхойлогдоогүй бол анхдагч бодлого нь бүх трафикийг бүр ping хүсэлтийг хүртэл татгалзан хаадаг! Дэлгэрэнгүйг Галт хана-с уншина уу.

Заримдаа картын ажиллагаа муу эсвэл дунджаас доогуур байдаг. Эдгээр тохиолдолд зөөгч конголтын горимыг `autoselect` горимоос зөв зөөгчийн конголт уруу болгож тааруулах нь шилдэг арга юм. Энэ нь ихэнх тоног төхөөрөмжийн хувьд ихэвчлэн ажиллах боловч хүн болгоны хувьд байгаа ийм асуудлыг шийдэхгүй ч байж болох юм. Дахин хэлэхэд бүх сүлжээний тохиргоонуудыг шалгаж tuning(7) гарын авлагын хуудсыг уншаарай.

## 12.9. Виртуал Хостууд

FreeBSD-ийн хамгийн түгээмэл хэрэглээ бол нэг сервер сүлжээн дээр олон сервер мэтээр ажиллах виртуал сайт хост хийх боломж юм. Үүнийг нэг интерфэйс дээр олон сүлжээний хаягууд тавьж хийдэг.

Өгөгдсөн сүлжээний интерфэйс нь нэг "жинхэнэ" хаягтай бөгөөд дурын тооны "өөр(alias)" хаягуудтай байж болох юм. Эдгээр өөр хаягуудыг ихэнхдээ /etc/rc.conf-д тохирох хаягийн оруулгуудыг оруулан нэмж өгдөг.

fxp0 интерфэйсд зориулсан өөр хаягийн оруулга нь иймэрхүү байна:

```
ifconfig_fxp0_alias0="inet xxx.xxx.xxx.xxx netmask xxx.xxx.xxx.xxx"
```

Өөр хаягийн оруулгууд нь `alias0` гэж эхлэх ёстой бөгөөд дээш өгсөх дарааллаар явдаг (жишээ нь `_alias1`, `_alias2`, гэх мэт). Тохиргооны үйл явц эхний байхгүй дугаар дээр хүрч зогсдог.

Өөр хаягийн сүлжээний багуудыг тооцоолох нь чухал байдаг, гэхдээ азаар энэ нь маш амархан. Өгөгдсөн интерфэйсийн хувьд сүлжээний багийг зөвөөр үзүүлдэг нэг хаяг байх ёстой. Энэ сүлжээн дэх өөр бусад хаягууд бүгд `1`-ээс (энэ нь `255.255.255.255` гэх буюу эсвэл `0xffffffff` гэж илэрхийлэгддэг) тогтсон сүлжээний багтай байх ёстой.

Жишээ нь fxp0 интерфэйс нь `10.1.1.0` сүлжээнд `255.255.255.0` болон `202.0.75.16` сүлжээнд `255.255.255.240` багуудыг ашиглаж хоёр сүлжээнд холбогдсон гэж бодъё. Бид системийг `10.1.1.1`-ээс `10.1.1.5` хүртэл болон `202.0.75.17`-ээс эхлээд `202.0.75.20` хүртэлх хаягууд дээр байлгахыг хүсэж байна. Дээр тэмдэглэсний дагуу өгөгдсөн сүлжээний хүрээн дэх зөвхөн эхний хаяг (энэ тохиолдолд `10.0.1.1` болон `202.0.75.17`) жинхэнэ сүлжээний багтай байх ёстой; бусад үлдсэн бүгд (`10.1.1.2`-ээс `10.1.1.5` хүртэл болон `202.0.75.18`-ээс эхлээд `202.0.75.20` хүртэл) `255.255.255.255` сүлжээний багтай байхаар тохируулагдах хэрэгтэй.

Дараах /etc/rc.conf оруулгууд нь энэ зорилгоор адаптерийг зөв тохируулж байна:

```
ifconfig_fxp0="inet 10.1.1.1 netmask 255.255.255.0"
ifconfig_fxp0_alias0="inet 10.1.1.2 netmask 255.255.255.255"
ifconfig_fxp0_alias1="inet 10.1.1.3 netmask 255.255.255.255"
ifconfig_fxp0_alias2="inet 10.1.1.4 netmask 255.255.255.255"
ifconfig_fxp0_alias3="inet 10.1.1.5 netmask 255.255.255.255"
ifconfig_fxp0_alias4="inet 202.0.75.17 netmask 255.255.255.240"
ifconfig_fxp0_alias5="inet 202.0.75.18 netmask 255.255.255.255"
ifconfig_fxp0_alias6="inet 202.0.75.19 netmask 255.255.255.255"
ifconfig_fxp0_alias7="inet 202.0.75.20 netmask 255.255.255.255"
```

# 12.10. Системийн лог хийгч syslogd-г тохируулах нь

Систем лог хийх нь системийг удирдахад чухал зүйл юм. Үүнийг тоног төхөөрөмж болоод програм хангамжийн асуудлууд, мөн систем дэх алдаануудыг олж илрүүлэхэд хэрэглэдэг. Аюулгүй байдлын аудит хийх болон аливаа учралд хариу үзүүлэхэд бас маш чухал үүрэг гүйцэтгэдэг. Хяналтын терминалгүй системийн демонууд мэдээллийг системийн лог хийгч рүү эсвэл бусад лог файл руу ихэвчлэн бас лог хийдэг.

Энэ хэсэгт FreeBSD системийн лог хийгч syslogd(8)-г хэрхэн тохируулж ашиглах талаар болон логийг багасгах ба newsyslog(8) ашиглан лог удирдах талаар хэлэлцэх болно. Локал машин дээр `syslogd`-г тохируулж ашиглах талаар анхаарах болно. Тусдаа лог хост ашиглах талаарх нэмэлт тохиргооны тухай дэлгэрэнгүйг syslogd ашиглан алсын хост руу бүртгэх нь хэсгээс үзнэ үү.

## 12.10.1. syslogd ашиглах нь

FreeBSD-н syslogd(8)-н анхдагч тохиргоо ачаалах үед эхэлдэг. Үүнийг /etc/rc.conf дахь `syslogd_enable` хувьсагчаар хянадаг. syslogd(8)-н ажиллагаанд нөлөөлдөг програмын хэд хэдэн аргументууд байдаг. Тэдгээрийг өөрчлөхийн тулд /etc/rc.conf дахь `syslogd_flags`-г ашиглана. Аргументуудын талаар дэлгэрэнгүйг syslogd(8)-оос, rc.conf(5) ба Гол Тохиргоо болон FreeBSD дээр rc(8) ашиглах нь хэсгээс /etc/rc.conf ба rc(8) дэд системийн талаар дэлгэрэнгүйг үзнэ үү.

## 12.10.2. syslogd-г тохируулах нь

Тохиргооны файл нь анхдагчаар /etc/syslog.conf бөгөөд логуудыг хүлээж авсныхаа дараа хэрхэн яаж ажиллахыг хянадаг. Ирж байгаа үйл явдлуудтай ажиллахыг хянах хэд хэдэн параметрүүд байдаг бөгөөд тэдгээрээс хамгийн хялбар нь *facility* ба *level* юм. Хэрэгсэл нь цөм эсвэл демон гэх мэт аль дэд систем логийг үүсгэснийг тайлбарлах бөгөөд түвшин нь учирсан үйл явдлын хор хөнөөлийг тайлбарладаг. Энэ нь логийг өөр лог файлууд рүү өгөх эсвэл хаях зэргээр тохиргоо болон түвшингээс хамааруулан хийх боломжтой болгодог. Лог илгээсэн програм болон алсаас лог хийж байгаа тохиолдолд лог үйл явц үүсгэж байгаа машины хостын нэрээс хамаарч арга хэмжээ авах боломж бас байдаг.

syslogd(8)-г тохируулах нь хялбар байдаг. Тохиргооны файл нь хийх үйлдэл бүрийн хувьд нэг мөртэй байх бөгөөд мөр бүрийн синтакс нь сонголтын талбар болон арга хэмжээний талбараас тогтоно. Сонголтын талбарын синтакс нь *facility.level* байх бөгөөд *facility* буюу хэрэгслээс ирж байгаа логуудыг *level* түвшинд буюу түүнээс дээш түвшинд авах тохиргоо юм. Мөн нэмэлтээр юу лог хийхийг илүү нарийн зааж өгөхийн тулд харьцуулах флагийг түвшингийн өмнө нэмж өгөх бас боломжтой. Адил үйлдэлд олон сонголтын талбарыг ашиглаж болох бөгөөд тэдгээрийг цэг таслалаар (`;`) тусгаарладаг. `*`-г ашиглавал бүгдийг гэсэн утгатай. Арга хэмжээний талбар нь файл эсвэл алсын лог хост зэрэг хаашаа логийг илгээхийг зааж өгдөг. Жишээ нь энд FreeBSD-н анхдагч syslog.conf байна:

```
# $FreeBSD$
#
#       Spaces ARE valid field separators in this file. However,
#       other *nix-like systems still insist on using tabs as field
```

```
#       separators. If you are sharing this file between systems, you
#       may want to use only tabs as field separators here.
#       Consult the syslog.conf(5) manpage.
*.err;kern.warning;auth.notice;mail.crit                /dev/console ①
*.notice;authpriv.none;kern.debug;lpr.info;mail.crit;news.err   /var/log/messages
security.*                                              /var/log/security
auth.info;authpriv.info                                 /var/log/auth.log
mail.info                                               /var/log/maillog ②
lpr.info                                                /var/log/lpd-errs
ftp.info                                                /var/log/xferlog
cron.*                                                  /var/log/cron
*.=debug                                                /var/log/debug.log ③
*.emerg                                                 *
# uncomment this to log all writes to /dev/console to /var/log/console.log
#console.info                                           /var/log/console.log
# uncomment this to enable logging of all log messages to /var/log/all.log
# touch /var/log/all.log and chmod it to mode 600 before it will work
#*.*                                                    /var/log/all.log
# uncomment this to enable logging to a remote loghost named loghost
#*.*                                                    @loghost
# uncomment these if you're running inn
# news.crit                                             /var/log/news/news.crit
# news.err                                              /var/log/news/news.err
# news.notice                                           /var/log/news/news.notice
!ppp ④
*.*                                                     /var/log/ppp.log
!*
```

① `err` болон түүнээс дээш, мөн `kern.warning`, `auth.notice` ба `mail.crit` түвшний бүх мэдээллийг лог хийж эдгээр мэдээллийг консол (/dev/console) руу гаргах.

② `mail` хэрэгслийн `info` буюу түүнээс дээш түвшний бүх мэдээллийг барьж логийг /var/log/maillog руу авах.

③ Энэ мөр нь `=` буюу харьцуулах флагийг ашиглаж байгаа бөгөөд `debug` түвшний мэдээллийг авч /var/log/debug.log руу бичихийг заана.

④ Энд *програмыг хэрхэн заах* талаар жишээг харуулсан байна. Энэ нь програмыг зааж өгсөн тэр програмын хувьд ажиллах тийм дүрэм бий болгоно. Энэ тохиолдлын

хувьд энэ мөр болон түүний дараах нь зөвхөн `ppp`-с гарах бүх мэдээллийг /var/log/ppp.log файл руу авч байна.

Энэ жишээ нь олон түвшин болон дэд системүүд байгааг харуулж байна. Түвшингүүд нь хамгийн чухлаас бага руу жагсаагдсан байна: `emerg`, `alert`, `crit`, `err`, `warning`, `notice`, `info` ба `debug`.

Хэрэгслүүд нь ямар нэг дараалалгүйгээр дараах байна: `auth`, `authpriv`, `console`, `cron`, `daemon`, `ftp`, `kern`, `lpr`, `mail`, `mark`, `news`, `security`, `syslog`, `user`, `uucp` ба `local0`-с `local7` хүртэл байна. Өөр үйлдлийн системүүдийн хувьд өөр хэрэгслүүд байж болохыг анхааараарай.

Эдгээрийг мэдсэний дараа `notice` болон түүнээс дээш түвшинд янз бүрийн демонгоос гарч

байгаа бүгдийг /var/log/daemon.log руу лог хийх тохиргооны мөрийг /etc/syslog.conf руу нэмэх нь хялбар байх болно. Дараахийг нэмэхэд л болно:

```
daemon.notice                                   /var/log/daemon.log
```

Түвшингүүд болон хэрэгслүүдийн талаарх дэлгэрэнгүй мэдээллийг syslog(3) ба syslogd(8)-с үзнэ үү. syslog.conf болон түүний синтакс, илүү нарийн тохиргоо бүхий жишээнүүдийн талаар дэлгэрэнгүйг syslog.conf(5) ба syslogd ашиглан алсын хост руу бүртгэх нь-с үзнэ үү.

## 12.10.3. Лог удирдах ба newsyslog ашиглан багасгах

Лог файлууд нь хурдан томорч аажмаар нэмэгдэх нь элбэг байдаг. Энэ нь тийм ч чухал биш мэдээллээр файл болон хатуу дискийг дүүргэхэд хүргэдэг. Үүнийг арилгахын тулд логийн удирдлагыг ашигладаг. FreeBSD-д newsyslog(8) ашиглан лог файлуудыг удирддаг. Энэ програм нь тодорхой давтамжтайгаар лог файлуудын хэмжээг багасгаж архивлах болон байхгүй болсон лог файлуудыг үүсгэх, лог файлуудыг зөөх үед дохио өгөх зэрэгт ашиглагддаг. Лог файлууд нь заавал syslog-с гарсан байх шаардлагагүй байдаг. newsyslog(8) нь дурын програмаас гарсан дурын логтой ажиллаж чаддаг. `newsyslog`-г cron(8)-с ихэвчлэн ажиллуулдаг бөгөөд системийн демон биш гэдгийг санах хэрэгтэй. Анхдагч тохиргоогоор цаг бүр ажиллахаар тохируулагдсан байдаг.

### 12.10.3.1. newsyslog-г тохируулах

Ямар арга хэмжээ авахыг мэдэхийн тулд newsyslog(8) анхдагчаар /etc/newsyslog.conf тохиргооны файлыг уншдаг. Энэ тохиргооны файл нь newsyslog(8) удирддаг файл бүрийн хувьд нэг мөрийг агуулсан байдаг. Мөр бүр нь файлын эзэн, зөвшөөрлүүд, файлын хэмжээг хэзээ багасгаж арвивлах болон логийг багасгахад (шахалт гэх мэт) нөлөөлөх нэмэлт флагууд ба логийг хэзээ багасгахыг хэлэх програмуудыг заадаг. Жишээ нь энд FreeBSD дээрх анхдагч тохиргоо байна:

```
# configuration file for newsyslog
# $FreeBSD$
#
# Entries which do not specify the '/pid_file' field will cause the
# syslogd process to be signalled when that log file is rotated.  This
# action is only appropriate for log files which are written to by the
# syslogd process (ie, files listed in /etc/syslog.conf).  If there
# is no process which needs to be signalled when a given log file is
# rotated, then the entry for that file should include the 'N' flag.
#
# The 'flags' field is one or more of the letters: BCDGJNUXZ or a '-'.
#
# Note: some sites will want to select more restrictive protections than the
# defaults.  In particular, it may be desirable to switch many of the 644
# entries to 640 or 600.  For example, some sites will consider the
# contents of maillog, messages, and lpd-errs to be confidential.  In the
# future, these defaults may change to more conservative ones.
#
```

```
# logfilename          [owner:group]    mode count size when  flags [/pid_file]
[sig_num]
/var/log/all.log                        600  7     *    @T00  J
/var/log/amd.log                        644  7     100  *     J
/var/log/auth.log                       600  7     100  @0101T JC
/var/log/console.log                    600  5     100  *     J
/var/log/cron                           600  3     100  *     JC
/var/log/daily.log                      640  7     *    @T00  JN
/var/log/debug.log                      600  7     100  *     JC
/var/log/init.log                       644  3     100  *     J
/var/log/kerberos.log                   600  7     100  *     J
/var/log/lpd-errs                       644  7     100  *     JC
/var/log/maillog                        640  7     *    @T00  JC
/var/log/messages                       644  5     100  @0101T JC
/var/log/monthly.log                    640  12    *    $M1D0 JN
/var/log/pflog                          600  3     100  *     JB
/var/run/pflogd.pid
/var/log/ppp.log        root:network    640  3     100  *     JC
/var/log/security                       600  10    100  *     JC
/var/log/sendmail.st                    640  10    *    168   B
/var/log/utx.log                        644  3     *    @01T05 B
/var/log/weekly.log                     640  5     1    $W6D0 JN
/var/log/xferlog                        600  7     100  *     JC
```

Мөр бүр багасгах файлын нэрээс эхэлдэг бөгөөд үүний дараа багасгасан болон шинээр үүссэн файлуудын эзэн болон бүлэг нэмэлтээр байж болно. Дараагийн талбар `mode` нь файлуудын горим бөгөөд `count` нь багасгасан файл хэдийг үлдээхийг зааж өгдөг. `size` ба `when` талбарууд нь файлыг хэзээ багасгахыг `newsyslog`-д хэлж өгнө. Лог файлыг `size` талбарт зааснаас том болсон үед эсвэл `when` талбарт заасан хугацаа өнгөрсөн үед багасгадаг. `*` нь энэ талбарыг орхино гэсэн утгатай. *flags* талбар нь багасгасан файлыг хэрхэн шахах эсвэл байхгүй байгаа лог файлыг үүсгэх зэрэг зааврыудыг newsyslog(8)-д өгдөг. Хамгийн сүүлийн хоёр талбар нь нэмэлт бөгөөд процессын PID-file болон сигналын дугаарыг зааж файлыг багасгах үед тухайн процесс руу илгээх сигналыг зааж өгдөг. Бүх талбарууд, флагууд болон багасгах хугацааг хэрхэн зааж өгөх талаарх дэлгэрэнгүй мэдээллийг newsyslog.conf(5)-с үзнэ үү. `newsyslog` нь `cron`-с ажилладаг бөгөөд cron(8)-ы ажиллах давтамжаас илүү олон ажиллах боложмгүй гэдгийг санаарай.

# 12.11. Тохиргооны Файлууд

## 12.11.1. /etc-н бүтэц

Тохиргооны мэдээллийг хадгалдаг хэд хэдэн сангууд байдаг. Эдгээр нь:

| | |
|---|---|
| /etc | Системийн ерөнхий тохиргооны мэдээлэл; энд байгаа өгөгдөл нь системийн хувьд өөр өөр. |
| /etc/defaults | Системийн тохиргооны файлуудын анхдагч хувилбарууд. |

| | |
|---|---|
| /etc/mail | sendmail(8)-ийн нэмэлт тохиргоо, бусад MTA тохиргооны файлууд. |
| /etc/ppp | Хэрэглэгч- болон цөмийн-ppp програмуудад зориулсан тохиргоо. |
| /etc/namedb | named(8) өгөгдөлд зориулсан анхдагч байрлал. Ихэнхдээ named.conf болон бүсийн файлууд энд хадгалагддаг. |
| /usr/local/etc | Суулгагдсан програмуудад зориулсан тохиргооны файлууд. Програм болгоны дэд сангуудыг агуулж болно. |
| /usr/local/etc/rc.d | Суулгагдсан програмуудад зориулсан эхлүүлэх/зогсоох скриптүүд. |
| /var/db | Багцын өгөгдлийн бааз, байршил олох өгөгдлийн бааз, гэх зэрэг систем болгоны хувьд автоматаар үүсгэгдсэн өгөгдлийн баазын файлууд. |

## 12.11.2. Хостын нэрс

### 12.11.2.1. /etc/resolv.conf

/etc/resolv.conf нь FreeBSD-ийн тодорхойлогч Интернэт Домэйн Нэрийн Системд (DNS) хэрхэн хандахыг заадаг.

resolv.conf дахь хамгийн түгээмэл оруулгууд нь:

| | |
|---|---|
| `nameserver` | Тодорхойлогчийн асуух нэрийн серверийн IP хаяг. Серверүүд нь хамгийн ихдээ гурав байх жагсаасан дарааллаар асуугддаг. |
| `search` | Хостын нэрийн хайлтад зориулж жагсаалтаас хайх. Энэ нь ихэнхдээ локал хостын нэрийн домэйноор тодорхойлогддог. |
| `domain` | Локал домэйн нэр. |

Ердийн resolv.conf:

```
search example.com
nameserver 147.11.1.11
nameserver 147.11.100.30
```

`search` болон `domain` тохируулгуудын зөвхөн нэг нь хэрэглэгдэх ёстой.

Хэрэв та DHCP ашиглаж байгаа бол dhclient(8) нь DHCP серверээс хүлээн авсан мэдээллээр resolv.conf-г дарж бичдэг.

#### 12.11.2.2. /etc/hosts

/etc/hosts нь хуучин Интернэтийн үлдэгдэл энгийн текст өгөгдлийн бааз юм. Энэ нь DNS болон NIS-тэй цуг нэрийг IP хаяг уруу болгож тааруулах боломжийг ханган ажилладаг. LAN-аар холбогдсон локал компьютеруудыг амархан нэрлэх зориулалтаар named(8) сервер суулгаж тохируулахын оронд энд байрлуулж болдог. Мөн /etc/hosts нь түгээмэл ханддаг нэрсэд зориулагдсан гадагшаа хандах хүсэлтийг багасгаж Интернэтийн нэрсийн локал бичлэгийг хангадаг байж болно.

```
# $FreeBSD$
#
#
# Host Database
#
# This file should contain the addresses and aliases for local hosts that
# share this file.  Replace 'my.domain' below with the domainname of your
# machine.
#
# In the presence of the domain name service or NIS, this file may
# not be consulted at all; see /etc/nsswitch.conf for the resolution order.
#
#
::1         localhost localhost.my.domain
127.0.0.1       localhost localhost.my.domain
#
# Imaginary network.
#10.0.0.2       myname.my.domain myname
#10.0.0.3       myfriend.my.domain myfriend
#
# According to RFC 1918, you can use the following IP networks for
# private nets which will never be connected to the Internet:
#
#   10.0.0.0    -   10.255.255.255
#   172.16.0.0  -   172.31.255.255
#   192.168.0.0 -   192.168.255.255
#
# In case you want to be able to connect to the Internet, you need
# real official assigned numbers.  Do not try to invent your own network
# numbers but instead get one from your network provider (if any) or
# from your regional registry (ARIN, APNIC, LACNIC, RIPE NCC, or AfriNIC.)
#
```

/etc/hosts нь энгийн хэлбэрийг агуулдаг:

```
[Internet address] [official hostname] [alias1] [alias2] ...
```

Жишээ нь:

```
10.0.0.1 myRealHostname.example.com myRealHostname foobar1 foobar2
```

Дэлгэрэнгүй мэдээллийн талаар hosts(5) хуудаснаас зөвлөгөө авна уу.

### 12.11.3. sysctl.conf

sysctl.conf нь rc.conf-той бараг л адил харагддаг. Утгууд нь `хувьсагч=утга` хэлбэрээр заагддаг. Тодорхойлсон утгууд нь систем олон-хэрэглэгчийн горимд шилжсэний дараа тохируулагддаг. Энэ горимд бүх хувьсагчууд тохируулагдах боломжгүй.

Сүйрлийн дохионы гаралтуудын бүртгэлийг хааж бусад хэрэглэгчдийн эхлүүлсэн процессуудыг өөр хэрэглэгчдэд харуулахгүй байлгахын тулд дараах тохируулгуудыг sysctl.conf файлд тохируулж өгч болно:

```
# Do not log fatal signal exits (e.g., sig 11)
kern.logsigexit=0

# Prevent users from seeing information about processes that
# are being run under another UID.
security.bsd.see_other_uids=0
```

## 12.12. sysctl(8) ашиглан тааруулах нь

sysctl(8) нь ажиллаж байгаа FreeBSD системд өөрчлөлтүүдийг хийхийг танд зөвшөөрдөг интерфэйс юм. Энэ нь туршлагатай системийн администраторын хувьд ажиллагааг мэдэгдэхүйц сайжруулж чадах TCP/IP болон виртуал санах ойн системийн олон нарийн тохируулгуудыг агуулдаг. Таван зуу гаруй системийн хувьсагчуудыг sysctl(8) ашиглан унших болон тохируулж болдог.

sysctl(8) нь голдоо хоёр үүргийг гүйцэтгэдэг: системийн тохиргоонуудыг унших болон өөрчлөх.

Уншигдаж болох бүх хувьсагчуудыг харахдаа:

```
% sysctl -a
```

Тухайн хувьсагчийг уншихдаа, жишээ нь, `kern.maxproc`:

```
% sysctl kern.maxproc
kern.maxproc: 1044
```

Тухайн хувьсагчийг заахдаа хялбар *хувьсагч=утга* синтаксийг ашиглаарай:

```
# sysctl kern.maxfiles=5000
```

```
kern.maxfiles: 2088 -> 5000
```

sysctl хувьсагчуудын тохиргоонууд нь ихэвчлэн тэмдэгтүүд (strings), тоонууд эсвэл boolean (boolean `1` нь тийм эсвэл `0` нь үгүй байна) утгууд байна.

Хэрэв та машин ачаалах болгонд автоматаар зарим хувьсагчуудыг тохируулахыг хүсвэл /etc/sysctl.conf файлд тэдгээрийг нэмээрэй. Дэлгэрэнгүй мэдээллийн талаар sysctl.conf(5) гарын авлагын хуудас болон sysctl.conf-с үзнэ үү.

### 12.12.1. Зөвхөн-унших sysctl(8)

Зарим тохиолдолд зөвхөн-унших sysctl(8) утгуудыг өөрчлөх шаардлагатай байж болох юм. Энэ нь заримдаа хийхээс өөр аргагүй байдаг боловч зөвхөн (дахин) ачаалахад хийгдэх боломжтой.

Жишээ нь зарим зөөврийн компьютерийн загваруудад cardbus(4) төхөөрөмж нь санах ойн хүрээг шалгадаггүй бөгөөд доор дурдсантай төстэй алдаанууд гарган амжилтгүй болдог:

```
cbb0: Could not map register memory
device_probe_and_attach: cbb0 attach returned 12
```

Дээрх шиг тохиолдлууд нь ихэвчлэн зөвхөн уншихаар тохируулагдсан зарим анхдагч sysctl(8) тохиргоонуудыг өөрчлөхийг шаарддаг. Эдгээр нөхцөлүүдийг давж гарахын тулд хэрэглэгч sysctl(8) "OID"-уудыг тэдгээрийн /boot/loader.conf файлд хийж өгч болно. Анхдагч тохиргоонууд /boot/defaults/loader.conf файлд байрладаг.

Дээр дурдсан асуудлыг шийдэхийн тулд хэрэглэгч урьд нь дурдсан файлд `hw.pci.allow_unsupported_io_range=1` гэж тохируулах шаардлагатай. Ингэснээр cardbus(4) зөв ажиллах болно.

## 12.13. Дискнүүдийг тааруулах нь

### 12.13.1. Sysctl хувьсагчууд

#### 12.13.1.1. `vfs.vmiodirenable`

`vfs.vmiodirenable` sysctl хувьсагч нь 0 (идэвхгүй) эсвэл 1 (идэвхтэй) гэж тохируулагдаж болно; анхдагчаар 1 байна. Энэ хувьсагч нь систем сангуудыг хэрхэн кэш (шуурхай санамж) хийхийг хянадаг. Ихэнх сангууд зөвхөн ганц фрагментийг (ихэвчлэн 1 K) файлын системд болон түүнээс багыг буфер кэшд хэрэглэн жижиг хэмжээтэй байдаг. Энэ хувьсагчийг хааснаар (0 болгосноор) буфер кэш нь таныг асар их хэмжээний санах ойтой байсан ч гэсэн зөвхөн тодорхой тооны сангуудыг кэш хийдэг. Нээгдсэн (1 болгосон) үед энэ sysctl нь бүх санах ойг кэш хийхэд бэлэн болгож буфер кэшд VM Хуудасны Кэшийг хэрэглэн сангуудыг кэш хийх боломжийг олгодог. Гэхдээ сангуудыг кэш хийх хамгийн бага гол дахь санах ой нь 512 байт биш харин физик хуудасны хэмжээ (ихэвчлэн 4 K) байдаг. Хэрэв та их олон тооны файлуудтай ажилладаг үйлчилгээ ажиллуулж байгаа бол бид энэ тохируулгыг идэвхтэй байлгахыг зөвлөж байна. Тийм үйлчилгээнүүдэд вэб кэшүүд, том захидлын системүүд,

болон мэдээний системүүд орж болно. Энэ тохируулгыг идэвхтэй байлгах нь хайр гамгүй зарцуулсан санах ойтой байхад ч гэсэн ерөнхийдөө ажиллагааг удаашруулдаггүй, гэхдээ та түүнийг мэдэхийн тулд туршиж үзэж болно.

#### 12.13.1.2. vfs.write_behind

`vfs.write_behind` sysctl хувьсагчийн анхдагч утга нь `1` (идэвхтэй) байна. Энэ нь том дараалсан файлуудыг бичих үед ихэвчлэн гардаг бүх кластеруудыг цуглуулсан үед зөөгчийн бичилтүүдийг хийхийг файлын системд хэлж өгдөг. Санаа нь бол I/O ажиллагааны хувьд ашиггүй байхад бохир буферууд бүхий буферийн кэшийг замхруулахаас зайлсхийхэд оршдог. Гэхдээ энэ нь процессуудыг зогсоож магадгүй бөгөөд зарим нөхцөл байдалд та магадгүй үүнийг идэвхгүй болгохыг хүсэж болох юм.

#### 12.13.1.3. vfs.hirunningspace

`vfs.hirunningspace` sysctl хувьсагч өгөгдсөн дурын хоромд системийн хувьд бүхэлд нь хэдий хэмжээний хүлээгдэж байгаа бичих I/O-г дискний хянагчуудад өгөх дараалалд оруулж болохыг тодорхойлдог. Анхдагч утга нь ихэвчлэн хангалттай гэхдээ олон дисктэй машинууд дээр та үүнийг дөрөв эсвэл таван *мегабайт* хүртэл ихэсгэхийг хүсэж болох юм. Утгыг хэтэрхий өндөр тавих нь (буфер кэшийн бичих тогтоосон хэмжээг давах нь) туйлын муу кластерлах ажиллагаанд хүргэж болно. Энэ утгыг хэтэрхий өндөр бүү тавь! Өндөр бичих утгууд нь яг тэр үед хийгдэж байгаа уншилтуудад хоцрогдол нэмж магадгүй юм.

Бусад төрөл бүрийн буфер-кэш болон VM хуудасны кэштэй холбоотой sysctl-ууд байдаг. Бид эдгээр утгуудыг өөрчлөхийг зөвлөдөггүй, VM систем нь өөрийгөө автоматаар тааруулж туйлын сайн ажилладаг.

#### 12.13.1.4. vm.swap_idle_enabled

`vm.swap_idle_enabled` sysctl хувьсагч нь маш олон хэрэглэгчид таны системд орж гарч байдаг, сул зогссон олон процессуудтай, том, олон-хэрэглэгчийн системүүд дээр ашигтай байдаг. Ийм системүүд нь чөлөөт санах ойн хадгалалтад ихээхэн хэмжээний байнгын дарамтыг үүсгэж байдаг. Энэ боломжийг идэвхтэй болгож ар араас нь swap хийн гаргахыг (зогссон секундээр) `vm.swap_idle_threshold1` болон `vm.swap_idle_threshold2` хувьсагчуудын тусламжтай тохируулснаар зогссон процессуудтай холбоотой санах ойн хуудаснуудын дарааллыг ердийн хуудаслаж гаргах алгоритмаас илүү хурднаар багасгах боломжийг олгодог. Энэ нь хуудаслаж гаргах дэмонд тусламжийн гарыг өгөх болно. Энэ тохируулгыг танд хэрэгтэй л биш бол идэвхтэй болгож болохгүй, учир нь үүнийг та хийснээр үндсэндээ санах ойг илүү түргэн урьдчилан-хуудаслаж ингэснээр swap болон дискний багтаамжийг илүүтэйгээр идэхэд хүргэх юм. Жижиг систем дээр энэ тохируулга нь тодорхойлогдож болохуйц нөлөөлөлтэй байх ба харин боломжийн хуудаслалт аль хэдийн хийгээд байгаа том системүүдэд энэ тохируулга нь VM системд бүх процессуудыг санах ой уруу болон санах ойгоос хялбараар гаргах боломжийг бүрдүүлдэг.

#### 12.13.1.5. hw.ata.wc

FreeBSD 4.3-д IDE бичих кэш хийлтийг хаасан байдаг. Энэ нь IDE дискэнд бичих багтаамжийг багасгасан боловч хатуу диск үйлдвэрлэгчдийн гаргасан өгөгдлийн бүрэн бүтэн байдлын ноцтой асуудлуудаас болоод шаардлагатай болсон. Тэр асуудал нь IDE хөтлөгчүүд бичилт

дуусах үед худлаа мэдээлдэг явдал юм. IDE бичих кэшийг идэвхтэй болгосноор IDE хатуу дискнүүд ямар нэг дараалалгүйгээр бичихээс гадна диск их ачаалалтай үед зарим блокуудыг бичихэд заримдаа тодорхойгүй саатдаг. Сүйрэл болон тэжээлийн уналт файлын системийн ноцтой эвдрэлд хүргэж болзошгүй байдаг. FreeBSD-ийн анхдагч нь аюулгүй байхаар өөрчлөгдсөн. Харамсалтай нь үүний үр дүнд ажиллагааны асар том алдагдалд хүргэсэн бөгөөд хувилбар гарсны дараа бид бичих кэш хийлтийг анхдагчаар идэвхтэй байхаар буцаан өөрчилсөн юм. Та өөрийн систем дээрээ `hw.ata.wc` sysctl хувьсагчийг ажиглан анхдагч утгыг шалгах хэрэгтэй. Хэрэв IDE бичих кэш хийлт хаалттай бол та цөмийн хувьсагчийн утгыг 1 болгон түүнийг идэвхжүүлж болно. Үүнийг ачаалах үед ачаалагчаас хийх шаардлагатай. Цөм ачаалсны дараа хийхийг оролдвол ямар ч нөлөө үзүүлэхгүй.

Дэлгэрэнгүй мэдээллийн талаар ata(4)-с үзнэ үү.

#### 12.13.1.6. `SCSI_DELAY` (`kern.cam.scsi_delay`)

`SCSI_DELAY` цөмийн тохиргоо нь системийн ачаалах хугацааг багасгахад хэрэглэгддэг. Анхдагч утга нь нэлээн өндөр бөгөөд `15` секундын саатлыг ачаалах процессийн үед өгөхийг хариуцдаг. `5` секунд хүртэл багасгахад ихэвчлэн ажилладаг (ялангуяа орчин үеийн хөтлөгчүүдийн хувьд). Ачаалах үеийн тохируулга болох `kern.cam.scsi_delay` хувьсагчийг ашиглах хэрэгтэй. Энэ тохируулга болон цөмийн тохиргооны тохируулга нь *секундээр бишмиллисекундээр* утгыг хүлээн авдаг.

### 12.13.2. Зөөлөн Шинэчлэлтүүд

tunefs(8) програм файлын системийг нарийн тааруулахад ашиглагдаж болно. Энэ програм нь олон янзын тохируулгуудтай гэхдээ одоохондоо бид зөвхөн Зөөлөн Шинэчлэлтүүдийг идэвхжүүлэх ба хаах дээр анхаарах бөгөөд үүнийг дараах аргаар хийнэ:

```
# tunefs -n enable /filesystem
# tunefs -n disable /filesystem
```

Файлын систем нь холбогдсон байхдаа tunefs(8)-ээр өөрчлөгдөх боломжгүй. Зөөлөн Шинэчлэлтүүдийг идэвхжүүлэхэд тохирох үе нь аль ч хуваалтууд холболт хийгдээгүй байгаа ганц хэрэглэгчийн горим юм.

Зөөлөн Шинэчлэлтүүд нь мета-өгөгдлийн ажиллагааг мэдэгдэхүйц сайжруулдаг бөгөөд санах ойн кэшийг ашиглан голчлон файлын үүсгэлт болон устгалтыг хурдасгадаг. Бид Зөөлөн Шинэчлэлтүүдийг өөрийн бүх файлын системүүдэд ашиглахыг зөвлөж байна. Зөөлөн Шинэчлэлтүүдийн хоёр дутагдалтай талыг та мэдэж байх ёстой: Нэгдүгээрт, Зөөлөн Шинэчлэлтүүд нь сүйрэл болсон тохиолдолд файлын системийн бүрэн бүтэн байдалд баталгаа өгдөг боловч физик дискийг шинэчлэхэд хэдэн секундын (минут ч байж болно!) хоцрогдолтой байж болно. Хэрэв таны систем сүйрэхэд бусад тохиолдлоос илүүтэйгээр та хийсэн ажлаа алдаж болзошгүй юм. Хоёрдугаарт, Зөөлөн Шинэчлэлтүүд нь файлын системийн блокуудыг чөлөөлөхийг саатуулдаг. Хэрэв та бараг дүүрсэн файлын системтэй (root файл систем гэх зэрэг) байгаа бол `make installworld` зэрэг гол шинэчлэлтийг хийх нь файлын системийг зайгүй болгож шинэчлэлт амжилтгүй болох шалтгаанд хүргэж болох юм.

#### 12.13.2.1. Зөөлөн Шинэчлэлтүүдийн талаар дэлгэрэнгүй

Файлын системийн мета-өгөгдлийг диск уруу бичих уламжлалт хоёр хандлага байдаг. (Мета-өгөгдлийн шинэчлэлтүүд нь inode эсвэл сангууд зэрэг агуулгын бус өгөгдөлд хийх шинэчлэлтүүд юм)

Түүхээс авч үзэхэд анхдагч ажиллах горим нь мета-өгөгдлийн шинэчлэлтүүдийг синхроноор буюу зэрэг бичдэг байсан явдал юм. Хэрэв сан өөрчлөгдсөн бол систем өөрчлөлтийг диск уруу бичигдэхийг хүлээдэг. Файлын өгөгдлийн буферууд (файлын агуулгууд) буфер кэшээр дамжин диск уруу сүүлд нь асинхроноор хадгалагддаг. Энэ шийдлийн давуу тал нь аюулгүй ажилладаг. Хэрэв шинэчлэлтийн үед амжилтгүй болбол мета-өгөгдөл нь үргэлж бүрэн бүтэн байдаг. Файл эсвэл бүрэн үүсч эсвэл бүр ерөөсөө үүсдэггүй. Хэрэв файлын өгөгдлийн блокууд сүйрэл болох үед буферийн кэшээс диск уруу өөрсдийн гарах замаа олохгүй байгаа бол fsck(8) нь үүнийг таньж файлын уртыг 0 болгон файлын системийг засварладаг. Нэмж хэлэхэд энэ шийдэл нь цэвэрхэн ба хялбар юм. Сул тал нь мета-өгөгдлийн өөрчлөлтүүд нь удаан байдаг. `rm -r` тушаал жишээ нь сан дахь бүх файлуудад дараалан хандах бөгөөд гэхдээ сан болгоны өөрчлөлт (файлын устгалт) синхроноор зэрэг диск уруу бичигддэг. Үүнд сан уруу өөрт нь хийгдэх шинэчлэлтүүд, inode хүснэгт болон магадгүй файлын гаргасан шууд бус блокуудад хийх шинэчлэлтүүд ордог. Том иерархуудыг задлахад (`tar -x`) үүний нэгэн адилаар авч үздэг.

Хоёр дахь нь асинхрон мета-өгөгдлийн шинэчлэлтүүд юм. Энэ нь Линукс/ext2fs-ийн хувьд анхдагч байх бөгөөд *BSD ufs-ийн хувьд `mount -o async` байх юм. Бүх мета-өгөгдлийн шинэчлэлтүүд нь буфер кэшээр бас дамждаг, тэгэхээр тэдгээр нь файлын агуулгын өгөгдлийн шинэчлэлтүүдтэй харилцан холилдох болно. Энэ шийдлийн давуу тал нь мета-өгөгдөл бүрийн шинэчлэлт диск уруу бичигдэхийг хүлээдэггүй бөгөөд ингэснээр ихээхэн хэмжээний мета-өгөгдлийн шинэчлэлтүүдийг хийдэг бүх үйлдлүүд синхрон хийгдэхээс хамаагүй хурдан ажилладаг. Мөн энэ шийдэл нь цэвэрхэн бас энгийн бөгөөд ингэснээр хорхойнууд (алдаа) код уруу мөлхөн орох эрсдэл бага юм. Сул тал нь файлын системийн бүрэн бүтэн төлвийн ямар нэг баталгаа ерөөсөө байдаггүй. Хэрэв их хэмжээний мета-өгөгдөл шинэчлэх үйлдлийн явцад амжилтгүй болсон бол (тэжээлийн тасалдал, эсвэл хэн нэг нь дахин эхлүүлэх товч дарсан зэрэгт) файлын систем тааж болшгүй төлөвт үлдэх болно. Систем дахин ачаалаад дуусахад файлын системийн төлөвийг мэдэх боломжгүй байдаг; inode хүснэгт эсвэл холбоотой сангийн шинэчлэлтүүд бичигдээгүй байхад файлын өгөгдлийн блокууд диск уруу аль хэдийн бичигдчихсэн байж болох юм. Ер нь гаргасан замбараагүйтлийг (учир нь хэрэгцээтэй мэдээлэл диск дээр байхгүй) цэвэрлэж чаддаг `fsck` тушаалын шийдлийг хийх боломжгүй. Хэрэв файлын систем засвар хийж чадахааргүй эвдэрсэн бол түүнд дээр newfs(8)-ийг хэрэглэж нөөцөөс сэргээхээс өөр аргагүй юм.

Энэ асуудлын шийдэл нь *бохир бүсийн бүртгэл* буюу бас *журналчлалт* гэгддэг шийдлийг гаргах явдал бөгөөд энэ ухагдахуун нь тогтвортой хэрэглэгддэггүй ба шилжүүлэлтийн бүртгэлийн бусад хэлбэрүүдэд бас заримдаа ашиглагддаг. Мета-өгөгдлийн шинэчлэлтүүд нь синхроноор бичигдсэн хэвээр байх бөгөөд гэхдээ зөвхөн дискний жижиг бүсэд бичигдэнэ. Дараа нь тэдгээрийг тэдний зөв байрлал уруу зөөдөг. Бүртгэлийн талбар нь диск дээр бага, үргэлжилсэн бүс байдаг учраас бүр хүнд үйлдлүүдийн үед ч гэсэн дискний толгойнууд шилжихэд хол зайтай биш байдаг болохоор эдгээр үйлдлүүд нь синхрон шинэчлэлтүүдээс илүү хурдан байдаг. Мөн энэ шийдлийн төвөгтэй байдал нь маш хязгаарлагдмал болохоор алдаанууд байх эрсдэл нь бага байдаг. Сул тал нь бүх мета-өгөгдөл нь хоёр удаа бичигддэг (бүртгэлийн бүсэд нэг удаа болон зөв байрлал уруу бас нэг удаа) болохоор энгийн ажлын

хувьд ажиллагааны "өөдрөг бус үзэгдэл" гарч болзошгүй юм. Нөгөө талаас сүйрэл болоод систем дахин ачаалаад дуусахад хүлээгдэж байгаа бүх мета-өгөгдлийн үйлдлүүд бүрттэлийн талбараас хурдан буцаагдаж эсвэл гүйцэд хийгдэн дуусч болох бөгөөд энэ нь файлын системийг хурдан эхлүүлэхэд хүргэдэг.

Беркли FFS-ийн хөгжүүлэгч Кирк МкКюзик энэ асуудлыг Soft Updates буюу Зөөлөн Шинэчлэлтүүдээр шийдсэн: хүлээгдэж байгаа бүх мета-өгөгдлийн шинэчлэлтүүд нь санах ойд хадгалагдах бөгөөд диск уруу эрэмбэлэгдсэн дарааллаар бичигддэг ("дараалуулсан мета-өгөгдлийн шинэчлэлтүүд"). Энэ нь мета-өгөгдлийн хүнд үйлдлүүдийн үед хэрэв эрт хийгдсэн шинэчлэлтүүд диск уруу бичигдээгүй санах ойд байж байхад нь сүүлд хийгдэх шинэчлэлтүүд тэдгээрийг "барьж" авдаг. Тэгэхээр сангийн хувьд хэлбэл түүнд хийгдэх бүх үйлдлүүд нь санах ойд шинэчлэлт диск уруу бичигдэхээс өмнө хийгддэг (өгөгдлийн блокууд нь мета-өгөгдлөөсөө түрүүлээд диск дээр байж байхгүйгээр өөрсдийн байрлалынхаа дагуу эрэмбэлэгддэг ). Хэрэв систем сүйрвэл энэ нь "бүртгэл урагшлуулахад" хүргэдэг: диск уруу гарах замаа олохгүй байгаа бүх үйлдлүүд хэзээ ч хийгдээгүй юм шиг байдаг. Файлын системийн бүрэн бүтэн төлөв хадгалагдаж 30-аас 60 секундын өмнөх төлөвт ордог. Хэрэглэгдэж байгаа эх үүсвэрүүдийг тэдгээрийн өөрсдийнх харгалзах битмапуудад: блокууд болон inode-уудад байдаг шигээр тэмдэглэхийг үүнд ашигласан алгоритм нь баталгаатай хангадаг. Сүйрэл болсны дараа зөвхөн гарсан эх үүсвэр суллан гаргалтын алдаа нь яг үнэндээ "чөлөөтэй" мөртлөө "ашиглагдаж байгаа" гэж тэмдэглэгдсэн эх үүсвэрүүд байдаг. fsck(8) энэ байдлыг таних бөгөөд ашиглагдаагүй байгаа эх үүсвэрүүдийг чөлөөлдөг. Сүйрлийн дараа файлын системийн бохир төлвийг авч үзэлгүйгээр хүчээр `mount -f` тушаалаар холбох нь аюулгүй юм. Ашиглагдаагүй байж болзошгүй эх үүсвэрүүдийг чөлөөлөхдөө fsck(8)-г сүүлд нь ажиллуулах хэрэгтэй. Энэ нь *ард ажиллах fsck*-ийн цаана байгаа санаа юм: системийг эхлүүлэх үед зөвхөн файлын системийн *хормын зураг* бичигддэг. `fsck`-г сүүлд нь ажиллуулж болно. Дараа нь бүх файлын системүүд "бохир" холбогдож системийн эхлэлт олон хэрэглэгчийн горимд үргэлжилдэг. Дараа нь ард ажиллах `fsck`-үүд ашиглагдаагүй байгаа эх үүсвэрүүдийг чөлөөлөхөөр шаардлагатай байгаа бүх файлын системийн хувьд ажиллахаар төлөвлөгддөг. (Зөөлөн Шинэчлэлтүүд ашигладаггүй файлын системүүдэд ердийн нүүрэн дээр ажиллах `fsck` хэрэгтэй хэвээр байна)

Давуу тал нь мета-өгөгдлийн үйлдлүүд нь асинхрон шинэчлэлтүүдтэй бараг л адил хурдан байдаг (өөрөөр хэлбэл мета-өгөгдлийг хоёр дахин бичдэг *бүртгэл хийлтээс* хурдан байдаг). Сул талууд нь төвөгтэй код (хэрэглэгчийн өгөгдлийн алдагдлын хувьд их мэдрэмтгий талбар дахь байж болох алдаануудын тэр өндөр эрсдэлийг хэлж байна) болон санах ойн илүү хэрэглээ юм. Мөн хэн нэгний хэрэглэж байсан хувийн тохиргоонууд ч бас байдаг. Сүйрэл болсны дараа файлын системийн төлөв "хуучин" юм шиг харагддаг. Стандарт синхрон хандлага нь `fsck`-ийн дараа зарим нэг тэг-урттай файлуудыг үлдээхэд хүргэсэн нөхцөлд тэдгээр файлууд нь Зөөлөн Шинэчлэлтүүдтэй файлын системийн үед огт байдаггүй бөгөөд учир нь мета-өгөгдөл болон файлын агуулгууд хэзээ ч диск уруу бичигдээгүй байдаг. Дискний зай нь магадгүй `rm` ажиллуулснаас хэсэг хугацааны дараа диск уруу шинэчлэлтүүд бичигдэх хүртэл сулардаггүй. Энэ нь бүх файлуудыг хоёр дахин хадгалахад хангалттай хүрэлцэхүйц хэмжээний чөлөөтэй зай байхгүй файлын систем дээр их хэмжээний өгөгдлийг суулгаж байх үед асуудлууд гарахад хүргэж болох юм.

# 12.14. Цөмийн хязгаарууд тохируулах нь

### 12.14.1. Файл/Процессийн хязгаарууд

#### 12.14.1.1. `kern.maxfiles`

`kern.maxfiles` нь таны системийн шаардлагуудаас хамаараад дээшилж эсвэл доошилж болно. Энэ хувьсагч нь таны систем дээрх файлын тодорхойлогчуудын (descriptor) хамгийн их тоог илэрхийлдэг. Файлын тодорхойлогчийн хүснэгт дүүрсэн тохиолдолд `file: table is full` буюу файл: хүснэгт дүүрсэн гэсэн мэдээлэл давтагдан системийн богино мэдээллийн буфферт үзэгдэх бөгөөд үүнийг `dmesg` тушаал ашиглан үзэж болдог.

Нээлттэй файл, сокет эсвэл fifo болгон нэг файлын тодорхойлогч хэрэглэдэг. Ажиллаж байгаа том-хэмжээний сервер зэрэгцээ ажиллаж байгаа үйлчилгээнүүдийн тоо болон төрлөөс хамааран олон мянган файлын тодорхойлогчуудыг өлхөн шаардаж болох юм.

Хуучин FreeBSD хувилбаруудад `kern.maxfiles`-ийн анхдагч утга нь таны цөмийн тохиргооны файлын `maxusers` тохируулгаас гарсан байдаг. `kern.maxfiles` нь `maxusers` утгатай пропорционалаар өсдөг. Өөрчлөн тохируулсан цөмийг бүтээхдээ энэ цөмийн тохиргооны тохируулгыг өөрийн системийн хэрэглээний дагуу зааж өгөх нь зүйтэй байдаг. Энэ тооноос хамаарч цөм өөрийн ихэнх урьдчилан-тодорхойлсон хязгаарууд өгдөг. Ажиллагаанд байгаа машин яг үнэндээ нэг удаа 256 хэрэглэгч зэрэг холбогдоогүй байж болох боловч өндөр-хэмжээний вэб серверийнхтэй адил эх үүсвэрүүд хэрэгтэй байж болох юм.

`kern.maxusers` хувьсагч нь системд байгаа санах ойн дээр үндэслэн ачаалах үед автоматаар тавигддаг бөгөөд ажиллаж байх явцад зөвхөн уншигдах `kern.maxusers` sysctl хувьсагчийн утгыг шалгаж тогтоогдож болох юм. Зарим сайтууд `kern.maxusers`-ийн илүү их эсвэл бага утгуудыг шаардаж үүнийг ачаалагчаар тааруулагдахаар тохируулж болох юм; 64, 128, болон 256 утгууд нь ховор байдаг. Танд асар их тооны файлын тодорхойлогчууд хэрэгтэй л биш бол бид 256-аас дээш байлгахыг зөвлөдөггүй; өөрсдийн анхдагч утгуудад `kern.maxusers`-р заагддаг, тааруулагдах боломжтой утгуудын олонх нь тус тусдаа ачаалалтын үед эсвэл ажиллах явцад /boot/loader.conf-оор эсвэл энэ баримтын хаа нэгтээ тайлбарласнаар өөрчлөгдөж болдог (loader.conf(5) гарын авлага эсвэл /boot/defaults/loader.conf файлыг санаа авахын тулд үзнэ үү).

Хуучин хувилбаруудад хэрэв та `maxusers`-ийг `0` гэж шууд зааж өгсөн бол систем автоматаар тааруулж өгдөг . Энэ тохируулгыг заахдаа ялангуяа та хэрэв X Цонхны Систем ашиглаж байгаа эсвэл програм хангамж хөрвүүлж байгаа бол `maxusers`-ийг хамгийн багадаа 4 гэж заахыг хүсэх болно. Шалтгаан нь гэвэл `maxusers`-ээр заагдсан хамгийн чухал хүснэгт бол `20 + 16 * maxusers` гэж заагдсан процессуудын хамгийн их тоо бөгөөд хэрэв та `maxusers`-ийг 1 гэж заасан бол та 18 орчмыг нь ачаалах үед системийг эхлүүлэхэд болон 15 орчмыг нь таныг X Цонхны Системийг эхлүүлэхэд магадгүй үүсэж та нийт зөвхөн 36 зэрэг процесстой байж болох юм. Гарын авлагыг унших зэрэг хялбар бодлого хүртэл шүүх, шахсаныг задлах, болон үзэхэд зориулж есөн процессийг эхлүүлдэг. `maxusers`-ийг 64 гэж заах нь бараг л бүх хэрэгцээнд хангалттай байх 1044 зэрэг процесстой байж болохыг танд зөвшөөрнө. Гэхдээ өөр програм эхлүүлэхээр оролдож байх үед эсвэл их олон тооны зэрэгцээ хэрэглэгчидтэй сервер (`ftp.FreeBSD.org`-той адил) ажиллуулж байхад айдас төрүүлэм буюу proc хүснэгт дүүрсэн гэсэн алдаа хэрэв та харах юм бол үргэлж энэ тоог ихэсгэн цөмийг дахин бүтээж болох юм.

`maxusers` нь таны машин уруу нэвтрэх хэрэглэгчдийн тоог *хязгаарладаггүй*. Энэ нь ердөө л таны систем дээр байж болох хамгийн их хэрэглэгчийн тоо

болон тэдгээр тус бүрийн ажиллуулах процессийн тооноос хамааран төрөл бүрийн хүснэгтийн хэмжээнүүдийг боломжийн утгуудаар зааж өгдөг.

#### 12.14.1.2. `kern.ipc.somaxconn`

`kern.ipc.somaxconn` sysctl хувьсагч нь шинэ TCP холболтуудыг хүлээн авахад зориулсан сонсох дарааллын хэмжээг хязгаарладаг. Анхдагч утга `128` нь ачаалал ихтэй вэб серверийн орчин дахь шинэ холболтуудыг хүлээж авахад ерөнхийдөө хэтэрхий бага юм. Тийм орчны хувьд энэ утгыг `1024` эсвэл түүнээс их болгохыг зөвлөдөг. Үйлчилгээний дэмон нь өөрөө сонсох дарааллын хэмжээгээ (өөрөөр хэлбэл sendmail(8), эсвэл Apache) хязгаарлаж болох боловч ихэвчлэн өөрийн тохиргооны файлдаа дарааллын хэмжээг тааруулах тохиргооны мөртэй байдаг. Их хэмжээний сонсох дарааллууд нь бас Үйлчилгээг Зогсоох халдлагуудаас () илүү сайн зайлсхийж ажилладаг.

### 12.14.2. Сүлжээний хязгаарууд

`NMBCLUSTERS` цөмийн тохиргооны тохируулга нь системд байгаа сүлжээний Mbuf-уудын тоог зааж өгдөг. Бага тооны Mbuf-уудтай трафикийн ачаалал ихтэй сервер FreeBSD-ийн чадварт саад болдог. Кластер бүр ойролцоогоор 2 K санах ойг илэрхийлдэг, тийм болохоор 1024 гэсэн утга нь сүлжээний буферуудад зориулж хадгалсан 2 мегабайт цөмийн санах ойг илэрхийлнэ. Хичнээн хэрэгтэйг олохын тулд хялбар тооцоо хийж болно. Хэрэв та хамгийн ихдээ 1000 зэрэгцээ холболтуудтай, холболт бүр нь 16 K хүлээн авах болон 16 K илгээх буферийг иддэг вэб сервертэй бол танд ойролцоогоор вэб серверийг хангахын тулд 32 MB хэмжээтэй тэнцэх сүлжээний буферууд хэрэгтэй болно. Практикаар ер нь 2-оор үржүүлдэг, тэгэхээр 2x32 MB / 2 KB = 64 MB / 2 kB = 32768 болох юм. Бид их санах ойтой машинуудын хувьд утгуудыг 4096-аас 32768-ын хооронд байлгахыг зөвлөдөг. Энэ параметрийн хувьд өндөр утгыг ямар ч нөхцөлд тавьж болохгүй, учир нь энэ нь ачаалах үеийн сүйрэлд хүргэж болно. netstat(1)-д `-m` тохируулгыг ашиглаж сүлжээний кластерийн ашиглалтыг ажиглаж болох юм.

`kern.ipc.nmbclusters` ачаалалтын тааруулах боломжтой тохируулга нь ачаалах үед үүнийг тааруулахад хэрэглэгдэх ёстой. Зөвхөн FreeBSD-ийн хуучин хувилбарууд `NMBCLUSTERS` цөмийн config(8) тохируулгыг ашиглахыг танаас шаарддаг.

sendfile(2) системийн дуудлагыг өргөнөөр ашигладаг завгүй серверүүдийн хувьд `NSFBUFS` цөмийн тохиргооны тохируулгын тусламжтай эсвэл түүний утгыг /boot/loader.conf-д зааж sendfile(2) буферуудын тоог ихэсгэх шаардлагатай байж болох юм (дэлгэрэнгүйг loader(8)-с үзнэ үү). Процессууд `sfbufa` төлөвт харагдах нь энэ параметрийг тааруулах хэрэгтэйг ихэвчлэн заадаг. `kern.ipc.nsfbufs` sysctl хувьсагч нь цөмөөр тохируулагдсан хувьсагч дахь зөвхөн уншигддаг гялбаа юм. Энэ параметр нь `kern.maxusers`-ийн хэмжээгээр тааруулагддаг, гэхдээ үүнийг түүний дагуу тохируурах шаардлагатай байж болох юм.

Сокет блок-хийгддэггүй гэж тэмдэглэгдсэн ч гэсэн блок-хийгддэггүй сокет дээр sendfile(2)-ийг дуудах нь хангалттай хэмжээний `struct sf_buf`-уудыг бий болготол sendfile(2) дуудлага блок хийгдэхэд хүргэж болох юм.

#### 12.14.2.1. net.inet.ip.portrange.*

net.inet.ip.portrange.* sysctl хувьсагчууд нь TCP болон UDP сокетуудад автоматаар уягдах портын дугаарын хүрээнүүдийг хянадаг. Гурван хүрээ байдаг: доод хүрээ, анхдагч хүрээ, болон өндөр хүрээ. Ихэнх сүлжээний програмууд нь анхдагчаар 1024 болон 5000 байдаг net.inet.ip.portrange.first болон net.inet.ip.portrange.last хувьсагчуудаар хянагддаг анхдагч хүрээг ашигладаг. Уягдах портын хүрээнүүд гарах холболтуудад ашиглагддаг бөгөөд зарим тохиолдолд систем дэх портууд дуусч болох юм. Энэ нь ихэвчлэн таныг ачаалал ихтэй вэб прокси ашиглаж байхад гардаг. Ихэвчлэн ирж байгаа холболтуудыг хүлээн авдаг ердийн вэб сервер эсвэл захидал дамжуулагч зэрэг хязгаарлагдмал тооны гарах холболтуудтай серверүүдийг ажиллуулж байхад портын хүрээ нь асуудал биш юм. Таны порт дуусаж болох тийм тохиолдлуудад net.inet.ip.portrange.last хувьсагчийг даруухнаар ихэсгэхийг зөвлөдөг. 10000, 20000 эсвэл 30000 нь боломжийн утгууд юм. Портын хүрээг өөрчилж байхдаа галт ханын нөлөөллүүдийг бас бодолцох хэрэгтэй. Зарим галт хана их хэмжээний портуудыг хааж болох бөгөөд (ихэнхдээ бага дугаарын портууд) систем өндөр дугаарын портуудыг гарах холболтууддаа ашигладгийг бодолцох ёстой - ийм учраас net.inet.ip.portrange.first-ийг багасгахыг зөвлөдөггүй.

#### 12.14.2.2. TCP хурд сааруулагч бүтээгдэхүүнүүд

TCP хурд сааруулагч бүтээгдэхүүний хязгаарлалт нь NetBSD дэх TCP/Vegas-тай адилхан юм. net.inet.tcp.inflight.enable sysctl хувьсагчийг 1 болгон тохируулж үүнийг идэвхжүүлдэг. Систем холболт бүрийн хувьд хурд сааруулагч бүтээгдэхүүнийг тооцоолохыг оролддог бөгөөд сүлжээн дэх дараалалд оруулах өгөгдлийн хэмжээг хамгийн боломжийн нэвтрүүлэх чадамжийг байнга барьж байх тэр хэмжээнд хүргэж хязгаарладаг.

Хэрэв та өгөгдлийг модемууд, Гигабит Ethernet, эсвэл бүр өндөр хурдны WAN холболтуудаар (эсвэл дурын өндөр хурд сааруулагч бүтээгдэхүүнтэй холболт) дамжуулж байгаа бол ялангуяа та бас цонх өсгөлтийг ашиглаж байгаа эсвэл том илгээх цонх тохируулсан бол энэ боломж нь ашигтай юм. Хэрэв та энэ тохируулгыг идэвхжүүлэх бол бас net.inet.tcp.inflight.debug-ийг 0 (дибаг хийхийг болиулах) болгож тохируулах хэрэгтэй бөгөөд үйлдвэрлэлийн ашиглалтад net.inet.tcp.inflight.min-ийг хамгийн багаар бодоход 6144 болгох нь ашигтай байж болох юм. Гэхдээ хамгийн бага тоог өндөр болгох нь холболтоос хамааран хурд хязгаарлалтыг идэвхтэйгээр болиулж болохыг санах хэрэгтэй. Хязгаарлах боломж нь дундын чиглүүлэлтийн үед бүтээгдсэн өгөгдлийн хэмжээг багасгах бөгөөд пакетийн дарааллуудыг сольж локал хостын интерфэйс дэх дараалал дээр бүтээгдсэн өгөгдийн хэмжээг мөн багасгадаг. Дарааламд орсон цөөн тооны пакетуудтай, ялангуяа удаан модемоор дамжсан интерактив холболтууд нь бага *Round Trip Times буюу Эргэн Аялах Хугацаатайгаар* ажиллаж бас чаддаг. Гэхдээ энэ боломж нь зөвхөн өгөгдөл дамжуулалтад (илгээх / сервер талын) нөлөөлдгийг санах хэрэгтэй. Энэ нь өгөгдөл хүлээн авахад нөлөө үзүүлэхгүй (татаж авах).

net.inet.tcp.inflight.stab-ийг тааруулахыг *зөвлөдөггүй*. Энэ параметр нь хурд сааруулах бүтээгдэхүүний цонхны тооцоололд нэмсэн 2 хамгийн их пакетийг илэрхийлж анхдагчаар 20 байдаг. Энэ алгоритмийг тогтворжуулах болон өөрчлөгдөж байгаа нөхцлүүдэд хариу өгөх боломжийг сайжруулахад нэмэлт цонх шаардлагатай боловч энэ нь бас удаан холболт дээр ping хийх хугацаа ихэсгэхэд хүргэдэг (гэхдээ таныг энэ (inflight) алгоритмийг ашиглаагүй байхад гарсан үр дүнгээс хамаагүй бага хэвээр л байна). Ийм тохиолдолд энэ параметрийг 15, 10, эсвэл 5 болгон багасгахыг хүсэж болох юм; мөн хүссэн үр дүндээ хүрэхийн тулд

`net.inet.tcp.inflight.min` хувьсагчийг (жишээ нь 3500 болгож) бас багасгаж болох юм. Эдгээр параметрүүдийг багасгах нь хамгийн сүүлд авах арга хэмжээ байх ёстой юм.

### 12.14.3. Виртуал санах ой

#### 12.14.3.1. `kern.maxvnodes`

vnode нь файл эсвэл сангийн дотоод дүрслэл юм. Тэгэхээр үйлдлийн системд байх vnode-ийн тоог ихэсгэх нь диск I/O-г багасгадаг. Энэ нь ихэвчлэн үйлдлийн системээр зохицуулагддаг бөгөөд өөрчлөх хэрэггүй байдаг. Зарим тохиолдолд диск I/O нь гол асуудал учруулж системд vnode байхгүй болж байвал энэ тохируулгыг ихэсгэх хэрэгтэй болно. Идэвхгүй болон чөлөөтэй RAM-ийн хэмжээг бодолцох шаардлагатай.

Тухайн үед ашиглагдаж байгаа vnode-уудыг үзэхдээ:

```
# sysctl vfs.numvnodes
vfs.numvnodes: 91349
```

Хамгийн их vnode-уудыг үзэхдээ:

```
# sysctl kern.maxvnodes
kern.maxvnodes: 100000
```

Хэрэв тухайн үеийн vnode ашиглалт хамгийн их хэмжээ уруу бараг дөхөж байвал `kern.maxvnodes`-ийг 1,000-аар ихэсгэх нь зүйтэй байж болох юм. `vfs.numvnodes`-ийн тоон дээр бас анхаарлаа хандуулаарай. Хэрэв энэ нь дахин хамгийн их уруугаа дээшилбэл `kern.maxvnodes`-ийг цааш ихэсгэх шаардлагатай болно. top(1)-ийн гаргасан дүнгээс таны санах ойн өөрчлөлт харагдах ёстой. Түрүүнийхээс илүү санах ой идэвхтэй байх ёстой.

## 12.15. Swap зай нэмэх нь

Та яаж ч сайн төлөвлөсөн байлаа гэсэн заримдаа систем таны бодсоноор ажилладагүй. Хэрэв танд swap зай илүү хэрэгцээтэйг мэдвэл та үүнийг амархнаар нэмж болно. Та гурван аргаар swap зайг ихэсгэж болно: шинэ хатуу диск нэмэх, NFS-ийн тусламжтай swap идэвхжүүлэх болон байгаа хуваалт дээр swap файл үүсгэж ихэсгэж болно.

Swap зайг хэрхэн шифрлэх, ямар тохируулгууд байгаа болон яагаад хийх ёстой талаар гарын авлагын Swap зайг шифрлэх хуудсанд хандана уу.

### 12.15.1. Шинэ эсвэл байгаа диск дээрх swap

swap-т зориулж шинэ хатуу диск нэмэх нь байгаа диск дээр хуваалт нэмэхээсээ илүүтэй ажиллагааны хувьд сайжруулдаг. Хуваалтуудыг үүсгэх болон хатуу диск нэмэх талаар Диск нэмэх хэсэгт тайлбарласан байгаа. Эхний Тохиргоо хэсэгт хуваалтын байдал болон swap хуваалтын зайн талаарх анхаарах зүйлсийг тайлбарласан байгаа.

swapon(8) ашиглан swap хуваалтыг системийг нэмж өгнө. Жишээ нь:

```
# swapon /dev/ada1s1b
```

Өгөгдөлтэй ч гэсэн холбогдоогүй байгаа хуваалтыг ашиглах боломжтой. swapon(8) ашигласнаар өгөгдэлтэй байгаа хуваалт дээр бичилт хийгдэж өгөгдлийг нь устгах болно. swap хэлбэрээр нэмэгдэх хуваалт нь яг тэр зорилгоор ашиглагдах гэж байгаа эсэхийг swapon(8) ажиллуулахаасаа өмнө шалгаарай.

Ачаалахад ашиглагдахаар автоматаар энэ swap хуваалтыг нэмэхийн тулд /etc/fstab файлд тухайн хуваалтын талаарх оруулгыг нэмнэ:

```
/dev/ada1s1b    none    swap    sw  0   0
```

/etc/fstab дахь оруулгуудын талаарх тайлбарыг fstab(5) гарын авлагын хуудаснаас үзнэ үү.

### 12.15.2. NFS-ийн тусламжтай swap хийх нь

NFS-ийн тусламжтай swap хийхийг зөвхөн swap хийх локал хатуу диск танд байхгүй үед л зөвлөдөг; NFS swap хийх нь байгаа сүлжээний хурдаар хязгаарлагддаг бөгөөд NFS серверт нэмэлт ачаалал үзүүлдэг.

### 12.15.3. Swap файлууд

Та swap файл болгон ашиглахаар заасан хэмжээтэй файлыг үүсгэж болно. Энд байгаа жишээн дээр бид /usr/swap0 гэсэн нэртэй 64MB файлыг ашиглана. Мэдээж та хүссэн ямар ч нэрээ ашиглаж болно.

*Жишээ 9. Swap файл FreeBSD дээр үүсгэх нь*

1. GENERIC цөм нь энэ үйлдэлд шаардлагатай санах ойн дискний драйверийг (md(4)) агуулсан байдаг. Цөмийг тохируулан өөрчлөх гэж байгаа бол доорх мөрийг цөмийн тохиргооны файлдаа оруулахаа мартуузай:

   ```
   device   md
   ```

   Өөрийн хэрэгцээнд зориулж цөм бүтээх талаар FreeBSD цөмийг тохируулах нь бүлгээс үзнэ үү.

2. Swap файл (/usr/swap0) үүсгэнэ:

   ```
   # dd if=/dev/zero of=/usr/swap0 bs=1024k count=64
   ```

3. Зөв зөвшөөрлүүдийг (/usr/swap0-д) нээж тохируулна:

```
# chmod 0600 /usr/swap0
```

4. /etc/rc.conf-д swap файлыг идэвхжүүлнэ:

```
swapfile="/usr/swap0"   # Set to name of swapfile if aux swapfile desired.
```

5. Машиныг дахин эхлүүлнэ эсвэл swap файлыг шууд идэвхжүүлэхийн тулд дараах тушаалыг ажиллуулна:

```
# mdconfig -a -t vnode -f /usr/swap0 -u 0 && swapon /dev/md0
```

# 12.16. Тэжээл болон Эх үүсвэрийн Удирдлага

Тоног төхөөрөмжийн эх үүсвэрүүдийг үр ашигтай ашиглах нь чухал юм. ACPI танилцуулагдахаас өмнө системийн тэжээлийн ашиглалт болон дулааны шинж чанаруудыг удирдахад үйлдлийн системүүдийн хувьд хэцүү, уян хатан биш байсан. Тоног төхөөрөмж нь BIOS-оор удирдагддаг байсан болохоор тэжээлийн удирдлагын тохиргоонуудын харагдац бага бөгөөд хэрэглэгчид хянах боломж бага байсан юм.Зарим нэгэн хязгаарлагдмал тохиргооны боломж *Advanced Power Management буюу Тэжээлийн Дэвшилттэй Удирдлага (APM)* интерфэйсээр хийгдэх боломжтой байсан. Тэжээл болон Эх үүсвэрийн Удирдлага нь орчин үеийн үйлдлийн системийн түлхүүр хэсгүүдийн нэг юм. Жишээ нь таны системийн хэм гэнэт нэмэгдэх тохиолдолд системийн хязгааруудыг үйлдлийн систем монитор хийхийг (магадгүй танд мэдээлэхийг) хүсэж болох юм.

FreeBSD Гарын авлагын энэ хэсэгт бид ACPI-ийн талаар нэвтэрхий мэдээллээр хангах болно. Цааш нэмж уншихад зориулсан мэдээллүүдийг төгсгөл хэсэгт оруулсан байгаа.

## 12.16.1. ACPI гэж юу вэ?

Advanced Configuration and Power Interface буюу Дэвшилттэй Тохиргоо ба Тэжээлийн Интерфэйс (ACPI) нь тоног төхөөрөмжийн эх үүсвэрүүд болон тэжээлийн удирдлагад (эндээс нэр гарсан) зориулсан стандарт интерфэйсийг хангах зорилгоор үйлдвэрлэгчдийн холбооноос бичин гаргасан стандарт юм. Энэ нь *Үйлдлийн Системээр заалгасан тохиргоо ба Тэжээлийн Удирдлагын* түлхүүр элемент юм, өөрөөр хэлбэл: энэ нь илүү хяналт болон уян хатан байдлыг үйлдлийн системд (OS) хангадаг. ACPI-г танилцуулахаас өмнө одоогийн Залгаад Тоглуулах интерфэйсүүдийн хязгааруудыг орчин үеийн системүүд "сунгасан" юм. ACPI нь APM-ийн (Advanced Power Management буюу Тэжээлийн Дэвшилтэт Удирдлага) шууд залгамжлагч юм.

## 12.16.2. Тэжээлийн Дэвшилтэт Удирдлагын (APM) сул талууд

*Тэжээлийн Дэвшилтэт Удирдлага (APM)* боломж нь системийн тэжээлийн ашиглалтыг түүний ажиллагаан дээр үндэслэн хянадаг. APM BIOS нь (систем) үйлдвэрлэгчээс хангагддаг бөгөөд тоног төхөөрөмжийн тавцан бүрийн хувьд онцлог байдаг. OS дахь APM драйвер нь

тэжээлийн түвшингүүдийн удирдлагыг зөвшөөрдөг *APM Програм хангамжийн Интерфэйс* уруу хандах хандалтыг зуучилж өгдөг. APM-ийг 2000 онд болон тэрнээс өмнө үйлдвэрлэсэн системүүдэд ашиглах ёстой хэвээр байдаг.

APM-д дөрвөн үндсэн асуудал байдаг. Нэгдүгээрт, тэжээлийн удирдлага (үйлдвэрлэгчийн онцлогтой) BIOS-оор хийгддэг бөгөөд OS нь энэ талын ямар ч мэдлэг байдаггүй. Үүний нэг жишээ нь хэрэглэгч хатуу дискний сул зогсох хугацааг APM BIOS дээр зааж өгөөд тэр нь зааснаас илүү гарвал BIOS хатуу дискийг OS-ийн зөвшөөрөлгүйгээр эргүүлдэг. Хоёрдугаарт, APM-ийн логик BIOS-д суулгагдсан байдаг бөгөөд OS-ийн эрх хэмжээнээс гадна ажилладаг. Энэ нь хэрэглэгчид өөрсдийн APM BIOS-ийг зөвхөн шинэ хувилбараар нь ROM уруу нь шарж асуудлуудыг засварлах боломжтой гэсэн үг юм; энэ нь амжилтгүй болбол системийг дахин сэргээгдэхгүй төлөвт орхиж болох боломжтой маш аюултай процедур юм. Гуравдугаарт, APM нь үйлдвэрлэгчийн онцлогтой технологи бөгөөд энэ нь маш олон адил төсөөтэй байдал (чармайлтуудын хуулбар) болон нэг үйлдвэрлэгчийн BIOS-д олдсон алдаанууд бусад үйлдвэрлэгчдийн хувьд шийдэгдээгүй байж болно гэсэн үг юм. Хамгийн сүүлд гэхдээ төгсгөлийнх биш, APM BIOS нь тэжээлийн маш нарийн бодлого эсвэл машины зориулалтад зориулагдан маш сайн тохируулагдах тийм шийдлийг хийхэд хангалттай зайгүй байдаг.

*Залгаад Тоглуулах BIOS (PNPBIOS)* нь олон тохиолдолд найдвартай биш байсан юм. PNPBIOS нь 16-битийн технологи, тийм болохоор OS нь PNPBIOS аргуудтай холбогдохдоо 16-битийн эмуляц хэрэглэх шаардлагатай болдог.

FreeBSD-ийн APM драйвер apm(4) гарын авлагын хуудсанд баримтжуулагдсан байдаг.

### 12.16.3. ACPI-г тохируулах нь

acpi.ko драйвер нь системийг эхлүүлэх үед loader(8)-оор анхдагчаар ачаалагддаг бөгөөд цөмд оруулж *хөрвүүлэгдэх* ёсгүй. Үүний цаадах шалтгаан нь модулиудтай ажиллах хялбар байдаг, өөрөөр хэлбэл цөмийг дахин хөрвүүлэлгүйгээр өөр acpi.ko уруу шилждэг. Энэ нь тест хийлтийг илүү амархан болгодог давуу талтай юм. Нөгөө нэг шалтгаан нь системийг ажиллуулж дууссаны дараа ACPI-г ажиллуулахад ихэвчлэн сайн ажилладаггүй. Хэрэв та асуудлуудтай учирч байгаа бол ACPI-г бүхэлд нь хаах хэрэгтэй. Энэ драйверийг ачаалсны дараа буулгаж болиулж чаддаггүй, болдоггүй, учир нь системийн шугам үүнийг төрөл бүрийн тоног төхөөрөмжүүдийн харилцан үйлдлүүдэд хэрэглэдэг. ACPI-г /boot/loader.conf файлд юм уу эсвэл loader(8) хүлээх мөрөнд `hint.acpi.0.disabled="1"` гэж тохируулан хааж болдог.

ACPI болон APM нь цуг байж болохгүй бөгөөд салангид хэрэглэгдэх ёстой. Сүүлд ачаалагдах драйвер нь хэрэв нөгөө нэгийг ажиллаж байгааг мэдвэл ажиллагаагаа дуусгавар болгодог.

ACPI нь acpiconf(8)-ийн `-s` туг болон `1-5` тохируулгын тусламжтайгаар системийг унтах горим шилжүүлэхэд хэрэглэгдэж болно. Ихэнх хэрэглэгчдэд зөвхөн `1` эсвэл `3` (RAM руу түр зогсоох) хэрэгтэй байдаг. `5` тохируулга нь дараах тушаалтай нэг ёсондоо адилыг гүйцэтгэнэ:

```
# halt -p
```

Бусад тохируулгууд sysctl(8)-ийн тусламжтай байдаг. Дэлгэрэнгүй мэдээллийн талаар acpi(4)

болон acpiconf(8) гарын авлагын хуудаснуудаас шалгана уу.

# 12.17. FreeBSD-ийн ACPI-г ашиглах нь ба дибаг хийх нь

ACPI нь төхөөрөмжүүдийг илрүүлэх, тэжээлийн ашиглалтыг удирдах болон урьд нь BIOS-оор удирдагддаг байсан төрөл бүрийн тоног төхөөрөмжид хандах стандартчилагдсан хандалтыг хангадаг цоо шинэ арга юм. Бүх системүүд дээр ACPI-г ажиллуулах тал дээр дэвшил хийгдсэн бөгөөд гэхдээ зарим эх хавтангуудын *ACPI Машины Хэл*ний (AML) байткод дахь алдаанууд, FreeBSD-ийн цөмийн дэд системүүдийн бүрэн бүтэн бус байдал болон Intel® ACPI-CA тайлбарлагч дахь алдаанууд илэрсээр байна.

Энэ баримт нь таныг FreeBSD-ийн ACPI дэмжигчдэд тусалж таны ажигласан асуудлуудын үндсэн учир шалтгааныг таних, дибаг хийх болон шийдлийг хөгжүүлэхэд туслах зорилготой юм. Үүнийг уншиж байгаад талархлаа илэрхийлэхийн ялдамд бид таны системийн асуудлуудыг шийдэж чадна гэдэгт найдаж байна.

## 12.17.1. Дибаг мэдээллийг илгээх нь

Асуудлыг илгээхээсээ өмнө та хамгийн сүүлийн үеийн BIOS-ийн хувилбар болон хэрэв байх юм бол суулгагдсан хянагчийн хамгийн сүүлийн firmware хувилбар ажиллуулж байгаа эсэхээ шалгаарай.

Асуудлыг шууд илгээхийг хүсэж байгаачууд дараах мэдээллийг freebsd-acpi@FreeBSD.org уруу илгээнэ үү:

- Системийн төрөл болон загварыг оролцуулан алдааг гаргаж байгаа зүйлийн хамтаар алдаатай ажиллагааг тайлбарласан мэдээлэл. Мөн хэрэв алдаа таны хувьд шинэ бол яг хэзээ гарч эхэлснийг аль болох тодорхой гаргаарай.
- `boot -v` ажилласны дараах dmesg(8)-ийн гаралтыг алдааг шалгаж байхад таны үүсгэсэн алдааны мэдээллүүдийн хамтаар.
- Хэрэв ACPI-г хаасан байхад асуудлыг шийдэж байвал тийм байх үе дэх `boot -v`-ийн гаралт.
- `sysctl hw.acpi`-ийн гаралт. Энэ нь таны систем ямар ямар боломжуудыг санал болгож байгааг мэдэх бас нэг сайн арга юм.
- Таны *ACPI Эх Хэл* (ASL) байх URL хаяг. ASL нь маш том байж болох учир шууд *битгий* жагсаалт уруу илгээгээрэй. Өөрийн ASL-ийн хуулбарыг энэ тушаалыг ашиглаж үүсгээрэй:

  ```
  # acpidump -dt > name-system.asl
  ```

  (Өөрийн нэвтрэх нэрийг *name*-ийн оронд болон үйлдвэрлэгч/загварыг *system*-ийн оронд солиорой. Жишээ нь: njl-FooCo6000.asl)

Ихэнх хөгжүүлэгчид FreeBSD-CURRENT захидлын жагсаалт үзэж байдаг, гэхдээ асуудлуудаа харагдуулахын тулд FreeBSD ACPI захидлын жагсаалт уруу илгээгээрэй. Бид бүгд хаа нэгтээ

өөр өөрийн үндсэн ажилтай учир тэвчээртэй байна уу. Хэрэв таны алдаа шууд илэрхий биш байх юм бол магадгүй бид таныг send-pr(1)-ийн тусламжтай PR илгээхийг асуух байх. PR оруулахдаа дээр хүссэний адил мэдээллээ оруулна уу. Энэ нь асуудлыг мөшгөж шийдвэрлэхэд бидэнд туслах юм. Бид PR-уудыг мэдээлэх механизмын зорилгоор биш байгаа асуудлуудыг санаж байх зорилгоор ашигладаг болохоор эхлээд FreeBSD ACPI захидлын жагсаалт уруу захидал илгээлгүйгээр PR битгий илгээгээрэй. Магадгүй таны асуудлыг урд нь өөр хэн нэгэн мэдээлсэн байж болох юм.

## 12.17.2. Оршил

ACPI нь ia32 (x86), ia64 (Itanium) болон amd64 (AMD) архитектуруудтай нийцтэй орчин үеийн бүх компьютерт байдаг. Бүрэн стандарт нь CPU-ны ажиллагааны удирдлага, тэжээлийн онгоцуудын хяналт, дулааны бүсүүд, төрөл бүрийн батарейний системүүд, суулгагдсан хянагчууд болон шугамын жагсаалт зэрэг олон боломжуудтай. Ихэнх системүүд нь бүрэн стандартыг бүгдийг хангасан шийдэлтэй байдаггүй. Жишээ нь зөөврийн компьютер хөргөх болон бас батарейний удирдлагын дэмжлэгтэй байхад ширээний систем зөвхөн шугамын жагсаалтын хэсгийн шийдлийг агуулсан байдаг. Зөөврийн компьютерууд нь бас өөр өөрийн ярвигтай асуудлуудыг агуулсан түр зогсоох болон үргэлжлүүлэх боломжуудыг агуулдаг.

ACPI-нийцтэй систем нь төрөл бүрийн хэсгүүдтэй байдаг. BIOS болон бичил схемийн үйлдвэрлэгчид APIC зураг (SMP-д ашиглагддаг), тохиргооны регистрүүд болон хялбар тохиргооны утгууд зэрэг зүйлсүүдийг заадаг төрөл бүрийн тогтмол хүснэгтүүдийг (өөрөөр хэлбэл FADT) санах ойд хангаж өгдөг. Мөн төхөөрөмжүүд болон аргуудын мод хэлбэрийн нэрийн талбарыг заадаг байткодын хүснэгтээр (*Differentiated System Description Table буюу Системийн Ялгаварласан Тайлбарын Хүснэгт* DSDT) бас хангадаг.

ACPI драйвер нь тогтмол хүснэгтүүдийг задлан ялгал хийх, байткодын тайлбарлагчийг шийдэх болон ACPI дэд системийн мэдээллийг хүлээн авахаар төхөөрөмжүүдийн драйверууд болон цөмийг өөрчлөх ёстой. FreeBSD-ийн хувьд Intel® нь Линукс болон NetBSD-тэй хуваалцан хэрэглэгддэг тайлбарлагчаар хангадаг. ACPI-CA эх кодын зам нь src/sys/contrib/dev/acpica. ACPI-CA-г FreeBSD дээр ажиллуулах тэр цавуу код нь src/sys/dev/acpica/Osd байршилд байдаг. Эцэст нь төрөл бүрийн ACPI төхөөрөмжүүдийн драйверууд src/sys/dev/acpica байршлаас олддог.

## 12.17.3. Нийтлэг асуудлууд

ACPI зөв ажиллахын тулд бүх хэсгүүд бас зөв ажилласан байх ёстой. Энд зарим нэг нийтлэг асуудлуудыг илэрч байгаа давтамжийн дарааллаар зарим нэг тойрон гарах замууд болон засваруудтайгаар нь дурдъя.

### 12.17.3.1. Хулганы асуудлууд

Зарим тохиолдолд түр зогсоох үйлдэл хийгдсэний дараа үргэлжлүүлэхэд хулганыг ажиллахгүй болгодог. Мэдэгдэж байгаа тойрон гарах арга зам нь `hint.psm.0.flags="0x3000"` мөрийг /boot/loader.conf файлд нэмэх явдал юм. Хэрэв энэ нь ажиллахгүй бол дээр тайлбарласны дагуу алдааны тайлан илгээхийг бодно уу.

#### 12.17.3.2. Suspend/Resume буюу Түр зогсоох/Үргэлжлүүлэх

ACPI нь RAM уруу түр зогсоох S1-S3 гэсэн гурван төлөвтэй (STR) бөгөөд диск уруу түр зогсоох S4 гэгддэг нэг төлөвтэй (STD). S5 нь "soft off буюу зөөлөн зогсоолт" бөгөөд тэжээлд залгагдсан боловч асаагдаагүй байх үеийн таны системийн жирийн төлөв юм. S4 нь хоёр тусдаа аргаар хийгдэх боломжтой. S4BIOS нь BIOS-ийн тусламжтайгаар диск уруу хийгдэх түр зогсоолт юм. S4OS нь бүхэлдээ үйлдлийн системээр хийгддэг.

Түр зогсоолттой холбоотой зүйлүүдийг `sysctl hw.acpi` тушаалаар шалгаж эхлээрэй. Энд Thinkpad-тай холбоотой үр дүнгүүд байна:

```
hw.acpi.supported_sleep_state: S3 S4 S5
hw.acpi.s4bios: 0
```

Энэ нь бид S3, S4OS болон S5-ийг шалгахад `acpiconf -s` тушаалыг ашиглаж болно гэсэн үг юм. Хэрэв `s4bios` нь нэг (1) байх юм бол бид S4OS-ийн оронд S4BIOS дэмжлэгтэй байх юм.

Түр зогсоолт/үргэлжлүүлэлтийг тест хийхдээ хэрэв дэмжигдсэн бол S1-ээс эхлээрэй. Энэ төлөв нь драйверийн дэмжлэг барагтаа л шаарддаггүй болохоор бараг л ажиллах болно. Хэн ч S2-ийг хийгээгүй байдаг бөгөөд танд энэ хэрэв байгаа бол энэ нь S1-тэй адил байна. Дараагийн оролдох зүйл нь S3 юм. Энэ нь хамгийн гүнзгий STR төлөв бөгөөд таны тоног төхөөрөмжийг дахин зөв эхлүүлэхийн тулд драйверийн ихээхэн дэмжлэг шаарддаг. Хэрэв үргэлжлүүлэх үед танд асуудлууд гарч байгаа бол FreeBSD ACPI захидлын жагсаалт жагсаалт уруу цахим захидал чөлөөтэй илгээгээрэй, гэхдээ илүү их тест хийлт, ажил шаардсан маш олон драйверууд/тоног төхөөрөмжүүд байдаг учир асуудал шийдэгдэхийг хүлээх хэрэггүй юм.

Түр зогсоолт/үргэлжлүүлэлттэй холбоотой түгээмэл асуудал бол олон төхөөрөмжийн драйверууд өөрсдийн эхлүүлэх програм, регистрүүд болон төхөөрөмжийн санах ойг зөв хадгалж, сэргээж, эсвэл дахин эхлүүлж чаддаггүй. Асуудлыг эхний удаа дибаг хийхийг оролдохдоо дараах тушаалыг ажиллуулж үзээрэй:

```
# sysctl debug.bootverbose=1
# sysctl debug.acpi.suspend_bounce=1
# acpiconf -s 3
```

Энэ тест нь S3 төлөв рүү жинхнээсээ оролгүйгээр бүх төхөөрөмжийн драйверуудын түр зогсолт/үргэлжлүүлэлтийн циклийг эмуляц хийдэг. Зарим тохиолдолд энэ аргыг ашиглан та асуудлыг хялбархнаар олж болно (жишээ нь эхлүүлэх програмын төлөв алдагдах, төхөөрөмжийн watchdog timeout болж дуусахгүй дахин оролдох). Систем нь жинхнээсээ S3 төлөвт орохгүй болохыг санаарай. Тэгэхээр төхөөрөмжүүд нь тэжээлээс салгагдахгүй бөгөөд түр зогсолт/үргэлжлүүлэлтийн арга тэдний хувьд байхгүй гэсэн олонхи нь зүгээр ажиллах болно. Харин жинхэнэ S3 төлвийн хувьд эсрэгээр байж магадгүй юм.

Хэцүү тохиолдлууд нэмэлт тоног төхөөрөмж шаарддаг, жишээ нь цуваа консолд зориулсан цуваа порт/кабель эсвэл dcons(4)-д зориулсан Firewire порт/кабел болон цөм дибаг хийх чадвар зэргийг дурдаж болно.

Асуудлыг тусгаарлахад туслахын тулд өөрийн цөмөөс аль болох олон драйверуудыг арилгаарай. Хэрэв энэ нь ажиллаж байвал та яг аль драйвер асуудалтай байгааг драйверуудыг амжилттүй ажиллах хүртэл ачаалан тодорхойлж болох юм. nvidia.ko, X11 дэлгэцийн драйверууд болон USB зэрэг хоёртын драйверууд нь ерөнхийдөө хамгийн их асуудлуудтай байдаг байхад Ethernet интерфэйсүүд ихэвчлэн зүгээр ажилладаг. Хэрэв та драйверуудыг зөв ачаалж/буулгаж чадаж байвал та тохирох тушаалуудыг /etc/rc.suspend болон /etc/rc.resume файлуудад хийж үүнийг автоматжуулж болно. Драйверийг буулгах болон ачаалахад зориулсан тайлбар болгосон жишээ байдаг. Хэрэв таны дэлгэц үргэлжлүүлэлт хийгдсэний дараа заваарсан бол `hw.acpi.reset_video`-г тэг (`0`) болгож үзээрэй. Хэрэв тусламж болохоор бол `hw.acpi.sleep_delay`-г арай урт эсвэл арай богино утгуудаар тохируулж үзээрэй.

Өөр нэг турших зүйл нь ACPI дэмжлэгтэй сүүлийн үеийн Линуксийн түгээлтийг ачаалан тэдний түр зогсоолт/үргэлжлүүлэлтийн дэмжлэгийг адил тоног төхөөрөмж дээр турших явдал юм. Хэрэв Линукс дээр ажиллаж байвал энэ нь FreeBSD-ийн драйверийн асуудал гэсэн үг бөгөөд яг аль драйвер асуудлыг үүсгэж байгааг олсноор асуудлыг засварлахад бидэнд тус болох болно. ACPI-ийг дэмжиж байдаг дэмжигчид нь өөр бусад драйверуудыг (өөрөөр хэлбэл дуу, ATA гэх мэт) ихэвчлэн дэмжин ажилладаггүй болохоор драйверийн асуудлыг мөшгөж хийгдсэн ажил бүр магадгүй эцсийн эцэст FreeBSD-CURRENT захидлын жагсаалт жагсаалт болон драйверийг дэмжигч уруу илгээгдэх хэрэгтэйг санаарай. Хэрэв та адал явдлыг эрж байгаа бол драйверийн үргэлжлүүлэлтийн функцын аль хэсэгт өлгөгдөж байгааг мөшгөхийн тулд зарим дибаг хийх printf(3)-үүдийг асуудалтай драйверт хийж эхлээрэй.

Эцэст нь ACPI-г хааж оронд нь APM-г нээж оролдоорой. Хэрэв түр зогсоолт/үргэлжлүүлэлт APM-тэй байхад ажиллаж байвал та APM-тэйгээ үлдэх нь ялангуяа хуучин тоног төхөөрөмжийн (2000 оноос өмнөх) хувьд бараг дээр байх бизээ. ACPI дэмжлэгийг зөв болгоход үйлдвэрлэгчдэд цаг хугацаа шаардах бөгөөд магадгүй хуучин тоног төхөөрөмжүүд нь ACPI-ийн хувьд BIOS-ийн асуудлуудтай ихэвчлэн байдаг.

#### 12.17.3.3. Систем өлгөгдөх (түр хугацаагаар эсвэл бүрмөсөн)

Ихэнх системийн өлгөгдлүүд нь гээгдсэн тасалдлууд эсвэл тасалдлын шуургын үр дүн юм. Бичил схемүүд нь ачаалахаас өмнө тасалдлуудыг BIOS хэрхэн тохируулдгаас болсон асуудлууд, APIC (MADT) хүснэгтийн зөв байдал болон *System Control Interrupt буюу Системийн Хянагч Тасалдлын* (SCI) чиглүүлэлт дээр тулгуурласан олон асуудлуудтай байдаг.

Тасалдлын шуургыг `vmstat -i` тушаалын гаралтаас `acpi0` бүхий мөрийг шалгаж гээгдсэн тасалдлуудаас ялгаж болно. Хэрэв тоологч секунд тутам хоёроор нэмэгдэж байвал та тасалдлын шуургатай байна. Хэрэв систем өлгөгдсөн юм шиг байвал DDB (консол дээр `CTRL` + `ALT` + `ESC`) уруу орж `show interrupts` гэж бичих хэрэгтэй.

Тасалдлын асуудлуудтай ажиллаж байхад таны хамгийн шилдэг итгэл найдвар бол loader.conf-д `hint.apic.0.disabled="1"` хэмээн зааж APIC дэмжлэгийг хаах явдал юм.

#### 12.17.3.4. Үймээнүүд

Үймээнүүд нь ACPI-ийн хувьд харьцангуй ховор байдаг бөгөөд засварлах нэн тэргүүн ээлжийн асуудал байдаг. Эхний алхам бол үймээнийг дахин гаргах (хэрэв боломжтой бол) алхмуудыг тусгаарлаж буцах мөрийг (backtrace) авах явдал юм. `options DDB` мөрийг нээж

сериал консол (crossref:serialcommsserialconsole-ddb,Цуваа шугамнаас DDB дибаг хийгч уруу орох]-г үзнэ үү) тохируулах эсвэл dump(8) хуваалтыг тохируулах зөвлөгөөг дагаарай. Та буцах мөрийг DDB дээр `tr`-р авч болно. Хэрэв та буцах мөрийг гараар бичих болбол мөр дэх хамгийн доодох тав (5) болон хамгийн дээдэх таван (5) мөрийг хамгийн багадаа бодоход аваарай.

Дараа нь асуудлыг тусгаарлахыг оролдож ACPI-г хааж ачаалж үзээрэй. Хэрэв энэ нь ажиллаж байвал `debug.acpi.disable`-ийн төрөл бүрийн утгуудыг хэрэглэж та ACPI дэд системийг тусгаарлаж болно. Зарим жишээнүүдийг acpi(4) гарын авлагын хуудаснаас үзнэ үү.

#### 12.17.3.5. Түр зогссоны дараа эсвэл унтраасны дараа систем дахин эхлэх

Эхлээд loader.conf(5) дээр `hw.acpi.disable_on_poweroff="0"` гэж тохируулаад үз. Энэ нь унтраах процессийн үед төрөл бүрийн үйл явцуудыг ACPI хаахыг болиулдаг. Энэ зорилгын нэгэн адил зарим системүүд энэ утгыг `1` (анхдагч) болгож тохируулахыг шаарддаг. Энэ нь түр зогсоолт эсвэл унтраалт хийгдсэний дараа аяндаа гарсан систем асаж эхлэх асуудлыг ихэвчлэн засварладаг.

#### 12.17.3.6. Бусад асуудлууд

Хэрэв танд ACPI-тай холбоотой бусад асуудлууд (суулгах станцтай ажиллах, төхөөрөмжүүд илрүүлэгдэхгүй гэх мэт) байвал тайлбарыг захидлын жагсаалт уруу бас илгээнэ үү; гэхдээ эдгээр асуудлуудын зарим нь ACPI дэд системийн дуусаагүй хэсгүүдтэй холбоотой байж болох бөгөөд тэдгээрийг шийдэж хийхэд нэлээн хугацаа зарцуулж болох юм. Тэвчээртэй байж бидний илгээж болох засваруудыг тест хийхэд бэлэн байгаарай.

### 12.17.4. ASL, `acpidump`, болон IASL

Хамгийн нийтлэг асуудал бол BIOS үйлдвэрлэгчдийн гаргасан буруу (эсвэл алдаатай!) байткод юм. Энэ нь ихэвчлэн дараах шиг цөмийн консол мэдээллүүдээр ил тод болдог:

```
ACPI-1287: *** Error: Method execution failed [\\_SB_.PCI0.LPC0.FIGD._STA] \\
(Node 0xc3f6d160), AE_NOT_FOUND
```

Ихэвчлэн та эдгээр асуудлуудыг өөрийн BIOS-ийг хамгийн сүүлийн хувилбар уруу шинэчилснээр шийдэж болно. Ихэнх консолын мэдээллүүд нь аюулгүй гэхдээ хэрэв танд батарейний төлөв ажиллахгүй гэх мэт өөр бусад асуудлууд байгаа бол тэдгээр мэдээллүүд нь AML-д байгаа асуудлуудыг хайж болох боломжийн газар нь юм. AML гэгддэг байткод нь ASL хэмээгддэг эх хэлээс хөрвүүлэгддэг. AML нь DSDT гэгддэг хүснэгтэд байдаг. Өөрийн ASL-ийн хуулбарыг авахын тулд acpidump(8)-ийг ашиглана. Та `-t` (тогтмол хүснэгтүүдийн агуулгуудыг үзүүлэх) болон `-d` (AML-ийг ASL уруу дизассембл хийх) тохируулгыг хоёуланг нь ашиглах хэрэгтэй. Синтаксын жишээг Дибаг Мэдээллийг Илгээх нь хэсгээс үзнэ үү.

Таны хийж болох хамгийн хялбар анхны шалгалт нь алдаануудыг шалгахын тулд өөрийн ASL-ийг хөрвүүлэх явдал юм. Анхааруулгуудыг ихэвчлэн орхиж болох боловч алдаанууд нь ACPI-г зөв ажиллуулахад гол төлөв саад болдог хорхойнууд байдаг. Өөрийн ASL-ийг дахин хөрвүүлэхдээ дараах тушаалыг ажиллуулна:

```
# iasl your.asl
```

### 12.17.5. Өөрийн ASL-г засварлах нь

Бидний эцсийн зорилго бол бараг хүн болгоны хувьд хэрэглэгчийн ямар ч оролцоогүйгээр ACPI-г ажиллуулах явдал юм. Гэхдээ өнөөг хүртэл бид BIOS үйлдвэрлэгчдийн гаргасан нийтлэг алдаануудад зориулан тойрон гарах арга замуудыг хөгжүүлсээр байгаа билээ. Microsoft®-ийн тайлбарлагч (acpi.sys болон acpiec.sys) нь стандартыг баримталж байгааг чанд шалгадаггүй бөгөөд BIOS-ийн олон үйлдвэрлэгчид ACPI-г зөвхөн Windows® дээр тест хийж өөрсдийн ASL-ийг хэзээ ч засдаггүй. Бид Microsoft®-ийн тайлбарлагчид зөвшөөрөгдсөн ямар стандартын бус ажиллагаа байгааг үргэлжлүүлэн нарийн таньж баримтжуулан хэрэглэгчдээр ASL-ийг хүчлэн засуулалгүйгээр FreeBSD ажиллаж чадахаар түүнийг хуулбарлах болно гэж найдаж байна. Тойрон гарах арга зам болгон биднийг энэ ажиллагааг танихад тусалж та ASL-ийг гараар засварлаж болно. Хэрэв таны хувьд энэ нь ажиллавал хуучин болон шинэ ASL-ийнхээ diff(1)-ийг илгээнэ үү, бид бололцоогоороо ACPI-CA дахь алдаатай ажиллагааг тойрон гарч ингэснээр хойшид таны засвар байнга хийгдэх шаардлагагүй болох юм.

Энд нийтлэг алдааны мэдээллүүд, тэдгээрийн шалтгаан болон хэрхэн засаж болох жагсаалтыг үзүүлэв:

#### 12.17.5.1. _OS хамаарлууд

Зарим AML нь ертөнц төрөл бүрийн Windows® хувилбаруудаас тогтдог гэж үздэг. Хэрэв танд байгаа асуудлыг засаж чадаж байвал та FreeBSD-г ямар нэг OS гэж харагдуулахаар хэлж өгч болно. Үүнийг хялбар аргаар дарж бичихийн тулд /boot/loader.conf-д `hw.acpi.osname="Windows 2001"` гэж эсвэл ASL дахь өөр бусад адил мөрүүдийг тохируулж өгнө.

#### 12.17.5.2. Буцах мэдээллүүд байхгүй бол

Зарим аргууд нь стандартын дагуу шууд утга буцаадаггүй. ACPI-CA нь үүнтэй ажиллаж чадахгүй байхад FreeBSD үүнийг далдаар утга буцаалгах боломжийг олгодог тойрон гарах арга замтай байдаг. Хэрэв та утга буцаагдах ёстойг мэдэж байвал шаардлагатай газар нь Return буюу Буцах мэдээллүүдийг шууд нэмж болно. ASL-ийг `iasl` тушаалаар хүчээр хөрвүүлэхдээ `-f` тугийг ашиглана.

#### 12.17.5.3. Анхдагч AML-ийг дарж өөрчлөх нь

your.asl-ийг өөрчилсний дараа үүнийг та хөрвүүлэхдээ:

```
# iasl your.asl
```

Хөрвүүлэх явцад алдаанууд байсан ч гэсэн та `-f` тугийг нэмж AML-ийг хүчээр үүсгэж болно. Зарим алдаануудыг (өөрөөр хэлбэл Буцах мэдээллүүд байхгүй гэх мэт) тайлбарлагчийн тусламжтайгаар автоматаар тойрон гардгийг санаарай.

DSDT.aml нь `iasl`-ийн анхдагч гаралт файлын нэр юм. Та өөрийн BIOS-ийн алдаатай

хуулбарын (флэш санах ойд байсаар байгаа) оронд /boot/loader.conf-ийг дараах байдлаар засварлан үүнийг ачаалж болно:

```
acpi_dsdt_load="YES"
acpi_dsdt_name="/boot/DSDT.aml"
```

Өөрийн DSDT.aml файлын хуулбарыг /boot сан уруу хуулах хэрэгтэй.

### 12.17.6. ACPI-аас дибаг мэдээлэл гаргаж авах нь

ACPI драйвер нь маш уян хатан дибаг хийх боломжтой. Энэ нь дэд системүүдийн олонлог болон харуулах түвшинг зааж өгөхийг танд зөвшөөрдөг. Таны дибаг хийхийг хүсэж байгаа дэд системүүд нь "давхаргууд" болж заагдсан байдаг бөгөөд ACPI-CA хэсгүүд (ACPI_ALL_COMPONENTS) болон ACPI тоног төхөөрөмжийн дэмжлэг (ACPI_ALL_DRIVERS) болж задардаг. Дибаг гаралтын харуулалт нь "үе"ээр заагддаг бөгөөд ACPI_LV_ERROR (зөвхөн алдаануудыг хэлдэг) тогтмолоос ACPI_LV_VERBOSE (бүгд) хүртэл байдаг. "Үе" нь олон тохируулгуудыг нэг удаа зайгаар зааглан тохируулж болох бит баг (bitmask) юм. Хэрэв энэ нь маш урт тэгээд консолын мэдээллийн буферийг арилган шинэчилж байвал та практик дээр гаралтыг бүртгэх сериал консолыг ашиглахыг хүсэж болох юм. Бие даасан давхаргууд болон түвшингүүдийн бүрэн жагсаалт acpi(4) гарын авлагын хуудсанд байдаг.

Дибаг гаралт анхдагчаар идэвхжүүлэгдээгүй байдаг. Идэвхтэй болгохын тулд ACPI хэрэв цөмд хөрвүүлэгдсэн бол `options ACPI_DEBUG` мөрийг өөрийн цөмийн тохиргооны файлд нэмэх хэрэгтэй. Нийтэд нь идэвхтэй болгохын тулд /etc/make.conf-д `ACPI_DEBUG=1` мөрийг нэмж болно. Хэрэв энэ нь модуль бол та өөрийн acpi.ko модулийг дараах маягаар дахин хөрвүүлж болно:

```
# cd /sys/modules/acpi/acpi
&& make clean &&
make ACPI_DEBUG=1
```

acpi.ko-г /boot/kernel-д суулгаад өөрийн хүссэн давхарга болон түвшинг loader.conf-д нэмнэ. Энэ жишээ нь ACPI-CA-ийн бүх хэсгүүд болон бүх ACPI тоног төхөөрөмжийн драйверуудад (CPU, LID, гэх мэт.) зориулан дибаг мэдээллүүдийг идэвхжүүлдэг. Энэ нь зөвхөн алдааны мэдээллүүдийг хамгийн багаар гаргаж харуулна.

```
debug.acpi.layer="ACPI_ALL_COMPONENTS ACPI_ALL_DRIVERS"
debug.acpi.level="ACPI_LV_ERROR"
```

Хэрэв таны хүссэн мэдээлэл онцгой үйл явцаар эхэлсэн бол (түр зогсоолт болон үргэлжлүүлэлт гэж бодъё) та loader.conf-ийн өөрчлөлтүүдийг орхиж оронд нь `sysctl`-ийг ашиглан давхарга болон түвшинг ачаалсны дараа зааж онцгой үйл явцад зориулан өөрийн системийг бэлдэж болно. `sysctl`-ууд нь loader.conf дахь тохируулгуудын адилаар нэрлэгддэг.

### 12.17.7. Лавлагаанууд

ACPI-ийн талаар дэлгэрэнгүй мэдээллийг дараах байршлуудаас олж болно:

- FreeBSD ACPI захидлын жагсаалт
- ACPI Захидлын Жагсаалтын Архивууд http://lists.freebsd.org/pipermail/freebsd-acpi/
- Хуучин ACPI Захидлын Жагсаалтын Архивууд http://home.jp.FreeBSD.org/mail-list/acpi-jp/
- ACPI Тодорхойлолт
- FreeBSD Гарын авлагын хуудаснууд: acpi(4), acpi_thermal(4), acpidump(8), iasl(8), acpidb(8)
- DSDT дибаг эх үүсвэр. (Compaq-ийг жишээ болгон хэрэглэсэн боловч ерөнхийдөө хэрэгтэй.)

# Chapter 13. FreeBSD-ийн Ачаалах процесс

## 13.1. Ерөнхий агуулга

Компьютерийг эхлүүлж үйлдлийн системийг ачаалах процесс нь "эхлүүлэгч процесс (bootstrap process)" буюу "ачаалах" гэж хэлэгддэг. FreeBSD-ийн ачаалах процесс нь танд системийг эхлүүлэх үед компьютер дээр суусан өөр үйлдлийн системүүд эсвэл адил үйлдлийн системийн өөр хувилбарууд эсвэл суусан өөр цөмийг сонгохыг зөвшөөрч юу хийгдэхийг өөрчлөх боломжийг бүрдүүлж нэлээн уян хатан чанарыг хангаж өгдөг.

Энэхүү бүлэг нь тохируулж болох тохиргооны тохируулгуудыг тайлбарласан. Үүнд FreeBSD цөм эхлэх, төхөөрөмжүүдийг шалгах болон init(8) эхлэх хүртэлх болж байгаа FreeBSD-ийн ачаалах процессийг өөрчилж болох бүх үйл явдал багтана. Энэ нь текстийн өнгө тод цагаанаас саарал уруу өөрчлөгдөж байх үед болдог.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD-ийн эхлүүлэгч системийн хэсгүүд болон тэдгээр нь хэрхэн харилцан үйлдэл хийдэг талаар.
- Ачаалах процессийг хянахын тулд FreeBSD-ийн эхлүүлэгч дэх хэсгүүдэд өгч болох тохируулгуудын талаар.
- device.hints(5)-ийн үндсүүд.

Энэ бүлэг нь зөвхөн Интел x86 системүүд дээр ажиллаж байгаа FreeBSD-ийн ачаалах процессийг тайлбарлана.

## 13.2. Ачаалалтын асуудал

Компьютерийг асааж үйлдлийн системийг эхлүүлэх нь сонирхолтой мухардалд оруулдаг. Тодорхойлолтоор бол компьютер нь үйлдлийн систем эхлэх хүртэл юу хийхээ мэддэггүй. Үүнд програмуудыг дискнээс ажиллуулах ордог. Тэгэхээр хэрэв компьютер үйлдлийн системгүйгээр програмыг дискнээс ажиллуулж чаддаггүй тэгээд бас үйлдлийн системийн програмууд диск дээр байдаг гэхээр үйлдлийн систем хэрхэн эхэлдэг болж таарах вэ?

Энэ асуудал нь Мянгуужингийн адал явдал (The Adventures of Baron Munchausen) номонд гардагтай төстэй юм. Гол баатар маань нүх уруу унаад өөрийнхөө гутлын оосроос барьж өөрийгөө өргөн татаж гаргадаг. Тооцоололтын эриний эхэн үед *bootstrap буюу эхлүүлэгч (эхлүүлэлт)* гэдэг ойлголт нь үйлдлийн системийг ачаалахад ашиглагддаг арга замд хэрэглэгддэг байсан бөгөөд "booting буюу ачаалах" гэж богиноссон юм.

x86 тоног төхөөрөмж дээр Үндсэн Ороолт/Гаралтын Систем (BIOS) нь үйлдлийн системийг ачаалах үүрэгтэй. Үүнийг хийхийн тулд BIOS хатуу диск дээрээс Master Boot Record (MBR) буюу Мастер Ачаалах Бичлэгийг хайдаг бөгөөд энэ нь дискний онцгой газар байрлах ёстой. BIOS нь MBR-г ачаалж ажиллуулах хангалттай мэдээлэлтэй бөгөөд дараа нь MBR үйлдлийн системийг ачаалахтай холбоотой бусад үйлдлүүдийг магадгүй BIOS-ийн тусламжтайгаар зохицуулна гэж тооцдог.

MBR доторх код нь ялангуяа хэрэглэгчтэй ажиллахдаа *bootmanager буюу ачаалагч менежер* хэмээгддэг. Энэ тохиолдолд ачаалагч менежер нь ихэвчлэн дискний эхний *зам* эсвэл OS-ийн зарим файлын систем дээр илүү кодтой байдаг. (Ачаалагч менежер нь заримдаа *ачаалагч дуудагч* гэгддэг, гэхдээ FreeBSD энэ нэрийг ачаалалтын сүүлийн шатуудад хэрэглэдэг.) Алдартай ачаалагч менежерүүдэд boot0 (Boot Easy гэгддэг, FreeBSD-ийн стандарт ачаалагч менежер), Grub, GAG, болон LILO ордог. (Зөвхөн boot0 MBR-д багтдаг.)

Хэрэв зөвхөн нэг үйлдлийн систем суулгагдсан бол стандарт PC MBR хангалттай. Энэ MBR нь диск дээрээс эхний ачаалагдах (идэвхтэй) зүсмэлийг хайгаад дараа нь үйлдлийн системийн үлдсэнийг дуудахын тулд тэр зүсмэл дээрх кодыг ажиллуулдаг. Анхдагчаар fdisk(8)-ээр суулгагддаг MBR нь тийм MBR бөгөөд /boot/mbr дээр тулгуурладаг.

Хэрэв олон үйлдлийн систем суулгасан бол үйлдлийн системүүдийн жагсаалтыг харуулж аль нэгээс нь ачаалахыг сонгож болдог өөр ачаалагч менежер суулгаж болно. Эдгээрээс хоёр нь дараагийн дэд хэсэгт ярилцагдана.

FreeBSD-ийн эхлүүлэгч системийн үлдсэн хэсэг нь гурван шатанд хуваагддаг. Эхний шат нь компьютерийг тусгай төлөв уруу оруулахыг хангалттай мэдэж хоёр дахь шатыг ажиллуулах MBR-р ажиллуулагддаг. Хоёр дахь шат нь гурав дахь шатыг ажиллуулахаас өмнө арай илүүг хийж чаддаг. Гурав дахь шат нь үйлдлийн системийг дуудах үйлдлийг дуусгадаг. Энэхүү гурван шатанд ажил нь хуваарилагдсан байдаг бөгөөд учир нь PC стандартууд эхний болон хоёрдугаар шатуудад ажиллуулж болох програмуудын хэмжээнүүдэд хязгаарлалт тавьдаг юм. Үйлдлүүдийг цугт нь гинжлэн холбох нь FreeBSD-д илүү уян хатан дуудагчийг бий болгодог.

Дараа нь цөм ачаалж төхөөрөмжүүдийг шалгаж эхлэн ашиглахад зориулж эхлүүлдэг. Цөмийн ачаалах процесс дууссаны дараа цөм хяналтыг init(8) хэрэглэгчийн процесс руу дамжуулж дараа нь дискнүүд хэрэглэгдэж болох төлөвт байгаа эсэхийг шалгадаг. init(8) дараа нь файлын системүүдийг холбон сүлжээнд холбогдох сүлжээний картыг тохируулж FreeBSD систем эхлэх үед ихэвчлэн ажилладаг бүх процессуудыг эхлүүлэх хэрэглэгчийн түвшний эх үүсвэрийн тохиргоог эхлүүлдэг.

# 13.3. Ачаалагч Менежер болон Ачаалалтын шатууд

## 13.3.1. Ачаалагч Менежер

MBR эсвэл ачаалагч менежер дэх код нь заримдаа ачаалах процессийн *тэг (0) шат* гэж нэрлэгддэг. Энэ хэсэг нь хоёр ачаалагч менежерийг авч үзнэ: boot0 болон LILO.

**boot0 Ачаалагч Менежер:** FreeBSD-ийн суулгагч эсвэл boot0cfg(8)-р суулгагдсан MBR /boot/boot0 дээр тулгуурладаг. boot0-н хэмжээ болон боломж нь зүсмэлийн хүснэгт болон MBR-ийн төгсгөл дэх `0x55AA` танигчаас болоод 446 байт байдаг. Хэрэв boot0 болон олон үйлдлийн системийг суулгасан бол ачаалах үед доор дурдсантай төсөөтэй дэлгэцийг харах болно:

*Жишээ 10. boot0 дэлгэцийн агшин*

```
F1 DOS
```

```
F2 FreeBSD

Default: F2
```

Бусад үйлдлийн системүүд ялангуяа Windows® нь FreeBSD-ийн дараа суусан бол байгаа MBR-ийг өөрийнхөөрөө дарж бичдэг. Хэрэв энэ тохиолдвол эсвэл та байгаа MBR-аа FreeBSD-ийн MBR-аар солихыг хүсвэл дараах тушаалыг ашиглана:

```
# fdisk -B -b /boot/boot0 device
```

Дээр бичигдсэн *device* нь эхний IDE дискний хувьд ad0, хоёр дахь IDE хянагч дээрх эхний IDE дискний хувьд ad2, эхний SCSI дискний хувьд da0 гэх зэрэг ачаалах диск байх юм. MBR-ийн өөрчлөн тохируулсан тохиргоог хүсэж байвал boot0cfg(8)-ийг ашигла.

**LILO Ачаалагч Менежер:** FreeBSD-г бас ачаалах энэ ачаалагч менежерийг суулгахын тулд Линукс эхлүүлээд дараах тохиргоог /etc/lilo.conf тохиргооны файлд нэмээрэй:

```
other=/dev/hdXY
table=/dev/hdX
loader=/boot/chain.b
label=FreeBSD
```

Линуксийн тодорхойлогчдыг ашиглан *X*-г Линуксийн дискний үсгээр, *Y*-г Линуксийн анхдагч хуваалтын дугаараар сольж FreeBSD-ийн анхдагч хуваалт болон дискийг зааж өгнө. SCSI диск ашиглаж байвал */dev/hd*-г */dev/sd* болгон өөрчлөх хэрэгтэй. Хэрэв хоёр үйлдлийн систем хоёулаа нэг диск дээр байвал `loader=/boot/chain.b` мөр орхигдож болно. Дараа нь `/sbin/lilo -v` тушаалыг ажиллуулж шинэ өөрчлөлтийг системд оруулна. Зөв эсэхийг шалгахын тулд дэлгэц дээр гаргах мэдээллүүдийг хянан шалгах хэрэгтэй.

### 13.3.2. Нэгдүгээр шат /boot/boot1 болон Хоёрдугаар шат /boot/boot2

Ерөнхий төсөөллөөр бол эхний болон хоёр дахь шатууд нь дискний нэг талбар дахь нэг л програмын хэсэг юм. Зайнаас хамаараад тэдгээрийг хоёр хэсэг болгон хуваасан, гэхдээ үргэлж цуглаа суудаг. Тэдгээр нь нэгдсэн /boot/boot файлаас суулгагчаар эсвэл bsdlabel-р хуулагддаг.

Тэд файлын системүүдээс гадна ачаалах зүсмэлийн эхний зам дээр эхний секторoос эхлээд байрладаг. Энд л boot0 эсвэл өөр аль нэг ачаалагч менежер ачаалах процессийг үргэлжлүүлэх програмыг олно гэдэгт найдаж байдаг. Ашиглагдах секторуудын тоо /boot/boot -ийн хэмжээнээс амархнаар тодорхойлогдоно.

Зөвхөн 512 байт хэмжээтэй байдаг болохоор boot1 нь их энгийн хялбар бөгөөд boot2-ийг олж ажиллуулах, зүсмэлийн тухай мэдээллийг хадгалах, FreeBSD-ийн *bsdlabel*-ийн тухай хангалттай мэдээлэлтэй байдаг.

boot2 нь арай илүү төвөгтэй бөгөөд файлуудыг олоход хангалттай FreeBSD-ийн файлын

системийг ойлгож цөм эсвэл дуудагчийг ажиллуулахыг сонгох маш энгийн интерфэйсээр хангаж чаддаг.

Дуудагч нь хамаагүй илүү төвөгтэй бөгөөд boot2-р ажилладаг ачаалалтын тохиргоотой байдаг.

*Жишээ 11. boot2 дэлгэцийн агшин*

```
>> FreeBSD/i386 BOOT
Default: 0:ad(0,a)/boot/loader
boot:
```

Суулгасан boot1 болон boot2 файлуудыг солиход bsdlabel(8)-ийг ашиглаж болно:

```
# bsdlabel -B diskslice
```

Дээр бичигдсэн *diskslice* нь эхний IDE диск дээрх эхний зүсмэлийн хувьд ad0s1 гэх мэтээр ачаалах диск болон зүсмэл юм.

*Аюултайгаар Зориулагдсан Горим*

Хэрэв ad0 гэх мэтээр дискний нэрийг ашиглавал bsdlabel(8) нь зүсмэлүүдгүй аюултайгаар зориулагдсан диск үүсгэх болно. Энэ нь мэдээж хүсээгүй зүйл болохоор bsdlabel(8) тушаалыг `Return` дарж ажиллуулахаасаа өмнө түүнд дамжуулах *diskslice*-г дахин шалгаарай.

## 13.3.3. Гуравдугаар шат /boot/loader

Дуудагч нь гурван шаттай эхлүүлэгчийн төгсгөлийн шат бөгөөд файлын систем дээр гол төлөв /boot/loader гэж байрладаг.

Дуудагч нь илүү цогц тушаалын цуглуулга бүхий илүү хүчирхэг хөрвүүлэгчээр дэмжигдсэн тушаалын цуглуулга ашиглан тохиргоо хийхэд интерактив арга байхаар зориулагдсан.

### 13.3.3.1. Дуудагч програмын урсгал

Эхлүүлэх явцад дуудагч нь консол болон дискнүүдийн хувьд шалгаж аль дискнээс ачаалж байгаагаа тогтоодог. Энэ нь хувьсагчуудыг шаардлагын дагуу тохируулах бөгөөд скрипт эсвэл лавлаж хариулах зарчмаар хэрэглэгчийн тушаалууд дамждаг тайлбарлагч эхэлдэг.

Үүний дараа дуудагч нь хувьсагчуудын боломжийн анхдагчуудыг тохируулдаг /boot/defaults/loader.conf дотор байгааг анхдагчаар уншдаг /boot/loader.rc файлыг унших бөгөөд тэдгээр хувьсагчуудад хийх локал өөрчлөлтүүдэд зориулсан /boot/loader.conf файлыг мөн уншдаг. Дараа нь loader.rc аль модулиуд болон цөмийг сонгосон тэдгээрийг дуудан эдгээр хувьсагчуудын дагуу ажилладаг.

Эцэст нь анхдагчаар дуудагч нь 10 секунд гаргаж товч дарахыг хүлээж тасалдуулаагүй бол

цөмийг ачаалдаг. Хэрэв тасалдуулбал тушаалын цуглуулгыг ойлгодог тушаал хүлээх мөрийг хэрэглэгчид өгөх бөгөөд үүнийг ашиглан хэрэглэгч хувьсагчуудыг тааруулах, бүх модулиудыг буулгаж болиулах, модулиуд дуудах болон тэгээд төгсгөлд нь ачаалах эсвэл дахин ачаалж болох юм.

#### 13.3.3.2. Дуудагчид багтсан тушаалууд

Эдгээр нь хамгийн ихээр ашиглагддаг дуудагчийн тушаалууд юм. Байгаа бүх тушаалуудын тухай бүрэн хэлэлцүүлгийг loader(8)-с үзнэ үү.

**autoboot *seconds***

Секундээр өгөгдсөн хугацаанд тасалдаагүй бол цөмийг ачаалахаар үргэлжлүүлдэг. Энэ нь тоолуур харуулах бөгөөд анхдагч хугацаа нь 10 секунд байна.

**boot [-options] [kernelname]**

Өгөгдсөн тохируулгууд эсвэл цөмийн нэртэйгээр цөмийг нэн даруй ачаалахаар үргэлжлүүлнэ. *unload* тушаалыг ажиллуулсны дараа зөвхөн тушаалын мөрөөс цөмийн нэрийг өгч болох бөгөөд хэрэв ингэхгүй бол өмнө нь дуудагдсан цөмийг ашиглах болно.

**boot-conf**

Ихэвчлэн `kernel` байх өгөгдсөн хувьсагчууд дээр үндэслэн модулиудын автомат тохиргоогоор орно. Зарим хувьсагчуудыг өөрчлөхөөсөө өмнө `unload`-г эхэлж ашиглавал энэ нь зөвхөн ач холбогдолтой байдаг.

**help [topic]**

/boot/loader.help файлаас тусламжийн мэдээллүүдийг үзүүлнэ. Хэрэв өгөгдсөн сэдэв нь `индекс` бол байгаа сэдвүүдийн жагсаалтыг үзүүлнэ.

**include *filename* ...**

Өгөгдсөн файлын нэртэй файлыг процесс хийнэ. Файл уншигдаж мөр мөрөөр хөрвүүлэгдэнэ. Алдаа гарвал include буюу оруулах тушаалыг нэн даруй зогсооно.

**load [-t type] *filename***

Цөм, цөмийн модуль, эсвэл өгөгдсөн төрлийн файлыг нэртэй нь дуудна. *filename*-н дараах дурын нэмэлт өгөгдлүүд нь файлд дамжуулагдана.

**ls [-l path]**

Өгөгдсөн зам эсвэл зам өгөгдөөгүй бол root сан дахь файлуудын жагсаалтыг харуулна. Хэрэв `-l` өгөгдсөн бол файлын хэмжээнүүдийг бас харуулдаг.

**lsdev [-v]**

Модулиуд магадгүй дуудагдаж болох бүх төхөөрөмжүүдийг жагсаана. Хэрэв `-v` өгөгдсөн бол илүү дэлгэрэнгүй мэдээлэл хэвлэгдэнэ.

**lsmod [-v]**

Дуудагдсан модулиудыг харуулна. Хэрэв `-v` өгөгдсөн бол илүү дэлгэрэнгүй мэдээллийг үзүүлнэ.

**more *filename***

`LINES` болгоныг харуулан түр зогсож өгөгдсөн файлуудыг үзүүлнэ.

**reboot**

Системийг нэн даруй дахин ачаална.

**set *variable***

Дуудагчийн орчны хувьсагчуудыг тохируулна.

**unload**

Дуудагдсан модулиудыг арилгана.

#### 13.3.3.3. Дуудагчийн жишээнүүд

Дуудагчийн хэрэглээний практик жишээнүүдийг энд дурдав:

- өөрийн ердийн цөмийг ганц-хэрэглэгчийн горимд ачаалахдаа:

```
boot -s
```

- Ердийн цөм болон модулиудыг буулган болиулж дараа нь хуучин эсвэл өөр цөмийг дуудахдаа:

```
unload
load kernel.old
```

kernel.GENERIC-г суулгацын дисктэй цуг ирсэн анхдагч цөмийг дуудахдаа ашиглаж болох бөгөөд эсвэл цөмийг шинэчилж тохируулахаасаа өмнө эсвэл системийн шинэчлэл хийхээсээ өмнө суулгасан цөмөө дуудахдаа kernel.old-г ашиглаж болно.

Ердийн модулиудыг өөр цөмийн хамт дуудахдаа доор дурдсаныг ашигла:

```
unload
set kernel="kernel.old"
boot-conf
```

- Цөмийн автомат тохиргооны скриптийг дуудахдаа:

```
load -t userconfig_script /boot/kernel.conf
```

#### 13.3.3.4. Ачаалах үеийн дэлгэцийн зураг

Ачаалах үеийн дэлгэцийн зураг нь өөр ачаалах дэлгэцийг бий болгодог. Энэхүү дэлгэц нь тушаалын мөр эсвэл график нэвтрэлт хүлээх цонхыг харуулахаас өмнө ачаалах үеийн шалган илрүүлэх мэдэгдлүүд болон үйлчилгээний эхлэх мэдэгдлүүдийг харуулдаггүй.

FreeBSD дээр үндсэн хоёр орчин байдаг. Эхнийх нь хуучны анхдагч виртуал консол тушаалын мөрний орчин юм. Систем ачаалж дууссаны дараа консолын нэвтрэлт хүлээх мөр харуулагддаг. Хоёр дахь орчин нь Xorg графикийн орчин юм. График дэлгэцийн менежер болон график нэвтрэх менежерийг суулгаж тохируулах талаарх дэлгэрэнгүй мэдээллийг зохих бүлгээс үзнэ үү.

#### 13.3.3.4.1. Ачаалах үеийн дэлгэцийн функц

Ачаалах үеийн дэлгэцийн функц нь зөвхөн 256 өнгийн битмап (.bmp), ZSoft PCX (.pcx) эсвэл TheDraw (.bin) хэлбэрүүдийг дэмждэг. Зураг файлууд нь стандарт VGA адаптер дээр ажиллахын тулд 320-ийг харьцах 200 пикселийн нягтралтай байх ёстой.

1024-ийг харьцах 768 пикселийн максимум нягтрал хүртэлх илүү том зургийг ашиглахын тулд VESA модулийг систем ачаалах үед дуудах хэрэгтэй. өөрчлөн тохируулсан цөмийн хувьд `VESA` цөмийн тохиргоог нэмэх хэрэгтэй. VESA дэмжлэгийг дуудсанаар бүх дэлгэцийг бүрхэх ачаалах үеийн дэлгэцийн зургийг харуулах боломжийг хэрэглэгчид олгодог.

Ачаалах үеийн дэлгэцийг харуулагдаж байх үед гарын аль ч товчлуурыг дарж болиулж болно.

Ачаалах үеийн дэлгэцийн зураг нь анхдагчаар гаднах дэлгэц амраагч болдог. Ашиглаагүй тодорхой хугацааны дараа ачаалах үеийн дэлгэцийн зураг гарч тодоос бүр харанхуй болон дахин дахин солигдон эргэлдэх болно. Дэлгэцийн зургийн тохиргоог /etc/rc.conf файлд `saver=` мөрийг нэмэн өөрчилж болно. Хэд хэдэн сонгож болох дэлгэц амраагч байдаг бөгөөд splash(4) гарын авлагын хуудаснаас олж болно. `saver=` тохиргоо нь зөвхөн виртуал консолд хамаатай гэдгийг санаарай. Энэ нь график дэлгэцийн менежерүүдэд ямар ч нөлөөгүй болно.

Ачаалах үеийн дэлгэцийг идэвхжүүлсэн ч гэсэн ачаалагчийн тохиргооны цэс болон тушаал хүлээн секунд гүйх үеийн мөр зэрэг ачаалагч дуудагчийн мэдэгдлүүд нь ачаалах үед харуулагдсан хэвээр байх болно.

Жишээ ачаалах үеийн дэлгэцийн файлуудыг http://artwork.freebsdgr.org хаяг дахь галерейгаас татаж авч болно. sysutils/bsd-splash-changer портыг суулгаснаар ачаалах үеийн дэлгэцийн зургийг ачаалах болгонд санамсаргүйгээр цуглуулгаас сонгож харуулах боломжтой болно.

#### 13.3.3.4.2. Ачаалах үеийн дэлгэцийн функцыг идэвхжүүлэх

Ачаалах үеийн дэлгэцийн .bmp, .pcx эсвэл .bin файлыг root хуваалт дээр жишээ нь /boot санд байрлуулах ёстой.

Ачаалагчийн анхдагч дэлгэцийн 256 өнгө, 320-ийг харьцах 200 пиксел юм уу эсвэл түүнээс бага нягтралын хувьд /boot/loader.conf файл дараахийг агуулсан байхаар засварлах хэрэгтэй:

```
splash_bmp_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.bmp"
```

1024-ийг харьцах 768 пикселийн максимум хүртэлх илүү том видео нягтралуудын хувьд

/boot/loader.conf файл дараахийг агуулсан байхаар засварлах хэрэгтэй:

```
vesa_load="YES"
splash_bmp_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.bmp"
```

Дээрх нь /boot/splash.bmp-ийг ачаалах үеийн дэлгэцдээ ашиглахыг зааж өгч байна. PCX файл ашиглахын тулд дараах илэрхийллийг `vesa_load="YES"` мөрийн хамтаар нягтралаас хамааруулан ашиглана.

```
splash_pcx_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.pcx"
```

FreeBSD 8.3 хувилбараас эхлээд TheDraw хэлбэрийн ascii зураг ашиглах өөр нэг сонголт бий.

```
splash_txt="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.bin"
```

Файлын нэр нь дээрх жишээ дээрх шиг заавал "splash" гэдгээр хязгаарлагдахгүй. splash_640x400.bmp эсвэл bluewave.pcx зэрэг дэмжигдсэн төрлийн байхад болох юм.

loader.conf-ийн бусад сонирхолтой тохиргоонуудыг дурдвал:

`beastie_disable="YES"`

: Энэ нь ачаалагчийн тохиргооны цэсийг харуулахгүй болгоно. Гэхдээ секунд гүйсэн тушаал хүлээх мөрийг харуусан хэвээр байх болно. Ачаалагчийн тохиргооны цэсийг хаасан ч гэсэн секунд гүйсэн тушаал хүлээх мөрөнд сонгосон сонголтын дагуу систем ачаалах болно.

`loader_logo="beastie"`

: Энэ нь ачаалагчийн тохиргооны цэсний баруун талд харуулагдах "FreeBSD" гэсэн анхдагч үгийг өнгөт чөтгөрийн логогоор солих болно.

Дэлгэрэнгүй мэдээллийг splash(4), loader.conf(5), болон vga(4) гарын авлагын хуудаснуудаас үзнэ үү.

# 13.4. Ачаалах үе дэх цөмийн харилцан үйлдэл

Цөм анхдагчаар дуудагч эсвэл дуудагчийг алгасан boot2-ийн тусламжтай дуудагдсаны дараа өөрийн ачаалалтын тугууд байгаа бол тэдгээрийг шалгаж өөрийн ажиллагааг шаардлагатай бол тохируулдаг.

### 13.4.1. Цөмийн ачаалалтын тугууд

Энд илүү нийтлэг ачаалалтын тугуудыг дурдав:

`-a`

Цөмийг эхлүүлэх явцад root файлын систем болгон холбох төхөөрөмжийг асуух.

`-C`

CDROM-с ачаалах.

`-c`

Ачаалалтын үеийн цөмийн тохируулга UserConfig-г ажиллуулах

`-s`

Нэг(ганц)-хэрэглэгчийн горим уруу ачаалах

`-v`

Цөмийн эхлэх үед илүү дэлгэрэнгүй байх

Бусад ачаалалтын тугуудын талаарх илүү мэдээллийг boot(8)-с уншина уу.

## 13.5. Төхөөрөмжийн Сануулгууд

Системийн эхний эхлүүлэлтийн үеэр ачаалагч loader(8) нь device.hints(5) файлыг уншдаг. Энэ файл нь заримдаа "device hints буюу төхөөрөмжийн сануулгууд" ч гэгддэг цөмийн ачаалах мэдээлэл хувьсагчуудыг хадгалдаг. Эдгээр "device hints буюу төхөөрөмжийн сануулгууд" нь төхөөрөмжийг тохируулах зориулалтаар төхөөрөмжийн драйверуудад ашиглагддаг.

Төхөөрөмжийн сануулгууд нь бас 3 дахь шатны ачаалагч дуудагчид бас заагдаж өгч болдог. Хувьсагчуудыг `set` тушаалыг ашиглан нэмж, `unset` тушаалаар хасаж `show` тушаалаар үзэж болно. /boot/device.hints файлд тохируулагдсан хувьсагчууд бас өөрчлөгдөж дарагдан бичигдэж болно. Ачаалагч дуудагчид оруулж өгсөн төхөөрөмжүүдийн сануулгууд нь тогтмол биш бөгөөд дараа дахин ачаалах үед мартагддаг.

Систем ачаалагдсаны дараа kenv(1) тушаалыг ашиглаж бүх хувьсагчуудыг харуулж болно.

/boot/device.hints файлын синтакс нь мөр бүрт нэг хувьсагч байх бөгөөд чагт "#"-г тайлбар тэмдэглэгчээр ашигладаг. Мөрүүдийг дараах байдлаар бүтээдэг:

```
hint.driver.unit.keyword="value"
```

3 дахь шатны ачаалагч дуудагчийн синтакс нь:

```
set hint.driver.unit.keyword=value
```

Дээрх `driver` нь төхөөрөмжийн драйверийн нэр, `unit` нь төхөөрөмжийн драйверийн нэгжийн дугаар, `keyword` нь сануулга түлхүүр үг юм. Түлхүүр үг нь дараах тохируулгуудаас тогтож болно:

- `at`: төхөөрөмж холбогдсон шугамыг заана.
- `port`: ашиглагдах I/O-ны эхлэх хаягийг заана.
- `irq`: ашиглагдах тасалдал хүсэлтийн дугаарыг заана.
- `drq`: DMA сувгийн дугаарыг заана.
- `maddr`: төхөөрөмжийн эзэлж байгаа физик санах ойн хаягийг заана.
- `flags`: төхөөрөмжийн хувьд төрөл бүрийн тугийн битүүдийг тохируулна.
- `disabled`: хэрэв `1` гэж тохируулагдсан бол төхөөрөмж хаагдаж идэвхгүй болно.

Төхөөрөмжийн драйверууд нь энд жагсаагдсанаас илүү сануулгуудыг хүлээн авч эсвэл шаардаж болох бөгөөд гарын авлагын хуудсыг үзэхийг зөвлөж байна. Дэлгэрэнгүй мэдээллийн талаар device.hints(5), kenv(1), loader.conf(5), болон loader(8) гарын авлагуудаас зөвлөгөө авна уу.

# 13.6. Init: Процесс хяналтын эхлүүлэлт

Цөм ачаалж дууссаны дараа /sbin/init эсвэл `loader`-д `init_path` хувьсагчид заагдсан програмын замд байрлах хэрэглэгчийн процесс init(8)-д хяналтаа дамжуулдаг.

## 13.6.1. Автомат дахин ачаалах дараалал

Автомат дахин ачаалах дараалал нь систем дэх файлын системүүд бүрэн бүтэн байгаа эсэхийг шалгадаг. Хэрэв тэдгээр нь тийм биш бөгөөд fsck(8) нь UFS файлын системийн бүрэн бүтэн бус байдлыг засварлаж чадахгүй байгаа бол init(8) нь администраторуудад асуудлыг шийдэх боломж олгон ганц-хэрэглэгчийн горим уруу системийг оруулдаг.

## 13.6.2. Ганц-хэрэглэгчийн горим

Энэ горимд автомат дахин ачаалах дарааллын дагуу, хэрэглэгч `-s` тохируулга ашиглан ачаалах эсвэл `loader`-д `boot_single` хувьсагчийг тохируулж орж болдог.

Мөн shutdown(8)-ийг дахин ачаалах `-r` эсвэл зогсоох `-h` тохируулгуудгүйгээр дуудан олон-хэрэглэгчийн горимоос энэ горим уруу орж бас болно.

Хэрэв системийн `консол` нь `insecure буюу ажултай` гэж /etc/ttys-д тохируулагдсан бол систем ганц-хэрэглэгчийн горимыг эхлүүлэхээсээ өмнө `root` нууц үгийг асуудаг.

*Жишээ 12. /etc/ttys дахь ажултай консол*

```
# name  getty                           type    status          comments
#
# If console is marked "insecure", then init will ask for the root password
# when going to single-user mode.
```

```
console none                            unknown off insecure
```

`Аюултай` консол гэдэг нь консолын хувьд физик аюулгүй байдлыг аюултай гэж авч үзэн зөвхөн `root` нууц үгийг мэддэг хэн нэгэн ганц хэрэглэгчийн горимыг ашиглаж болох юм. Тиймээс аюулгүй байдлын энэ аргыг нэмэхийн тулд `secure буюу аюулгүй`г биш `insecure буюу аюултай`г сонгоорой.

### 13.6.3. Олон-хэрэглэгчийн горим

Хэрэв init(8) файлын системийг цэгцтэйг мэдвэл эсвэл хэрэглэгч өөрийн тушаалуудаа ганц-хэрэглэгчийн горимд ажиллуулаад дууссаны дараа системийн эх үүсвэрийн тохиргоог эхлүүлдэг олон-хэрэглэгчийн горим уруу систем ордог.

#### 13.6.3.1. Эх үүсвэрийн тохиргоо (rc)

Эх үүсвэрийн тохиргооны систем нь /etc/defaults/rc.conf файлаас тохиргооны анхдагчууд болон системийн тусгайлсан нарийн зүйлүүдийг /etc/rc.conf файлаас уншиж дараа нь /etc/fstab-д дурдагдсан системийн файлын системүүдийг холбодог. Сүлжээний үйлчилгээнүүд, бусад системийн дэмонуудыг эхлүүлэн төгсгөлд нь локал суулгагдсан багцуудын эхлүүлэх скриптүүдийг ажиллуулдаг.

Эх үүсвэрүүдийн тохиргооны системүүдийн талаар дэлгэрэнгүйг мэдэхийг хүсвэл rc(8) гарын авлагын хуудаснаас харж скриптүүдийг өөрсдийг нь шалгаж үзээрэй.

## 13.7. Унтраах дараалал

shutdown(8) ашиглан хянагдсан унтраалт хийгдэхэд init(8) нь /etc/rc.shutdown скриптийг ажиллуулахыг оролдож дараа нь бүх процессууд уруу `TERM` дохио явуулах бөгөөд дараагаар нь хугацаандаа дуусаагүй процессууд уруу `KILL` дохио илгээдэг.

Тэжээлийн удирдлагыг дэмждэг архитектурууд болон системүүд дээрх FreeBSD машиныг унтраахдаа тэжээлийг даруй унтраахын тулд `shutdown -p now` тушаалыг ашиглаарай. FreeBSD системийг дахин ачаалахдаа `shutdown -r now` тушаалыг ашиглана. shutdown(8)-г ажиллуулахын тулд `root` эсвэл `operator` бүлгийн гишүүн байх хэрэгтэй. halt(8) болон reboot(8) тушаалууд бас ашиглагдаж болно. Тэдгээрийн гарын авлагын хуудсууд болон shutdown(8)-ы хуудсанд хандан дэлгэрэнгүй мэдээлэл авна уу.

Тэжээлийн удирдлага нь acpi(4)-г модул хэлбэрээр дуудах эсвэл цөмд статикаар орж хөрвүүлэгдсэн байхыг шаарддаг.

# Chapter 14. Хэрэглэгчид ба үндсэн бүртгэл зохицуулалт

## 14.1. Ерөнхий агуулга

FreeBSD суусан компьютерийг нэг зэрэг олон хэрэглэгч хэрэглэж болдог. Мэдээж, тодорхой мөчид дэлгэцийн урд гар ашиглаж зөвхөн нэг хэрэглэгч суудаг ч олон хэрэглэгчид сүлжээгээр холбогдож өөрсдийнхөө ажлыг гүйцэтгэж чадна. Системийг хэрэглэхийг хүссэн хэрэглэгч нь бүртгэлтэй байх ёстой.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD систем дэх хэрэглэгчдийн төрөл бүрийн бүртгэлийн ялгаа.
- Хэрэглэгчийн бүртгэл хэрхэн нэмэх болон устгах.
- Хэрэглэгчийн бүтэн нэр, эсвэл эрхэмлэдэг shell зэрэг бүртгэл мэдээллийг хэрхэн өөрчлөх.
- Бүртгэлүүдийн бүлэг ба хэрэглэгч бүрд зарцуулагдах Процессор, мөн санах ойн хэмжээг хэрэглэх эрх зэрэгт хэрхэн хязгаарлалт хийх.
- Бүртгэлийн зохицуулалтыг хялбар болгохын тулд бүлгийг хэрхэн хэрэглэх.

Энэ бүлгийг уншиж эхлэхээсээ өмнө та дараах зүйлсийг мэдсэн байх хэрэгтэй:

- UNIX® болон FreeBSD-н үндсийг ойлгох.

## 14.2. Танилцуулга

FreeBSD систем рүү хандах бүх хандалт, мөн хэрэглэгчээс гүйцэтгэгдэж байгаа үйлдлүүд нь бүгд бүртгэлээр зохицуулагдана. Тийм болохоор FreeBSD дээр бүртгэлийн зохицуулалт тун чухал.

FreeBSD системийн бүртгэл нь уг бүртгэлээ тодорхойлсон мэдээлэл агуулж байдаг.

**Хэрэглэгчийн нэр**

Энэ хэрэглэгчийн нэр нь `login:` хэсэгт нэвтрэхдээ бичдэг нэр. Хэрэглэгчийн нэр нь энэ компьютерт ийм нэрээр цорын ганц байх ёстой. Хоёр хэрэглэгч нэг ижил нэртэй байж болохгүй. passwd(5);-д заасны дагуу зөв хэрэглэгчийн нэр үүсгэх маш олон дүрэм бий. Ерөнхийдөө, та найман тэмдэгтээс хэтрэлгүй, цөөхөн байж болно, бүгд бага үсгээр бичигдэх нэр хэрэглэнэ. Энэ нь хуучин програмуудын хувьд нийцтэй байдаг.

**Нууц үг**

Бүх бүртгэл нь харгалзах нууц үгтэй. Нууц үг нь хоосон байж болох боловч тэгэхийг зөвлөдөггүй бөгөөд бүх бүртгэл өөрийн нууц үгтэй байх хэрэгтэй.

**Хэрэглэгчийн ID (UID)**

UID бол ихэнхдээ 0 ээс 65535 хүртэлх тоо байдаг ба , системд хэрэглэгчийг уг ганц гоц тоогоор танихад хэрэглэгддэг. Дотоод ажиллагаандаа, FreeBSD нь UID дугаарыг хэрэглэж

хэрэглэгчийг танихдаа ашигладаг. Хэрэглэгчийн нэр ашигладаг бүх тушаалууд нь уг хэрэглэгчтэй ажиллахаасаа өмнө эхлээд UID руу нь хувиргадаг. Тэгэхээр, хэд хэдэн бүртгэл нь янз бүрийн нэртэй мөртлөө нэг UID дугаартай байж болно гэсэн үг. Хэрэв тийм бол FreeBSD тэд нарыг нэг хэрэглэгч гэж тооцно. Мэдээж танд ингэж хийх шаардлага байхгүй.

**Бүлгийн ID (GID)**

GID бол ихэнхдээ 0 ээс 65535 хүртэлх тоо байдаг ба , хэрэглэгчдийг бүлэглэсэн бүлгийг гоцолж танихад хэрэглэгддэг. Бүлэг нь хэрэглэгчдийн UID болгон руу хандалгүйгээр бүлэглэгдсэн хэсэг хэрэглэгчдийг GID бүлгийн дугаараар нь хандаж зохицуулахад хэрэглэдэг зарчим. Энэ нь зарим тохируулагч файлуудад маш их зай хэмнэхэд тус болдог. Нэг хэрэглэгч нь нэгээс илүү олон бүлэгт байж болно.

**Нэвтрэх төрөл**

Нэвтрэх төрөл нь бүлэглэх зарчмын нэмэгдэл бөгөөд систем дэх хэрэглэгчдийг ангилахад илүү дөхөм үзүүлдэг.

**Нууц үг өөрчлөх давтамж**

Анхны горимоороо FreeBSD нь хэрэглэгчдийг тодорхой давтамжтай нууц үгээ солихыг шаарддаггүй. Нууц үгийн шаардлагыг нэгж хэрэглэгчид хэрэгжүүлэхгүй байх, эсвэл хэсэг буюу бүх хэрэглэгчдийг тодорхой хугацаа өнгөрсөний дараа нууц үгээ солих ёстойгоор хэрэгжүүлж болно.

**Бүртгэлийн хүчинтэй огноо**

Анхны горимоороо FreeBSD нь бүртгэлд хязгаар тавьдаггүй. Хэрэв бүртгэл нь тодорхой хугацаанд хүчинтэй байх бол жишээлбэл сургуульд оюутнуудын хувьд бүртгэлд та хүчинтэй байх огноог нь зааж өгч болдог. Хүчинтэй огноо хэтрэх үед хэрэглэгч системд нэвтэрч чадахгүй бөгөөд харин уг бүртгэлд хамаарч байсан файл болон сангууд хэвээрээ байна.

**Хэрэглэгчийн бүтэн нэр**

Хэрэглэгчийн нэр нь FreeBSD дээр цор ганц гоц байх ёстой бөгөөд энэ нь заавал хэрэглэгчийн жинхэнэ нэр байх алба үгүй. Тийм учраас энэ нэр нь уг бүртгэлд тохирсон нэмэлт мэдээлэл байж болно.

**Гэрийн сан**

Гэрийн сан гэдэг нь систем дээр буй бүхлээр нь бичсэн сангийн байршил бөгөөд хэрэглэгч системд нэвтэрч оронгуут энэ санд эхэлж ордог. Энгийн тогтсон журманд, бүх хэрэглэгчийн гэрийн сангууд нь /home/username эсвэл /usr/home/username -д байрлуулсан байдаг. Хэрэглэгч нь гэрийн сан дотроо хувийн файлуудаа хадгалах бөгөөд мөн тэр дотроо дурын сан үүсгэх боломжтой.

**Хэрэглэгчийн бүрхүүл**

Бүрхүүл нь хэрэглэгчийг системтэй харьцахад хэрэглэгдэх анхны орчны бүрдлээр хангаж өгдөг. Маш олон төрлийн бүрхүүлүүд байдаг бөгөөд туршлагатай хэрэглэгчид өөрсдийнхөө бүртгэлийн тохиргоог өөрчилж эрхэмлэдэг бүрхүүлээ сонгож авдаг.

Бүртгэлийн гол гурван төрөл бий: супер хэрэглэгч, систем хэрэглэгч, тэгээд хэрэглэгчийн бүртгэл. Супер хэрэглэгчийн бүртгэлийг голдуу `root` гэж нэрлэдэг бөгөөд, ямар ч эрхийн

хязгаарлалтгүйгээр системийг зохицуулж чаддаг. Систем хэрэглэгчид нь үйлчилгээнүүдийг ажиллуулах эрхтэй. Хэрэглэгчийн бүртгэл нь болохоор жирийн хүмүүст хэрэглэгддэг бөгөөд нэвтэрч орох, захиагаа шалгах зэрэг үйлдэлд ашиглагддаг.

## 14.3. Супер хэрэглэгчийн бүртгэл

Голдуу `root` гэж нэрлэгдэх супер хэрэглэгчийн бүртгэл нь системийг удирдах зорилгоор ашиглагддаг бөгөөд захиа шалгах, програм ажиллуулах, системийн зарим үйлдлийг турших зэрэг үйлдэлд хэрэглэх хэрэггүй.

Жирийн хэрэглэгчээс ялгаатай нь, супер хэрэглэгч нь ямар ч хязгаарлалтгүйгээр системд үйлдэл хийж чаддаг болохоор, болгоомжгүй хийсэн үйлдэл тань маш том алдаанд хүргэж болзошгүй юм. Энгийн хэрэглэгчийн бүртгэлээр алдаатай үйлдэл хийсэн ч системийг эвдэх чадваргүй учраас онцгой шаардлага гарахгүй л бол энгийн хэрэглэгчийн бүртгэлийг байнга хэрэглэх нь сайн.

Супер хэрэглэгч байх үедээ гүйцэтгэх гэж буй тушаалаа үргэлж давхар удаа эсвэл гурван удаа шалгаж байхгүй бол, нэмэлт зай авалт эсвэл тэмдэг дутсанаас болж нөхөж баршгүй өгөгдлийн гарз гарч болзошгүй билээ.

Системийн администраторын хувьд хэрэглэгчийн бүртгэл нээж ердийн хэрэглээндээ зориулж ашиглах нь зүйтэй юм. Энэ арга нь олон хэрэглэгчтэй эсвэл ганц хэрэглэгчтэй машины аль алинд нь хэрэгжигдэх хэрэгтэй. Энэ бүлгийн сүүл хэсэгт нэмэлт бүртгэл үүсгээд, жирийн хэрэглэгч болон супер хэрэглэгчийн хооронд хэрхэн сольж болох талаар зааварлах болно.

## 14.4. Систем бүртгэл

Систем хэрэглэгчид нь DNS, захиа, вэб сервер гэх мэтчилэн үйлчилгээнүүдийг ажиллуулах эрхтэй байдаг. Гол шалтгаан нь аюулгүй байдал; хэрэв бүх үйлчилгээнүүд супер хэрэглэгч эрхээр ажиллавал хязгааргүй үйлдэл хийх эрхтэй болох аюултай.

Систем хэрэглэгчдийн нэг жишээ бол `дэмон буюу далд чөтгөр`, `operator`, `bind` (Домэйн нэрний үйлчилгээнд зориулсан), `news`, мөн `www` нар юм.

`nobody` бол системийн ердийн онцгой эрхгүй систем хэрэглэгч. Гэхдээ, хэрэв та хэдий чинээ файл болон үйлдлүүдэд хамаатай үйлчилгээнүүдийг энэ `nobody` хэрэглэгчээр ашиглана, төдий хэмжээний эрхтэй байна гэсэн үг.

## 14.5. Энгийн хэрэглэгчийн бүртгэл

Хэрэглэгчдийн энэ бүртгэл нь бодит хүмүүсийг систем рүү нэвтрэхэд хэрэглэгдэж, тухайн хэрэглэгч болон орчныг тусгаарлан, энгийн хэрэглэгчийг систем эвдэхээс сэргийлэх, мөн хэрэглэгчид өөрсдийнхөө орчныг бусдад нөлөөлөхгүйгээр өөрчилж болох зэрэгт зориулагдсан.

Таны систем рүү орж байгаа бүх хүн өөрийн гэсэн бүртгэлтэй байх хэрэгтэй. Ингэснээр та хэн юу хийж байгааг мэдэж болохоос гадна, хүн болгоны тохируулга бусдын тохируулгатай

холилдох, эсвэл нэг нэгнийхээ захиаг унших зэрэг нөхцлүүдээс сэргийлж чадна.

Хэрэглэгч болгон өөрийнхөө эрхэмлэдэг бүрхүүл, засварлагч, гарын товчлуур, мөн хэл зэрэг тохируулгыг өөртөө зориулж системд тохируулж болно.

## 14.6. Бүртгэлд өөрчлөлт хийх

FreeBSD дээр хэрэглэгчийн бүртгэлийг өөрчилж болох маш олон тушаалууд байдаг. Доорх хэсэгт хамгийн түгээмэл хэрэглэгддэг тушаалуудыг хэрэглэх жишээтэй нь хамт нэгтгэж харууллаа.

| Тушаал | Тайлбар |
|---|---|
| adduser(8) | Шинэ хэрэглэгч нэмэхэд хэрэглэгддэг, санал болгогдсон, бичиж гүйцэтгэгддэг тушаал програм. |
| rmuser(8) | Хэрэглэгчийг устгахад хэрэглэгддэг, санал болгогдсон, бичиж гүйцэтгэгддэг тушаал програм. |
| chpass(1) | Хэрэглэгчийн өгөгдлийн мэдээллийг өөрчилдөг нийцтэй багаж. |
| passwd(1) | Хэрэглэгчийн нууц үгийг сольдог, энгийн бичиж гүйцэтгэгддэг тушаах багаж. |
| pw(8) | Хэрэглэгчийн бүх мэдээллийг өөрчилж чаддаг, хүчирхэг нийцтэй багаж. |

### 14.6.1. adduser

adduser(8) нь шинэ хэрэглэгч нэмдэг энгийн програм. Шинэ хэрэглэгч нэмэхэд энэ програм нь автоматаар passwd болон group файлуудыг шинэчилдэг. Мөн энэ програм нь шинэ хэрэглэгчид зориулж гэрийн сан үүсгэж, анхдагч тохируулгын файлуудыг /usr/shared/skel сангаас хуулан, хэрэв нэмэлт сонголт хийсэн бол шинэ хэрэглэгч рүү мэндчилгээний захиа илгээдэг.

*Жишээ 13. FreeBSD дээр хэрэглэгч нэмэх нь*

```
# adduser
Username: jru
Full name: J. Random User
Uid (Leave empty for default):
Login group [jru]:
Login group is jru. Invite jru into other groups? []: wheel
Login class [default]:
Shell (sh csh tcsh zsh nologin) [sh]: zsh
Home directory [/home/jru]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]:
```

```
Use an empty password? (yes/no) [no]:
Use a random password? (yes/no) [no]:
Enter password:
Enter password again:
Lock out the account after creation? [no]:
Username   : jru
Password   : ****
Full Name  : J. Random User
Uid        : 1001
Class      :
Groups     : jru wheel
Home       : /home/jru
Shell      : /usr/local/bin/zsh
Locked     : no
OK? (yes/no): yes
adduser: INFO: Successfully added (jru) to the user database.
Add another user? (yes/no): no
Goodbye!
#
```

Оруулсан нууц үг тань танд харуулагдахгүй бас одоор ч дүрсэлж харуулагддаггүй. Тийм болохоор нууц үгээ хянаж зөв оруулах хэрэгтэй.

### 14.6.2. `rmuser`

Та rmuser(8)-г хэрэглэн системээс хэрэглэгчийг нэг мөсөн устгаж болно. rmuser(8) нь дараах алхмуудыг гүйцэтгэдэг:

1. Хэрэглэгчийн crontab(1) оруулгуудыг устгана (хэрэв байвал).
2. Хэрэглэгчид хамаарах at(1) ажлуудыг устгана.
3. Хэрэглэгчийн эзэмшдэг бүх процессуудыг устгана.
4. Системийн дотоод нууц үгийн файлаас хэрэглэгчийг устгана.
5. Хэрэглэгчийн гэрийн санг устгана (хэрэв хэрэглэгч эзэмшдэг бол).
6. /var/mail доторх уг хэрэглэгчид ирсэн захианууудыг устгана.
7. Хэрэглэгчийн эзэмшиж байсан файлуудыг /tmp гэх мэтийн түр хадгалагч сангаас устгана.
8. Эцэст нь /etc/group дотор буй хэрэглэгчийн хамрагддаг бүх бүлгүүдээс уг хэрэглэгчийг устгана.

Хэрэв бүлгийн нэр нь уг хэрэглэгчийн нэртэй ижил бөгөөд уг бүлэг нь хоосорч байвал уг бүлэг нь adduser(8) програмаар зөвхөн уг хэрэглэгчид гоцолж үүсгэгдсэн болохоор бас устгагдана.

Супер хэрэглэгчийн бүртгэлийг устгавал маш олон устгалга хийгдэх болохоор rmuser(8) програм хэрэглэгдэж чадахгүй.

Анхны горим нь, таныг юу хийж байгаад итгэлтэй байхын тулд лавлаж харилцах горим хэрэглэгддэг.

*Жишээ 14. `rmuser` Лавлаж харилцан бүртгэл устгах*

```
# rmuser jru
Matching password entry:
jru:*:1001:1001::0:0:J. Random User:/home/jru:/usr/local/bin/zsh
Is this the entry you wish to remove? y
Remove user's home directory (/home/jru)? y
Updating password file, updating databases, done.
Updating group file: trusted (removing group jru -- personal group is empty) done.
Removing user's incoming mail file /var/mail/jru: done.
Removing files belonging to jru from /tmp: done.
Removing files belonging to jru from /var/tmp: done.
Removing files belonging to jru from /var/tmp/vi.recover: done.
#
```

### 14.6.3. chpass

chpass(1) нь хэрэглэгчийн өгөгдлийн бааз болох нууц үг, бүрхүүл, мөн хувийн мэдээлэл зэргүүдийг өөрчилдөг.

Зөвхөн супер хэрэглэгч chpass(1)-г хэрэглэн бусдын мэдээллийг өөрчлөх эрхтэй.

Хэрэглэгчийн нэрийг нэмэлт сонголт болгож өгөхөөс бусад үед, ямар нэгэн сонголт өгөөгүй тохиолдолд chpass(1) нь хэрэглэгчийн мэдээллийг агуулсан засварлагч нээж харуулдаг. Хэрэглэгч уг засварлагчаас гарах үед хэрэглэгчийн мэдээлэл шинэ мэдээллээр шинэчлэгддэг.

Засварлагчаас гарах үед хэрэв та супер хэрэглэгч биш бол нууц үгээ оруулахыг асуух болно.

*Жишээ 15. Супер хэрэглэгчийн лавлаж асуух `chpass` хэрэглээ*

```
#Changing user database information for jru.
Login: jru
Password: *
Uid [#]: 1001
Gid [# or name]: 1001
Change [month day year]:
Expire [month day year]:
Class:
Home directory: /home/jru
Shell: /usr/local/bin/zsh
```

```
Full Name: J. Random User
Office Location:
Office Phone:
Home Phone:
Other information:
```

Энгийн хэрэглэгч нь өөрийнхөө энэ мэдээллийн зөвхөн жаахан хэсгийг л өөрчилж чадна.

*Жишээ 16. Энгийн хэрэглэгчийн лавлаж асуух chpass хэрэглээ*

```
#Changing user database information for jru.
Shell: /usr/local/bin/zsh
Full Name: J. Random User
Office Location:
Office Phone:
Home Phone:
Other information:
```

chfn(1) ба chsh(1) нар нь зөвхөн chpass(1)-н холбоосууд юм. Нэг ёсондоо ypchpass(1), ypchfn(1), болон ypchsh(1) нартай ижилхэн холбоос. NIS дэмжлэг автоматаар хийгдсэн байгаа. Тийм болохоор тушаалын урд нь yp угтвар залгах шаардлагагүй. NIS хэрхэн тохируулах талаар Сүлжээний орчны Серверүүд хэсэгт тайлбарлагдах болно.

### 14.6.4. passwd

passwd(1) нь энгийн хэрэглэгч үедээ эсвэл бусдын нууц үгийг солихоор бол супер хэрэглэгч болж нууц үг сольдог энгийн арга.

Болчимгүй алдаа эсвэл зөвшөөрөөгүй өөрчлөлт оруулахаас сэргийлж, шинэ нууц үгийг суулгахаас өмнө анхны нууц үгийг асуудаг. Супер хэрэглэгч шууд солих боломжтой.

*Жишээ 17. Өөрийнхөө нууц үгийг солих нь*

```
% passwd
Changing local password for jru.
Old password:
New password:
Retype new password:
passwd: updating the database...
passwd: done
```

*Жишээ 18. Супер хэрэглэгч бол бусдын нууц үгийг солих нь*

```
# passwd jru
Changing local password for jru.
New password:
Retype new password:
passwd: updating the database...
passwd: done
```

chpass(1), yppasswd(1) зэрэг нь ердөө passwd(1)-н холбоосууд юм. Тийм болохоор NIS дээр эдгээр нь бүгд ажилладаг.

### 14.6.5. pw

pw(8) нь хэрэглэгч нэмэх, устгах, өөрчлөх, мөн хэрэглэгч ба бүлгийг харуулдаг тушаалын мөрийн багаж юм. Системийн хэрэглэгч ба бүлгийн файлд хамгийн түрүүнд ханддаг багаж. pw(8) нь маш олон хүчтэй тушаалын сонголтуудтай байдаг болохоор бүрхүүлийн орчны скриптэд хэрэглэхэд тохиромжтой. Гэхдээ шинэ хэрэглэгчдэд энэ програм нь өмнө үзүүлсэн тушаалуудыг бодвол илүү хэцүү хүндрэлтэй гэж тооцогдож магадгүй.

## 14.7. Хэрэглэгчдэд хязгаарлалт хийх

FreeBSD нь хэрэглэгчдийн систем хэрэглэх боломжийг нь хязгаарлах боломжтой байдаг. Эдгээр хязгаарлалт нь хоёр хэсэгт хуваагддаг: дискний хуваарилалт, тэгээд бусад нөөцийн хязгаарлалтууд.

Дискийн хуваарилалт нь хэрэглэгчийн диск хэрэглэх хэмжээг зааглаж өгдөг бөгөөд тухай бүрд уг хэмжээг тооцож бодолгүй шуурхай шалгах боломж өгдөг. Хуваарилалт нь Файлын системийн ноогдлууд бүлэгт авч хэлэлцэгдсэн.

Бусад нөөцийн хязгаарлалтуудад Процессор, санах ойн хэмжээ болон бусад хэрэглэгчийн хэрэглэх нөөцүүд ордог. Эдгээр нь нэвтрэх ангилал ашиглаж зааглаж өгдөг бөгөөд энд одоо ярих болно.

Нэвтрэх ангилал нь /etc/login.conf-д заагдсан байгаа. Нарийвчилсан ойлголт энэ бүлэгт ороогүй, гэхдээ login.conf(5) гарын авлагад тодорхой бичигдсэн байгаа. Хэрэглэгч бүрд нэвтрэх ангилал заагдсан байдаг гэж хэлэхэд хангалттай (анхдагч утга нь `default`), бөгөөд нэвтрэх ангилал бүр өөртөө тодорхой нэвтрэх чанаруудыг агуулсан байдаг. Нэвтрэх чанар гэдэг нь `нэр=утга` гэсэн хослол бөгөөд үүний *нэр* нь бидний мэдэх хэрэглэгчийн нэр ба *утга* нь *нэр*тэй хамаатай боловсруулагдсан мөр байдаг. Нэвтрэх ангилал ба чанарыг тохируулах нь харьцангуй амархан бөгөөд мөн login.conf(5) дотор тодорхой бичигдсэн.

FreeBSD ердийн үед шууд /etc/login.conf файлыг уншдаггүй, харин түргэн хайж харахад илүүгээр нь /etc/login.conf.db өгөгдлийн бааз файлыг уншдаг. /etc/login.conf-д засвар хийх бүртээ /etc/login.conf.db файлыг шинэчлэх ёстой бөгөөд ингэхийн тулд дараах тушаалыг гүйцэтгэх хэрэгтэй:

```
# cap_mkdb /etc/login.conf
```

Нөөцийн хязгаарлалт нь энгийн нэвтрэх чанараас хоёр замаар ялгардаг. Эхнийх нь, хязгаарлалт бүр зөөлөн (одоогийн) ба хатуу хязгаарлалт гэж байдаг. Зөөлөн хязгаарлалт нь хэрэглэгчээр юм уу програмаар тохируулагдаж болдог бөгөөд хатуу хязгаарлалтаас дээгүүр биш. Харин хатуу хязгаарлалт нь хэрэглэгчээр доошоо тохируулагдаж болох боловч хэзээ ч дээшлэгдэж чадахгүй. Хоёр дахь нь, ихэнх нөөцийн хязгаарлалт нь тусгай хэрэглэгчдийн процесс дээр хэрэгжигдэх бөгөөд харин хэрэглэгч дээр бүхлээрээ хэрэгжигдэхгүй. Тэмдэглэж хэлэхэд, хэдийгээр эдгээр ялгаанууд нь онцгой хязгаарлалт хийх үед голлох боловч нэвтрэх чанарын үндсэн хүрээнд хэрэгжигдэхгүй. (өөрөөр хэлбэл, эдгээр нь *үнэхээр* онцгой нэвтрэх чанарын тохиолдол биш).

Тэгэхээр, элдвийг нуршилгүйгээр, энгийн байнга хэрэглэгддэг нөөцийн хязгаарлалтуудыг доор жагсаав (үлдсэн хязгаарлалтуудыг мөн тэдгээрийн нэвтрэх чанаруудыг нь login.conf(5) -с харна уу).

`coredumpsize`

Програмаар үүсгэгдсэн core файлын хэмжээний хязгаар нь мэдээж дискний хэрэглэх хязгаарт багтана. (`файлын хэмжээ`, эсвэл диск хуваарилалт г.м). Тэгсэн хэдий ч, энэ нь дискний хэрэглэх хэмжээг хянахад хэрэглэгддэг нэг их чухал биш тохируулга: core файлыг хэрэглэгчид өөрсдөө үүсгэдэггүй бөгөөд тэдгээрийг дандаа устгаад байдаггүй учир үүгээр тохируулсан үед том програм (жишээ нь, emacs) ажиллахдаа core файл үүсч диск дүүрэхээс сэргийлж болох юм.

`cputime`

Энэ нь хэрэглэгчийн процесс хэрэглэж болох хамгийн их процессорын давтамж юм. Хэтрүүлж хэрэглэсэн процесс цөмөөс устгагддаг.

Энэ нь процессорын *давтамж* дээр тавигдсан хэрэглээний хязгаар бөгөөд top(1) ба ps(1)-н зарим талбарт харагддаг процессорын хувийг заасан заалт биш.

`filesize`

Энэ хэмжээ нь хэрэглэгчийн хэрэглэж болох хамгийн их файлын хэмжээ. диск хуваарилалтаас ялгаатай нь, энэ хязгаарлалт нь файл бүрд тавигддаг бөгөөд хэрэглэгчийн эзэмшдэг бүх файлд биш.

`maxproc`

Энэ нь хэрэглэгчийн ажиллуулж болох хамгийн их процессийн тоо юм. Үүнд нүүрэн талын болон ар талын процессууд багтана. Мэдээж, энэ нь `kern.maxproc`-д заагдсан sysctl(8) системийн хязгаараас хэтэрч болохгүй. Мөн тэмдэглэж хэлэхэд, үүнийг хэтэрхий багаар тохируулбал хэрэглэгчийн олон дахин нэвтэрч орох, дамжуулах хоолой ашиглах зэрэг бүтээмжид нөлөөлөх болно. Том програмыг хөрвүүлэх гэх мэт зарим гүйцэтгэл нь мөн олон процесс шаарддаг.

`memorylocked`

Энэ нь ямар нэг процесс санах ойд mlock(2) ашиглан түгжсэн санамж шаардах үед хязгаарлах хамгийн их санах ойн хэмжээ юм. Зарим amd(8) гэх мэт системийн ноцтой програмууд нь үндсэн санах ойг түгждэг бөгөөд swap хийх үед системийн зохицуулалтанд тусалдаггүй.

`memoryuse`

Энэ нь ямар нэг процесс ямар ч үед хэрэглэж болох хамгийн их санах ойн хэмжээ. Энэ нь үндсэн санах ой болон swap зай хоёуланд хамаатай. Санах ойн хязгаарлалт бүгдийг нь хийдэггүй ч эндээс эхлэх нь зөв эхлэлд тооцогдоно.

`openfiles`

Энэ нь процессийн нээж болох файлуудын хамгийн их хэмжээ. FreeBSD дээр сокет, мөн IPC сувгууд нь файл хэлбэрээр хэрэглэгддэг болохоор үүнийг маш багаар тохируулахаа тун болгоомжлох хэрэгтэй. Систем даяар хязгаарлахын тулд `kern.maxfiles`-д зааж өгнө sysctl(8).

`sbsize`

Энэ нь сүлжээнд хэрэглэгдэх санах ойн хязгаар буюу хэрэглэгчийн хэрэглэж болох mbufs юм. Энэ нь сүлжээний холболтыг хязгаарлахад ерөнхийдөө хэрэглэгдэж болно.

`stacksize`

Энэ нь процессийн хэрэглэх стек санах ойн ихсэж болох хамгийн их хэмжээ. Энэ нь програмын хэрэглэж болох санах ойн хэмжээг хязгаарлахад хангалттай биш. Голдуу бусад хязгаарлалтуудтай хамтарч хэрэглэгдэх хэрэгтэй.

Нөөцийн хязгаарлалтыг тохируулж байхдаа бусад зарим зүйлсийг санаж байх хэрэг байдаг. Зарим энгийн зөвлөгөө, санал мөн төрөл бүрийн тайлбаруудыг доор жагсаав.

- Систем эхлэх үед /etc/rc-с ажилладаг процессууд нь `дэмон` нэвтрэх төрөлд багтдаг.
- Хэдийгээр ихэнх хязгаарлалтуудыг агуулсан /etc/login.conf файл системтэй цуг ирдэг ч гэсэн зөвхөн систем удирдагч та л таны системд ямар нь хамаатайг мэднэ. Тохируулгыг хэт их болговол системээ буруу хэрэглэгчдэд нээж өгөх бөгөөд хэрэв хэт багаар тохируулбал бүтээмжид хэт нарийдна.
- Xorg хэрэглэгчид бусад хэрэглэгчдийг бодвол илүү их нөөцөөр хангагдах хэрэгтэй. Xorg нь өөрөө маш их нөөц авдаг бөгөөд бас хэрэглэгчдэд олон програмыг зэрэг ажиллуулах боломж өгдөг.
- Маш олон хязгаарлалт нь хэрэглэгч дээр бүхэлд нь биш харин нэгж процессууд дээр тавигддаг гэдгийг санах хэрэгтэй. Жишээлбэл, `openfiles` тохируулгыг 50 гэвэл, энэ нь хэрэглэгчийн ажиллуулж буй процесс болгон 50 файл нээж болно гэсэн үг. Тийм болохоор, хэрэглэгчийн нээж болох файлуудын нийт хэмжээ нь `openfiles`-н утгыг `maxproc` -н утгаар үржүүлж гарна. Энэ нь мөн санах ойн хэрэглээнд бас хэрэгжинэ.

Нөөцийн хязгаарлалт ба нэвтрэх ангилал, мөн ерөнхий нэвтрэх чанаруудын тухай нэмэлт мэдээллийг харгалзах гарын авлага: cap.mkdb(1), getrlimit(2), login.conf(5) нараас харна уу.

# 14.8. Бүлэг

Бүлэг гэдэг нь хэрэглэгчдийн бүлэглэж жагсаасан жагсаалт юм. Бүлгүүд нь бүлгийн нэр ба GID (бүлгийн ID)-аар танигддаг. FreeBSD (мөн ихэнх бусад UNIX® төрлийн систем) дээр, цөмөөс хэрэглэгчийн процессийг юм хийхэд нь зөвшөөрөл өгөхдөө хоёр зүйлийг хэрэглэж шийддэгээс нэг нь хэрэглэгчийн UID, нөгөө нь тэр хэрэглэгчийн хамаардаг бүлэг байдаг. Ихэнхдээ хэрэглэгчийн процессын GID нь жагсаалтын эхний бүлэг байдаг.

Бүлгийн нэрийг бүлгийн GID-д харгалзуулсан бичлэг /etc/group файлд бий. Энэ нь энгийн текст файл бөгөөд тодорхойлох хоёр цэгээр таслагдсан дөрвөн талбар байдаг. Эхний талбар нь бүлгийн нэр, хоёр дахь нь нууцлалын хөрвүүлэлт хийгдсэн нууц үг, гурав дахь нь бүлгийн GID, тэгээд сүүлийнх нь таслалаар тусгаарлагдсан гишүүдийн жагсаалт. Бичиглэлийн дэлгэрэнгүй мэдээллийг group(5) хуудаснаас харна уу.

Супер хэрэглэгч /etc/group-г гараар засварлаж болно. Мөн pw(8) тушаалыг бүлэг нэмэх болон засварлахад хэрэглэж болно. Жишээлбэл, дараах жишээ нь `teamtwo` нэртэй бүлгийг нэмж байна:

*Жишээ 19. pw(8) хэрэглэж бүлэг нэмэх*

```
# pw groupadd teamtwo
# pw groupshow teamtwo
teamtwo:*:1100:
```

Дээрх `1100` гэсэн дугаар нь `teamtwo` нэртэй бүлгийн GID. Одоохондоо `teamtwo` бүлэгт гишүүд байхгүй байгаа. Дараах тушаал `jru` нөхрийг `teamtwo` бүлэгт нэмнэ.

*Жишээ 20. pw(8) ашиглан шинэ бүлэгт хэрэглэгчдийн бүртгэл нэмэх нь*

```
# pw groupmod teamtwo -M jru
# pw groupshow teamtwo
teamtwo:*:1100:jru
```

`-M` тохируулгад өгөх утга нь шинэ (хоосон) бүлэгт байх эсвэл солигдох таслалаар тусгаарлагдсан хэрэглэгчдийн жагсаалт байна. Хэрэглэгчийн хувьд энэ бүлгийн гишүүнчлэл нь нууц үгийн файлд заасан хэрэглэгчийн анхдагч бүлэг дээр нэмэгдээд өөр байна. pw(8) тушаалыг `groupshow` гэсэн сонголтоор ажиллуулахад хэрэглэгчид харуулагддаггүй бөгөөд, харин мэдээллүүд нь id(1) юм уу өөр ижил төрлийн тушаалаар харуулагддаг. pw(8) нь зөвхөн /etc/group файлыг өөрчилдөг болохоос нэмэлт мэдээллийг /etc/passwd файлаас уншдаггүй.

*Жишээ 21. pw(8) ашиглан шинэ гишүүнийг бүлэгт нэмэх нь*

```
# pw groupmod teamtwo -m db
# pw groupshow teamtwo
```

```
teamtwo:*:1100:jru,db
```

`-m` тохируулгад өгөх утга нь бүлэгт нэмэгдэх таслалаар тусгаарлагдсан хэрэглэгчдийн жагсаалт байна. Өмнөх жишээнээс ялгаатай нь эдгээр хэрэглэгчид нь бүлэгт нэмэгдэх бөгөөд бүлэг дэх хэрэглэгчдийн жагсаалтыг өөрчлөхгүй юм.

*Жишээ 22. id(1) хэрэглэж уг гишүүний бүлгийн мэдээллийг мэдэх нь*

```
% id jru
uid=1001(jru) gid=1001(jru) groups=1001(jru), 1100(teamtwo)
```

Энэ жишээн дээр `jru` гэдэг гишүүн `jru` болон `teamtwo` бүлгийн гишүүн байна.

Энэ тушаалын талаар болон /etc/group файлын хэлбэршилтийн мэдээллийг pw(8) болон group(5) гарын авлагаас харна уу.

# Chapter 15. Аюулгүй байдал

## 15.1. Ерөнхий агуулга

Энэ бүлэг нь системийн аюулгүй байдлын ухагдахуунуудын үндэс, зарим нэг нийтлэг практикийн сайн аргууд болон FreeBSD дэх зарим нэг дэвшилттэй сэдвүүдийг танилцуулах болно. Энд дурдагдсан олон сэдвүүдийг бас системийн болон Интернэтийн аюулгүй байдалд хэрэглэж болох юм. Интернэт нь хүн бүр таны найрсаг хөрш байхыг хүсдэг "найзархаг" газар байхаа аль хэдийн больсон. Өөрийн системийг аюулгүй болгох нь таны өгөгдөл, оюуны өмч, цаг хугацаа зэрэг олон зүйлсийг хакерууд зэргийн савраас хамгаалахад хойшлуулашгүй чухал юм.

FreeBSD нь таны систем болон сүлжээний аюулгүй байдал болон бүрэн бүтэн байдлыг хангаж байдаг хэрэгслүүд болон арга замуудын цуглуулгыг агуулдаг.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD-ийн хувьд системийн аюулгүй байдлын үндсэн ухагдахуунууд.
- FreeBSD-д байдаг DES болон MD5 зэрэг төрөл бүрийн нууцлах арга замуудын талаар.
- Нэг удаагийн нууц үгийн нэвтрэлтийг хэрхэн тохируулах талаар.
- TCP Wrappers буюу TCP Гүйцэтгэлийг хялбаршуулагчдыг inetd-д ашиглан хэрхэн тохируулах талаар.
- FreeBSD дээр Kerberos5-г хэрхэн тохируулах талаар.
- IPsec-г хэрхэн тохируулж FreeBSD/Windows® машинуудын хооронд VPN үүсгэх талаар.
- FreeBSD-ийн SSH шийдэл болох OpenSSH-г хэрхэн тохируулж ашиглах талаар.
- Файлын системийн ACL-үүд гэж юу болох, тэдгээрийг хэрхэн ашиглах талаар.
- Portaudit хэрэгслийг хэрхэн ашиглаж Портын цуглуулгаас суулгагдсан гуравдагч програм хангамжийн багцуудыг аудит хийх талаар.
- FreeBSD-ийн аюулгүй байдлын зөвлөмжүүдийн сонордуулгуудыг хэрхэн хэрэглэх талаар.
- Процессийн Бүртгэл хөтлөх гэж юу болох талаар ойлголттой болж түүнийг FreeBSD дээр хэрхэн идэвхжүүлэх талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- FreeBSD болон Интернэтийн үндсэн ухагдахуунуудыг ойлгох.

Энэ номонд нийтдээ аюулгүй байдлын нэмэлт сэдвүүд хамрагдсан болно. Жишээ нь Mandatory Access Control буюу Шаардлагатай Хандалтын Хяналт Mandatory Access Control буюу Албадмал Хандалтын хяналт-д, Интернэт галт хананууудын талаар Галт хана-д хэлэлцэгдсэн байгаа.

## 15.2. Танилцуулга

Аюулгүй байдал нь системийн администратораас эхэлж түүнтэй дуусдаг үйл ажиллагаа юм.

BSD UNIX® олон хэрэглэгчийн системүүд нь угаасаа зарим нэг аюулгүй байдлыг хангаж байдаг боловч тэдгээр хэрэглэгчдийг "үнэнч" байлгахыг эрмэлздэг аюулгүй байдлын нэмэлт арга замуудыг бүтээж түүний ажиллагааг хангах ажил нь сисадмины магадгүй ганц, хамгийн том үүргүүдийн нэг юм. Таныг аюулгүй болгосон зөвхөн тэр хэмжээгээр машинууд нь аюулгүй байдаг бөгөөд аюулгүй байдлын санаа зовнилууд нь хүний ая тухтай хялбар байлгах гэсэн хэрэгцээтэй үргэлж тэмцэлдэж байдаг. Ерөнхийдөө UNIX® системүүд нь асар олон тооны зэрэгцээ процессуудыг ажиллуулах чадвартай бөгөөд эдгээр процессуудын ихэнх нь серверүүд болон ажилладаг - энэ нь гаднын зүйлс тэдэнтэй холбогдож ярилцах боломжтой гэсэн үг юм. Өчигдрийн миникомпьютерууд, мэйнфрэймүүдээс өнөөгийн ширээний компьютерууд болж компьютерууд нь сүлжээнд холбогдож сүлжээнүүд нь хоорондоо холбогдох тусам аюулгүй байдал нь улам илүү том асуудал болсоор байна.

Системийн аюулгүй байдал нь сүйрүүлэхийг оролдсон эсвэл системийг ашиглагдахааргүй болгох гэсэн, гэхдээ `root` бүртгэлийг буулган авах ("root-г эвдэх") оролдлого хийдэггүй, халдлагууд зэрэг төрөл бүрийн халдлагуудыг зогсоохтой бас хамааралтай юм. Аюулгүй байдлын санаа зовнилуудыг хэд хэдэн зэрэглэлд хувааж болно:

1. Үйлчилгээг зогсоох халдлагууд.
2. Хэрэглэгчийн бүртгэл буулган авалтууд.
3. Хандаж болох серверүүдээр дамжин root-г буулган авах.
4. Хэрэглэгчийн бүртгэлүүдээс дамжин root-г буулган авах.
5. Арын хаалга үүсгэлт.

Үйлчилгээг зогсоох халдлага нь машиныг хэрэгцээтэй эх үүсвэрээс нь салгах үйлдэл юм. Ихэвчлэн DoS халдлагууд нь сүйрүүлэхийг оролдсон эсвэл машиныг түүн дээрх серверүүд болон сүлжээний стекийг эзэмдэн ашиглах боломжгүй болгодог балмадаар хүчлэх арга замууд юм. Зарим DoS халдлагууд нь сүлжээний стек дэх алдаануудыг ашиглан ганц пакетаар машиныг сүйрүүлэхийг оролддог. Үүнийг зөвхөн алдааны засварыг цөмд хийснээр засах боломжтой. Систем дээрх хөнөөлтэй нөхцөлд байх тэр серверийн дуудлагыг хязгаарладаг тохируулгуудыг зөв зааж серверүүд уруу хийсэн халдлагуудыг ихэвчлэн засаж болдог. Сүлжээний балмадаар хүчлэх халдлагуудын эсрэг арга хэмжээ авахад илүү төвөгтэй байдаг. Жишээ нь хууран мэхэлсэн пакетийн халдлагыг зогсоох бараг л боломжгүй, таны системийг Интернэтээс салгахад хүргэж болох юм. Энэ нь таны машиныг зогсоож чадахгүй байж болох боловч таны Интернэтийн холболтыг дүүргэж болно.

Хэрэглэгчийн бүртгэлийг буулган авах халдлага нь DoS халдлагаас илүү их тохиолддог. Одоо болтол олон сисадминууд стандарт telnetd, rlogind, rshd, болон ftpd серверүүдийг өөрсдийн машинууд дээр ажиллуулсаар байна. Анхдагчаар серверүүд нь шифрлэсэн холболт дээр ажилладаггүй. Ийм холболт дээр хэрэв та багагүй хэмжээний хэрэглэгчидтэй бөгөөд тэдгээр хэрэглэгчдээс нэг болон хэд хэд нь алсаас (энэ нь систем уруу нэвтрэн орох хамгийн нийтлэг тав тухтай арга юм) таны систем уруу нэвтрэн орж байгаа бол тэдгээр хэрэглэгчийн нууц үг дундаасаа сүлжээгээр шиншлэгдэн алдагдах боломжтой байдаг. Анхааралтай системийн админ тэр хэрэглэгчийн алсаас хандсан бүртгэлүүд дээрээс бүр амжилттай болсон нэвтрэлтүүдэд хүртэл сэжигтэй эхлэл хаягууд байгаа эсэхийг хайн шинжилдэг.

Халдагч хэрэглэгчийн бүртгэлд хандаж чадсаны дараа `root`-г бас эвдэж чадна гэдгийг үргэлж бодож байх хэрэгтэй. Гэхдээ жинхэнэ амьдрал дээр бол сайн аюулгүй байдлыг хангаж

нууцлаг болгосон байнга ажиллагааг нь хянаж байдаг систем дээр хэрэглэгчийн бүртгэлд хандах нь халдагч заавал ч үгүй `root` эрхэд хандаж чадна гэсэн үг биш юм. Энэ ялгааг зөв салгаж ойлгох хэрэгтэй. Учир нь `root` уруу хандах боломжгүй халдагч ерөнхийдөө өөрийн мөрийг баллаж нууж чаддаггүй бөгөөд тухайн хэрэглэгчийн файлуудыг замбараагүйтүүлэх эсвэл машиныг сүйрүүлэхээс илүүтэйг хийж чаддаггүй. Хэрэглэгчид нь сисадминууд шиг аюулгүй байдлын арга хэмжээг тэр болгон авдаггүй болохоор хэрэглэгчийн бүртгэлийн буулган авалт нь маш элбэг байдаг юм.

Машин дээрх `root` бүртгэлийг эвдэх боломжит олон аргууд байдгийг системийн администраторууд санаж байх хэрэгтэй. Халдагч нь `root`-н нууц үгийг мэдэж болно. Эсвэл халдагч root эрхээр ажилладаг серверт алдаа олж сүлжээгээр тэр сервер уруу дамжин орж `root`-г эвдэж болно. Эсвэл халдагч нь suid-root програмд алдаа байгааг мэдэж хэрэглэгчийн бүртгэлийг эвдэн орсныхоо дараа тэр алдаагаар дамжин `root`-г эвдэн орж болох юм. Хэрэв халдагч машин дээрх `root`-г эвдэх аргаа олсон бол заавал арын хаалга суулгах шаардлагагүй болж болох юм. `root`-н цоорхойнуудын олонхийг тухайн үед аль хэдийн олоод хаачихсан байдаг бөгөөд энэ үед халдагчид өөрийн мөрөө цэвэрлэхэд ихээхэн ажиллагаа шаарддаг болохоор ихэнх халдагчид арын хаалга суулгадаг. Арын хаалга нь систем уруу хандах `root` хандалтыг халдагчид амархнаар дахин олж авах боломжийг олгодог боловч энэ нь ухаалаг системийн администраторт халдлагыг амархнаар илрүүлэх боломжийг бас олгодог юм. Халдагчийн хамгийн эхлээд эвдэн орсон цоорхойг хааж чаддаггүй болохоор арын хаалга суулгахыг боломжгүй болгох нь магадгүй таны аюулгүй байдалд ашиггүй байж болох юм.

Аюулгүй байдлын засварууд нь олон давхраатай "сонгины хальс" хандлагаар үргэлж шийдэгдэж байх шаардлагатай бөгөөд тэдгээрийг дараах маягаар зэрэглэж болно:

1. `root` болон staff бүртгэлүүдийг нууцлаг/аюулгүй болгох.
2. `root`-ажилладаг серверүүд болон suid/sgid хоёртын файлуудыг аюулгүй болгох.
3. Хэрэглэгчийн бүртгэлүүдийг аюулгүй болгох.
4. Нууц үгийн файлыг аюулгүй болгох.
5. Цөмийн гол хэсэг, түүхий төхөөрөмжүүд болон файлын системүүдийг аюулгүй болгох.
6. Системд хийгдсэн зохисгүй өөрчлөлтүүдийг түргэн илрүүлэх.
7. Параной буюу хэт зовнил.

Энэ бүлгийн дараагийн хэсэг нь дээр дурдсан зүйлсүүдийг илүү гүнзгийгээр авч үзэх болно.

# 15.3. FreeBSD-н аюулгүй байдлыг хангах нь

*Тушаалыг Протоколтой харьцуулахад (Command vs. Protocol)*

Энэ баримтын туршид бид тод текстээр програмыг `monospaced` фонтоор тусгай тушаалуудыг тэмдэглэх болно. Протоколууд ердийн фонт ашиглах болно. Тэмдэглэгээний энэ ялгаа нь ssh зэргийн хувьд ашигтай, учир нь энэ ssh нь протоколоос гадна бас тушаал юм.

Үүнээс хойшх хэсгүүд нь түрүүчийн бүлгийн сүүлийн хэсэгт дурдсан таны FreeBSD системийг аюулгүй болгох аргуудыг авч үзнэ.

### 15.3.1. root бүртгэл болон staff бүртгэлүүдийг аюулгүй болгох

Эхлээд хэрэв та root бүртгэлийг аюулгүй болгоогүй бол staff бүртгэлүүдийг аюулгүй болгоход санаа зовсны хэрэггүй. Ихэнх системүүд root бүртгэлд нууц үг өгсөн байдаг. Таны эхний хийх зүйл бол нууц үг *үргэлж* эвдэгдэж болно гэдгийг бодох хэрэгтэй. Энэ нь та нууц үгээ устгах хэрэгтэй гэсэн үг биш юм. Нууц үг нь машин уруу консол хандалт хийхэд үргэлж хэрэгтэй байдаг. Энэ нь юу гэсэн үг вэ гэхээр та нууц үгийг консолоос гадна эсвэл болж өгвөл бүр su(1) тушаалтай ашиглаж болохоор хийх ёсгүй гэсэн үг юм. Жишээ нь telnet эсвэл rlogin -p хийгдэх шууд root нэвтрэлтүүдийг хаах pty-уудын тохиргоог insecure буюу аюултай гэж /etc/ttys файлд заасан эсэхийг шалгаарай. Хэрэв бусад нэвтрэх үйлчилгээнүүд болох sshd зэргийг ашиглаж байгаа бол шууд root нэвтрэлтүүдийг бас хаасан эсэхийг шалгаарай. Та үүнийг /etc/ssh/sshd_config файлыг засварлан PermitRootLogin тохируулгыг no болгон зааж өгөөрэй. Хандах арга бүр - FTP зэрэг үйлчилгээнүүдээр ихэвчлэн эвдлэн ордог болохыг бодолцох хэрэгтэй. Шууд root нэвтрэлтүүд зөвхөн системийн консолоор хийгдэхэд зөвшөөрөгдөх ёстой.

Мэдээж систем админы хувьд та root уруу орж чадаж байх ёстой болохоор бид хэдэн цоорхой үлдээдэг. Гэхдээ эдгээр цоорхойнууд нь нэмэлт нууц үг шалгаж ажилладаг байхаар бид хийдэг. root-г хандах боломжтой байлгах нэг арга нь тохирох staff бүртгэлүүдийг wheel бүлэгт (/etc/group файлд) нэмэх явдал юм. wheel бүлэгт оруулсан staff-ийн гишүүдэд root уруу su хийхийг зөвшөөрдөг. Та staff-ийн гишүүдийг тэдгээрийн нууц үгийн оруулгад wheel бүлэгт оруулан байрлуулж анхнаас нь wheel хандалт өгч хэзээ ч болохгүй. Staff бүртгэлүүдийг staff бүлэгт оруулах ёстой бөгөөд тэгээд дараа нь /etc/group файлын wheel бүлэгт нэмэх ёстой. Зөвхөн root хандалт заавал шаардлагатай тийм staff-ийн гишүүдийг wheel бүлэгт оруулах ёстой. Kerberos зэрэг жинхэнээ шалгуулж нэвтрэх аргыг ашиглаж байх тохиолдолд заавал wheel бүлэгт оруулалгүйгээр root бүртгэл дэх Kerberos-ийн .k5login файлыг ашиглаж root уруу ksu(1) хийхийг зөвшөөрөх бас боломжтой байдаг. Энэ нь магадгүй давуу шийдэл байж болох юм. Учир нь хэрэв халдагч таны нууц үгийн файлыг олж аван staff бүртгэлийг эвдлэн орж чадах бол wheel арга нь халдагчид root-г эвдэх боломжийг олгосон хэвээр байдаг юм. wheel аргатай байх нь огт аргагүй байхаас илүү боловч энэ нь заавал ч үгүй хамгийн аюулгүй сонголт бас биш юм.

Бүртгэлийг бүрэн түгжихийн тулд pw(8) тушаалыг ашиглах хэрэгтэй:

```
# pw lock staff
```

Энэ нь ssh(1)-ийг оролцуулаад хэрэглэгчийг ямар ч арга ашиглан нэвтрэн орохыг хориглоно.

Бүртгэлүүдэд хандахыг хориглох өөр нэг арга бол нууцлагдсан нууц үгийг ганц “*” тэмдэгтээр солих явдал юм. Энэ тэмдэгт нь нууцлагдсан нууц үгтэй хэзээ ч таарахгүй бөгөөд хэрэглэгчийн хандалтыг хаах болно. Жишээ нь доор дурдсан staff бүртгэлийг:

```
foobar:R9DT/Fa1/LV9U:1000:1000::0:0:Foo Bar:/home/foobar:/usr/local/bin/tcsh
```

Ийм болгон өөрчлөх хэрэгтэй:

```
foobar:*:1000:1000::0:0:Foo Bar:/home/foobar:/usr/local/bin/tcsh
```

Энэ нь `foobar` хэрэглэгчийг ердийн аргууд ашиглан нэвтрэн орох боломжийг хаадаг. Энэ хандалт хязгаарлах арга нь Kerberos ашиглаж байгаа сайтууд эсвэл хэрэглэгч ssh(1) ашиглан түлхүүрүүд тохируулсан тохиолдлууд зэрэгт ажилладаггүй.

Эдгээр аюулгүй байдлын арга замууд нь бас таныг илүү хязгаарласан серверээс арай бага хязгаарласан машин уруу нэвтрэн орж байна гэж тооцдог. Жишээ нь хэрэв таны гол хайрцаг чинь бүх л төрлийн серверүүд ажиллуулж байвал таны ажлын компьютер чинь ямрыг ч ажиллуулах ёсгүй. Өөрийн компьютерийг боломжийн аюулгүй болгохын тулд та ерөөсөө сервергүй болтол аль болох цөөн сервер ажиллуулах хэрэгтэй бөгөөд та нууц үгээр хамгаалагдсан дэлгэц хооослогч ажиллуулах хэрэгтэй. Мэдээж ажлын компьютер уруу физик хандалт өгвөл халдагч ямар ч төрлийн аюулгүй байдлыг та хангасан байлаа гэсэн эвдэж чадна. Энэ нь таны бодох ёстой асуудлын нэг юм. Гэхдээ эвдлэн оролтуудын олонхи нь алсаас сүлжээгээр дамжин таны ажлын компьютер эсвэл серверүүдэд физик хандалт байхгүй хүмүүсээс ирдэг гэдгийг та бас л бодолцох хэрэгтэй юм.

Kereberos мэтийг ашиглах нь танд staff бүртгэлийн нууц үгийг нэг газар өөрчлөх эсвэл хаах боломжийг олгох бөгөөд staff-ийн гишүүдийн бүртгэл байж болох бүх машинууд дээр нэн даруй бас үйлчилдэг. Хэрэв staff-ийн гишүүний бүртгэл эвдэгдсэн бол түүний нууц үгийг бүх машинууд дээр нэн даруй өөрчлөх тэр боломжийг дутуу үнэлэх ёсгүй юм. Тусдаа байгаа нууц үгүүдийг N машинууд дээр өөрчлөх нь зовлонтой байдаг. Мөн та Kerberos-д нууц үг дахин өгөлтийг ноогдуулж болох бөгөөд Kerberos тасалбарыг хэсэг хугацааны дараа дуусдагаар хийж болохоос гадна Kerberos систем нь тодорхой хугацааны (жишээ нь сар бүр) дараа хэрэглэгчийг шинэ нууц үг сонгохыг шаарддагаар бас тохируулж болдог.

### 15.3.2. root-ажилладаг серверүүд болон suid/sgid хоёртын файлуудыг аюулгүй болгох

Хянамгай сисадмин илүү ч үгүй дутуу ч үгүй зөвхөн өөрийн хэрэгтэй серверүүдийг ажиллуулдаг. Гуравдагч талын серверүүд ихэвчлэн хамгийн алдаатай байх хандлагатай гэдгийг санаж байх хэрэгтэй. Жишээ нь imapd эсвэл popper серверийн хуучин хувилбарыг ажиллуулна гэдэг нь универсал `root` тасалбарыг бүх дэлхийд өгч байна гэсэн үг юм. Та няхуур шалгаагүй сервер битгий ажиллуул. Олон серверүүд заавал `root` эрхээр ажиллах шаардлагагүй байдаг. Жишээ нь ntalk, comsat, болон finger дэмонуудыг тусгай хэрэглэгчийн *sandboxes буюу хамгаалагдсан хязгаарлагдмал орчинд* ажиллуулах боломжтой байдаг. Хамгаалагдсан хязгаарлагдмал орчин нь асар их төвгүүдийг давж хийгээгүй л бол төгс биш бөгөөд өмнө дурдсан сонгины хандлагаар аюулгүй байдалд хандах нь хэвээр байна: хэрэв хэн нэгэн нь хамгаалагдсан хязгаарлагдмал орчинд ажиллаж байгаа серверт эвдэн орж чадсан ч гэсэн тэд хамгаалагдсан хязгаарлагдмал орчныг бас эвдэн гарах хэрэг болно. Аль болох олон давхаргыг халдагч эвдлэх ёстой болох тусам тэдгээрийн амжилттай болох нь улам багасах болно. Урьд нь root цоорхойнууд нь системийн үндсэн серверүүдээс авахуулаад бараг л бүх `root` ажилладаг сервер дээр олдож байсан. Хэрэв таны ажиллуулдаг машин уруу хүмүүс зөвхөн sshd ашиглан нэвтэрдэг бөгөөд telnetd, rshd эсвэл rlogind хэзээ ч ашиглан нэвтэрдэггүй бол эдгээр үйлчилгээнүүдийг хаагаарай!

Одоо FreeBSD нь ntalkd, comsat, болон finger үйлчилгээнүүдийг хамгаалагдсан

хязгаарлагдмал орчинд анхдагчаар ажиллуулдаг. Хамгаалагдсан хязгаарлагдмал орчинд ажиллуулж болох өөр нэг програм нь named(8) юм. /etc/defaults/rc.conf нь named-г хамгаалагдсан хязгаарлагдмал орчинд ажиллуулахад шаардлагатай нэмэлт өгөгдлүүдийг тайлбар хэлбэрээр агуулсан байдаг. Таны шинэ систем эсвэл байгаа системээ шинэчилж байгаагаас хамааран тэдгээр хамгаалагдсан хязгаарлагдмал орчинд ашиглагдах тусгай хэрэглэгчийн бүртгэлүүд суулгагдаагүй байж болох юм. Хянамгай сисадмин судалгаа хийж серверүүдийг хамгаалагдсан хязгаарлагдмал орчинд аль болох ажиллуулдаг.

Хамгаалагдсан хязгаарлагдмал орчинд ерөнхийдөө ажилладаггүй хэд хэдэн серверүүд байдаг: sendmail, popper, imapd, ftpd, болон бусад. Эдгээрийн зарим шиг бас өөр серверүүд байдаг боловч тэдгээрийг суулгах нь таны хүсэж байгаагаас илүү (амархан байх гэсэн асуудал энд сөхөгдөж байна) их ажиллагаа шаардаж магадгүй юм. Та эдгээр серверүүдийг магадгүй `root` эрхээр ажиллуулж тэдгээрт учирч болох эвдрэн оролтуудыг илрүүлэх өөр арга замуудад найдах хэрэгтэй болж болох юм.

Системийн өөр нэг том боломжтой `root` цоорхойнууд бол системд суусан suid-root болон sgid хоёртын файлууд юм. rlogin зэрэг эдгээрийн ихэнх нь /bin, /sbin, /usr/bin, эсвэл /usr/sbin сангуудад байрладаг. Юу ч 100% аюулгүй байдаггүй боловч системийн анхдагч suid болон sgid хоёртын файлууд нь боломжийн хэрээр аюулгүй гэж тооцогддог. Гэсэн хэдий ч эдгээр хоёртын файлуудад `root` цоорхойнууд үе үе олддог. xterm-г (энэ нь ихэвчлэн suid байдаг) эмзэг болгосон `root` цоорхойнууд 1998 онд `Xlib`-д олджээ. Харамсахаасаа өмнө аюулгүй байж байсан нь дээр учраас хянамгай сисадмин зөвхөн staff ажиллуулах ёстойгоор staff зөвхөн хандаж чадах тусгай бүлэгт зөвшөөрч suid хоёртын файлуудыг хязгаарладаг бөгөөд хэн ч ашигладаггүй suid хоёртын файлуудыг ажиллуулж болохгүй болгодог (`chmod 000`). Дэлгэцгүй серверт ер нь xterm хоёртын файл хэрэгцээгүй юм. Sgid хоёртын файлууд нь бас л аюултай юм. Хэрэв халдагч sgid-kmem хоёртын файлыг эвдэж чадвал тэр /dev/kmem-г уншиж чадах бөгөөд ингэснээр нууц үгтэй дурын бүртгэлийг эвдэн орж шифрлэсэн нууц үгийн файлыг уншихад хүргэдэг. Бас `kmem` бүлгийг эвдсэн халдагч secure буюу аюулгүй аргаар дамжин нэвтрэн орсон хэрэглэгчдийн ашиглаж байгаа pty-уудаар илгээгдсэн гарын товчнуудын даралтуудыг хянаж чаддаг. `tty` бүлгийг эвдсэн халдагч бараг дурын хэрэглэгчийн tty-д бичиж чадна. Хэрэв хэрэглэгч гар дуурайх боломж бүхий терминал програм эсвэл эмулятор ажиллуулж байгаа бол хэрэглэгчийн терминалыг тушаал буцаан харуулахаар болгодог өгөгдлийн урсгалыг халдагч үүсгэж дараа нь тэр тушаалыг тэр хэрэглэгчийн эрхээр ажиллуулдаг.

### 15.3.3. Хэрэглэгчийн бүртгэлүүдийг аюулгүй болгох

Хэрэглэгчийн бүртгэлүүдийг аюулгүй болгох нь ихэвчлэн хамгийн хэцүү байдаг. Та өөрийн staff-д ширүүн хандалтын хязгаарлалтууд оногдуулж тэдгээрийн нууц үгүүдийг "од болгож" болох боловч та ердийн хэрэглэгчийн бүртгэлүүдийг яг ингэж хязгаарлаж чадахгүй байж болох юм. Хэрэв та хангалттай хяналттай байх юм бол таны аз болж хэрэглэгчийн бүртгэлүүдийг зөвөөр аюулгүй болгож чадна. Хэрэв үгүй бол та тэдгээр бүртгэлүүдийг хянахдаа ердөө л илүү сонор сэрэмжтэй байх хэрэгтэй. ssh болон Kerberos-г хэрэглэгчийн бүртгэлүүдэд ашиглах нь нэмэлт удирдлага болон техникийн дэмжлэг шаардлагатайгаас болоод илүү асуудалтай байдаг боловч энэ нь шифрлэсэн нууц үгийн файлыг бодох юм бол маш сайн шийдэл хэвээр байдаг.

### 15.3.4. Нууц үгийн файлыг аюулгүй болгох

Цорын ганц итгэлтэй арга бол аль болох олон нууц үгүүдийг од болгон тэдгээр бүртгэлүүдэд хандахын тулд ssh эсвэл Kerberos ашигла. Шифрлэгдсэн нууц үгийн файлыг (/etc/spwd.db) зөвхөн `root` уншиж чаддаг боловч халдагч root-бичих хандалт олж авч чадаагүй ч гэсэн тэр файлд унших эрх олж авах боломжтой байж болох юм.

Таны аюулгүй байдлын скриптүүд нууц үгийн файлд хийгдсэн өөрчлөлтүүдийг үргэлж шалгаж тайлагнах шаардлагатай (доорх Файлын бүрэн бүтэн байдлыг шалгах хэсгийг үзнэ үү).

### 15.3.5. Цөмийн гол хэсэг, түүхий төхөөрөмжүүд болон файлын системүүдийг аюулгүй болгох

Хэрэв халдагч `root`-г эвдсэн бол тэр юуг ч хийж чадах боловч зарим ашиг сонирхлууд байдаг. Жишээ нь орчин үеийн ихэнх цөмүүдэд пакет шиншлэх төхөөрөмжийн драйвер бүтээгдсэн байдаг. FreeBSD-д энэ нь bpf төхөөрөмж гэж нэрлэгддэг. Халдагч ердөө буулган авсан машин дээрээ пакет шиншлэгчийг ажиллуулахыг оролддог. Та халдагчид энэ боломжийг өгөх хэрэггүй бөгөөд ихэнх системүүдэд bpf төхөөрөмжийг эмхэтгэн оруулах шаардлагагүй юм.

Гэхдээ bpf төхөөрөмжийг хаасан ч гэсэн та /dev/mem болон /dev/kmem файлуудад бас санаа тавих хэрэгтэй. Энэнээс болоод халдагч түүхий (raw) төхөөрөмжүүдэд бичиж чадсан хэвээр байна. Мөн цөмийн бас нэг боломж болох модуль ачаалагч гэж нэрлэгддэг kldload(8) байдаг. Самбаатай халдагч KLD модуль ашиглаад өөрийн bpf төхөөрөмж эсвэл бусад шиншлэх төхөөрөмжийг ажиллаж байгаа цөмд суулгадаг. Эдгээр асуудлуудаас зайлсхийхийн тулд та цөмийг илүү өндөр аюулгүй байдлын түвшинд ядаж аюулгүйн түвшин 1-д ажиллуулах хэрэгтэй.

Цөмийн аюулгүй байдлын түвшинг янз бүрийн аргаар тохируулж болно. Ажиллаж байгаа цөмийн аюулгүй байдлын түвшинг нэмэгдүүлэх хялбар алга бол цөмийн `kern.securelevel` хувьсагчийг `sysctl` ашиглан өөрчлөх явдал юм:

```
# sysctl kern.securelevel=1
```

Анхдагчаар FreeBSD цөм аюулгүй байдлын -1 түвшинтэй ачаалдаг. Аюулгүй байдлын түвшинг администратор эсвэл эхлүүлэх скриптүүд дэх тохиргооноос болоод init(8)-ээр өөрчлөөгүй л бол -1 хэвээр байх болно. /etc/rc.conf файлд `kern_securelevel_enable` хувьсагчийг `YES` ба `kern_securelevel` хувьсагчийн утгыг аюулгүй байдлын хүссэн түвшин рүүгээ болгон тохируулж системийг эхлүүлэх үед аюулгүй байдлын түвшинг нэмэгдүүлж болно.

Эхлүүлэх скриптүүд дөнгөж дуусаад байх үед FreeBSD системийн аюулгүй байдлын анхдагч түвшин -1 байдаг. Үүнийг "insecure mode" буюу "аюулгүй байдлыг хангаагүй горим" гэдэг бөгөөд учир нь хувиршгүй байлын тугуудыг болиулах, бүх төхөөрөмжөөс уншиж эсвэл тэдгээр рүү бичих гэх зэргийг хориогүй байдаг.

Аюулгүй байдлын түвшинг 1 эсвэл илүү өндөр утгаар тохируулсны дараа зөвхөн нэмэх болон хувиршгүй файлууд идэвхжиж тэдгээрийг болиулах боломжгүй болон түүхийн төхөөрөмжүүдэд хандахыг хориглодог. Илүү өндөр түвшингүүд бүр илүү олон үйлдлүүдийг

хязгаарладаг. Төрөл бүрийн аюулгүй байдлын түвшнүүдийн үйлчилгээний талаарх дэлгэрэнгүй тайлбарыг security(7) гарын авлагын хуудсыг уншина уу.

Аюулгүйн түвшинг 1 эсвэл илүү өндөр түвшнээр дээшлүүлэх нь X11 (/dev/io руу хандах хандалт хаалттай байна) эсвэл FreeBSD-ийн бүтээлтийг эхээс суулгах (процессын `installworld` хэсэг зарим файлуудын зөвхөн нэмэгдэх болон хувиршгүй тугуудыг түр зуур өөрчлөхийг шаарддаг) болон бусад цөөн тохиолдлуудын хувьд асуудлууд гаргаж болох юм. Заримдаа, жишээ нь X11-ийн хувьд ачаалах явцад xdm(1)-ийг нэлээн эрт аюулгүйн түвшин бага байгаа үед нь ажиллуулж энэ асуудлыг тойрон гарах боломжтой байж болох юм. Үүнтэй адил тойрон гарах арга замууд нь бүх аюулгүй байдлын түвшингүүд эсвэл тэдгээрийн мөрдөж шаарддаг боломжит бүх хязгаарлалтуудын хувьд боломжтой биш байж болох юм. Урьдчилаад бага зэрэг төлөвлөх нь зүйтэй байдаг. Аюулгүйн түвшин бүр системийн хэрэглээг нэлээн багасгах боломжтой байдаг учир тэдгээртэй хамааралтай хязгаарлалтуудыг ойлгох нь чухал юм. Энэ нь бас анхдагч тохиргоог сонгохыг илүү хялбар болгож санамсаргүй явдлаас урьдчилан сэргийлэх болно.

Хэрэв цөмийн аюулгүйн түвшин 1 эсвэл түүнээс илүү утгаар дээшлүүлэгдсэн бол `schg` тугийг чухал эхлүүлэх хоёртын файлууд, сангууд болон скрипт файлууд (өөрөөр хэлбэл аюулгүйн түвшин тохируулагдах хүртэлх ажиллах бүх файлууд) дээр тохируулах нь ашигтай байж болох юм. Энэ нь хэтэрхий хийгдэж байж болох бөгөөд аюулгүйн өндөр түвшинд ажиллаж байхад системийг шинэчлэх үйл явцыг илүү хэцүү болгодог. Арай бага хязгаарлалттай өөр нэг боломж нь системийг илүү өндөр аюулгүйн түвшинд ажиллуулж гэхдээ `schg` тугийг системийн файл болон сан бүр дээр тохируулахгүй байх явдал юм. Өөр нэг боломж нь / болон /usr санг зөвхөн уншигдахаар холбох явдал юм. Юу зөвшөөрөгдсөн байх дээр хэтэрхий чанга байх нь халдлага илрүүлэлтийн бүх чухал зүйлсийг хязгаарлаж болох юм.

### 15.3.6. Файлын бүрэн бүтэн байдлыг шалгах нь: Хоёртын файлууд, Тохиргооны файлууд, гэх мэт.

Тэр мөч ирэхэд, та зөвхөн системийн гол тохиргоо болон хяналтын файлуудаа ая тухын хүчин зүйл урьтахаас хамаагүй өмнө хамгаалж чадна. Жишээ нь `chflags` тушаал ашиглан / болон /usr сангууд дахь ихэнх файлуудад `schg` битийг тохируулах нь магадгүй үр ашиггүй байж болох бөгөөд учир нь ингэснээр файлуудыг хамгаалахын хажуугаар бас илрүүлэх цонхыг хаадаг юм. Таны аюулгүй байдлын сонгины сүүлийн давхарга нь илрүүлэлт бөгөөд энэ нь хамгийн чухал юм. Хэрэв та боломжит халдагчдыг илрүүлж чадахгүй л бол аюулгүй байдлын бусад үлдсэн асуудлуудын талаар бодоод ч бараг хэрэггүй юм (эсвэл бүр дэмий юм, аюулгүй байдлыг танд буруу ойлгуулахад хүргэдэг). Сонгины ажлын хагас нь халдагчийг үйлдэл дээр нь барихын тулд түүнийг зогсоохын оронд харин удаашруулах явдал юм.

Халдлагыг илрүүлэх хамгийн сайн арга бол өөрчлөгдсөн, алга болсон, эсвэл гэнэтийн файлуудыг хайх явдал юм. Өөрчлөгдсөн файлуудыг хайх хамгийн сайн арга бол тэдгээрийг өөр (ихэвчлэн төвлөрсөн) хязгаарлагдмал хандалттай системээс хайх явдал юм. Өөрийн аюулгүй байдлын скриптийг нэмэлт аюулгүй байдал хангасан хязгаарлагдмал хандалттай систем дээр бичих нь тэдгээрийг боломжит халдагчдад бараг харагдуулдаггүй бөгөөд энэ нь чухал юм. Давуу талыг хамгийн ихээр авахын тулд ерөнхийдөө хязгаарлагдмал хандалттай

хайрцагт бусад машинуудад хандах тэр ач холбогдолтой хандалтыг өгөх хэрэгтэй. Үүнийг ихэвчлэн бусад машинуудын зөвхөн унших NFS экспортыг хязгаарлагдмал хандалттай хайрцагт өгөх эсвэл ssh түлхүүр хослолыг тохируулж хязгаарлагдмал хандалттай хайрцгийг бусад машинууд уруу ssh хийхийг зөвшөөрөх замаар хийдэг. Өөрийн сүлжээний урсгалыг тооцохгүй юм бол NFS нь хамгийн харагддаггүй арга юм - энэ нь клиент хайрцаг бүр дэх файлын системүүдийг монитор хийхийг танд зөвшөөрч бараг л илэрдэггүй. Хэрэв таны хязгаарлагдмал хандалттай сервер нь клиент хайрцагнууд уруу hub буюу салаалагч эсвэл чиглүүлэлтийн хэд хэдэн давхаргаар дамжин холбогдсон бол NFS арга нь хэтэрхий аюултай (сүлжээний хувьд) байж болох бөгөөд ssh-ийг ашиглах нь түүний гаргадаг аудит мөрийн замуудтай байсан ч гэсэн магадгүй илүү сонголт байж болох юм.

Монитор хийгдэх клиент систем уруу хандахад хамгийн багаар бодоход унших эрхийг та хязгаарлагдмал хандалттай хайрцагт өгсний дараа яг мониторыг хийхдээ скрипт бичих хэрэгтэй. Өгөгдсөн NFS холболтод find(1) болон md5(1) зэрэг энгийн системийн хэрэгслүүд ашиглан та скриптүүд бичиж болно. Клиент хайрцгийн файлуудад өдөрт нэг удаа физикээр md5 хийж /etc болон /usr/local/etc сангууд дахь хяналтын файлуудыг бүр илүү давтамжтайгаар шалгаж байх нь зүйтэй юм. Хязгаарлагдмал хандалттай машины зөв гэж тооцсон md5 мэдээлэлтэй харьцуулахад тарахгүй файлууд олдвол сисадминд үүнийг очиж шалгахыг хашгиран мэдээлэх ёстой. Аюулгүй байдлын сайн скрипт нь тохирохгүй suid хоёртын файлууд болон / болон /usr зэрэг системийн хуваалтууд дээрх шинээр үүссэн эсвэл устгагдсан файлуудыг бас шалгадаг.

NFS биш ssh-ийг ашиглаж байх үед аюулгүй байдлыг скрипт бичих нь бүр илүү хэцүү байдаг. Та скриптүүдийг харагдуулж ажиллуулахын тулд тэдгээрийг клиент хайрцаг уруу үндсэндээ `scp` хийх хэрэгтэй бөгөөд аюулгүй байдлаа бодох юм бол та тэдгээр скриптүүдийн ашигладаг хоёртын файлуудыг (find гэх зэрэг) бас `scp` хийх хэрэгтэй юм. Клиент хайрцаг дээрх ssh клиент аль хэдийн эвдэгдсэн байж болох юм. Аюултай холболтоор ажиллаж байгаа бол ssh-г ашиглах нь шаардлагатай байж болох боловч бас түүнтэй ажиллахад бүр илүү хэцүү байдаг юм.

Аюулгүй байдлын сайн скрипт нь .rhosts, .shosts, .ssh/authorized_keys гэх зэрэг `MD5` шалгалтын хүрээний гадуур байх хэрэглэгч болон staff-ийн гишүүдийн хандалтын тохиргооны файлууд дахь өөрчлөлтүүдийг бас шалгадаг.

Хэрэв та асар их хэрэглэгчийн дискний зайтай бол тэдгээр хуваалтууд дээр байгаа файл бүр дээр ажиллахад хэт удаж болох юм. Энэ тохиолдолд suid хоёртын файлуудыг хаах холболтын тугуудыг зааж өгөх нь зүйтэй юм. `nosuid` нь таны хайж байгаа тэр тохируулга юм. Энэ давхаргын зорилго нь эвдлэн оролтын оролдлогуудыг амжилттай эсвэл амжилтгүй болсноос үл хамааран илрүүлэх явдал учраас ямар ч гэсэн ядаж долоо хоногт нэг удаа та тэдгээр файлуудыг магадгүй шалгаж байх хэрэгтэй юм.

Процессийн бүртгэл хийх нь (accton(8)-г үзнэ үү) эвдлэн оролтын дараах үнэлэх арга замууд болон тусалж болох харьцангуй бага ачаалал бүхий үйлдлийн системийн боломж юм. Энэ нь эвдлэн орсны дараа файлыг хөндөөгүй хэвээр гэж үзэн халдагч систем уруу хэрхэн эвдлэн орсныг мөрдөхөд ялангуяа ашигтай байдаг.

Эцэст нь аюулгүй байдлын скриптүүд нь бүртгэлийн файлуудыг процесс хийх ёстой бөгөөд бүртгэлүүд өөрсдөө аль болох аюулгүй байдлаар үүсгэгдэх ёстой бөгөөд алсын syslog нь их ашигтай байж болох юм. Халдагч өөрийн мөрийг арилгахыг оролдох бөгөөд эхний эвдлэн

оролтын арга болон хугацааг мөрдөхөд сисадмины хувьд бүртгэлийн файлууд нь маш чухал байдаг юм. Бүртгэлийн файлуудын байнгын бичлэгийг хадгалах нэг арга нь системийн консолыг сериал порт уруу ажиллуулж консолуудыг хянаж аюулгүй машин дээр мэдээллийг цуглуулах явдал юм.

### 15.3.7. Параной буюу хэт зовнил

Бага зэргийн хэт зовнил буруудахгүй. Дүрэм болгож тав тухтай байдлыг алдагдуулдаггүй дурын тооны аюулгүй байдлын боломжуудыг сисадмин нэмж болох бөгөөд зарим анхаарлыг бодолцон тав тухтай байдалд *нөлөөлөх* аюулгүй байдлын боломжуудыг бас нэмж болох юм. Бүр илүү чухал нь аюулгүй байдлын администратор үүнийг бага зэрэг хольж хэрэглэж болно - хэрэв та энэ баримтад дурдсан зааврууды үгчлэн ашиглавал энэ баримтыг уншсан ирээдүйн халдагчид та өөрийн арга замуудыг заан өгч байна гэсэн үг юм.

### 15.3.8. Үйлчилгээг Зогсоох Халдлагууд

Энэ хэсэг нь Үйлчилгээг Зогсоох халдлагуудыг хамарна. DoS халдлага нь ихэвчлэн пакетийн халдлага байдаг. Таны сүлжээг дүүргэж байгаа орчин үеийн хууран мэхэлсэн пакетийн халдлагуудын эсрэг нэг их юм хийж чадахгүй ч гэсэн халдлагууд таны серверүүдийг унагахгүйн тулд та ерөнхийдөө хохирлыг хязгаарлаж болно:

1. Серверийн fork хийлтийг хязгаарлах.
2. Springboard буюу бусад халдлагуудыг хязгаарлах (ICMP хариу халдлагууд, ping цацалт, гэх мэт.).
3. Цөмийн чиглүүлэлтийн кэшийг хэт ачаалах.

Нийтлэг DoS халдлагын дүр зураг бол fork хийгдэж байгаа серверт халдаж түүнээр асар их хүүхэд процесс үүсгүүлж эцсийн эцэст хост системийн хувьд санах ой, файлын тодорхойлогчууд гэх мэтүүд дуусч зогсоход хүргэдэг. inetd (inetd(8)-г үзнэ үү) нь энэ төрлийн халдлагыг хязгаарлах хэд хэдэн тохируулгатай. Машиныг зогсоохоос хамгаалах боломжтой боловч ерөнхийдөө үйлчилгээг халдлагад өртүүлэхгүй байх боломжгүйг энд тэмдэглэх нь зүйтэй юм. inetd гарын авлагын хуудсыг анхааралтай уншиж `-c`, `-C`, болон `-R` тохируулгуудад ялангуяа анхаарлаа хандуулаарай. Хууран мэхэлсэн IP халдлагууд нь inetd дахь `-C` тохируулгыг хуурах учраас ихэвчлэн тохируулгуудын хослолыг ашиглах шаардлагатай. Зарим дан серверүүд өөрийн fork хийгдэхийг хязгаарлах параметрүүдтэй байдаг.

Sendmail нь `-OMaxDaemonChildren` тохируулгатай байдаг бөгөөд энэ нь Sendmail-ийг ачаалал хязгаарлах тохируулгатай ажиллуулж ачааллын хоцрогдол үүсгэснээс хавьгүй илүүтэйгээр ажилладаг. Та Sendmail-г ажиллуулахдаа хүссэн ачааллыг даахаар гэхдээ компьютерийг унагахаар их хэмжээний тоогоор Sendmail-үүдийг ажиллуулах биш түүнээс багаар `MaxDaemonChildren` параметрийг хангалттай өндрөөр тавьж өгөх хэрэгтэй. Мөн sendmail-ийг дарааллын горимоор (`-ODeliveryMode=queued`) ажиллуулах болон дэмонг (`sendmail -bd`) дараалалтай (`sendmail -q15m`) ажиллуулдгаас тусад нь ажиллуулах нь чухал юм. Хэрэв та шууд илгээх горимыг хүсэж байгаа бол та дарааллыг `-q1m` зэргээр бүр бага интервалаар ажиллуулах боломжтой боловч `MaxDaemonChildren` тохируулгыг боломжийн утгаар хоорондоо холбоотой амжилтгүйтлүүдээс sendmail-ийг хамгаалахын *тулд* зааж өгсөн эсэхээ шалгаарай.

Syslogd-д шууд халдаж болох учраас аль болох `-s` тохируулгыг эсвэл `-a` тохируулгыг ашиглахыг танд зөвлөдөг.

Шууд халдлага хийгдэж болох TCP Wrapper-ийн буцах identd зэрэг буцан холбогддог үйлчилгээнүүдийн хувьд та маш хянамгай байх хэрэгтэй. Ийм учраас та TCP Wrapper-ийн буцах identd боломжийг ерөнхийдөө ашиглах хэрэггүй юм.

Та өөрийн захын чиглүүлэгчүүд дээрээ дотоод үйлчилгээнүүд уруугаа гаднаас хандуулахгүй болгож галт ханаар хамгаалах нь зүйтэй юм. Үүний цаадах санаа нь гаднаас ирж болзошгүй сүлжээ дүүргэх халдлагаас өөрийн LAN-г хамгаалах явдал бөгөөд сүлжээн дээр тулгуурласан `root` эрхийг буулгахаас дотоод үйлчилгээнүүдийг хамгаалах зүйлс тийм их биш юм. exclusive буюу хамааруулаагүй галт ханыг үргэлж тохируулах хэрэгтэй, өөрөөр хэлбэл "A, B, C, D болон M-Z портуудаас *бусад* бүгдийг галт ханаар хамгаалах хэрэгтэй". Ингэснээр та named (хэрэв та бүсийн хувьд анхдагч бол), ntalkd, sendmail болон бусад Интернэтээс хандах үйлчилгээнүүд зэрэг зарим нэг тусгай үйлчилгээнүүдийн портуудаас бусад бүх бага дугаарын портуудыг галт ханаар хамгаалж чадах юм. Хэрэв та галт ханыг өөр аргаар - inclusive буюу хамааруулсан эсвэл зөвшөөрсөн галт хана маягаар тохируулахыг оролдвол хэд хэдэн үйлчилгээнүүдийг "хаахаа" мартаж магадгүй юм, эсвэл та шинэ дотоод үйлчилгээ нэмээд галт ханаа шинэчлэхээ мартаж болох юм. Та галт хана дээр зөвшөөрсөнтэй адил үйлдлийг нэвтрүүлэхийн тулд бага дугаарын портуудыг нээлгүйгээр өндөр дугаарын портуудыг онгойлгож болох юм. Мөн FreeBSD нь динамик холболтод хэрэглэгддэг портуудыг `sysctl`-ийн төрөл бүрийн `net.inet.ip.portrange` хувьсагчуудаар (`sysctl -a | fgrep portrange`) хянах боломжийг танд олгодгийг бас тэмдэглэх нь зүйтэй юм. Энэ нь бас таны галт ханын тохиргооны төвөгтэй байдлыг амарчилдаг юм. Жишээ нь та ердийн 4000-аас 5000 хүртэлх портууд болон 49152-оос 65535 хүртэлх өндөр дугаарын портуудыг ашигладаг бол 4000-аас бага бүгдийг өөрийн галт хана дээр хаах хэрэгтэй (мэдээж Интернэтээс ханддаг хэдэн тусгай портуудаас бусад).

Өөр нийтлэг DoS халдлагуудын нэг нь springboard халдлага юм - сервер, дотоод сүлжээ эсвэл бусад машиныг хариу үйлдэл хийхийг нь ихэсгэж хэт ачаалахад хүргэдэг халдлага юм. Ийм маягийн хамгийн нийтлэг халдлага нь *ICMP ping broadcast буюу цацалт* юм. Халдагч таны LAN-ий цацах хаяг руу илгээсэн ping пакетийнхаа эхлэл IP хаягийг халдахыг хүсэж байгаа машиныхаа IP хаягаар сольж хуурдаг. Хэрэв таны захын чиглүүлэгчүүд цацах хаяг руу илгээх ping пакетуудыг зогсоохоор тохируулагдаагүй бол таны LAN хангалттай хариу үүсгэн хууран мэхэлсэн эхлэл хаяг руу илгээж, ялангуяа халдагч хэдэн арван цацах хаягууд руу өөр өөр хэдэн арван сүлжээнүүдээр дамжин энэ башир аргаа ашигласан үед, хохирогчийг дүүргэдэг. 120 мегабайтаас илүү хэмжээний цацах халдлага одоогоор хэмжигдээд байна. Энэ төрлийн хоёр дахь нийтлэг халдлага нь ICMP-ийн алдаа тайлагнах системийн эсрэг халдлага юм. ICMP алдааны мэдэгдэл үүсгэдэг пакетуудыг бүтээж халдагч серверийн орж ирж байгаа сүлжээг дүүргэж ингэснээр серверийг өөрийн гарах сүлжээг ICMP хариунуудаар дүүргэхэд хүргэдэг. Энэ төрлийн халдлага нь ялангуяа хэрэв сервер үүсгэж байгаа ICMP хариунуудаа хангалттай хурднаар шавхан гаргаж чадахгүй байгаа бол серверийг санах ойгүй болгож сүйрүүлж бас болох юм. sysctl-ийн `net.inet.icmp.icmplim` хувьсагчийг ашиглан эдгээр халдлагуудыг хязгаарлах хэрэгтэй. Springboard төрлийн халдлагуудын сүүлийн гол ангилал нь udp цуурай үйлчилгээ зэрэг зарим дотоод inetd үйлчилгээнүүдтэй холбоотой юм. Халдагч UDP пакетийг хууран мэхэлж A болон B сервер нь хоёулаа таны LAN-д байгаа тийм A серверийн цуурай порт дээрх эхлэл хаягаар болон төгсгөл хаягийг B серверийн цуурай порт дээрх хаягаар сольдог. Уг хоёр сервер дараа нь энэ ганц пакетийг хоорондоо шидэлцдэг.

Эдгээр серверүүд болон тэдгээрийн LAN-г энэ маягаар халдагч хэдхэн пакетуудыг хатган оруулан хэт ачаалж чаддаг. Үүнтэй адил асуудлууд дотоод chargen портод бас байдаг. Чадварлаг сисадмин эдгээр бүх дотоод inetd тест үйлчилгээнүүдийг хаадаг.

Хууран мэхэлсэн пакетийн халдлагуудыг цөмийн чиглүүлэлтийн кэшийг хэт ачаалахад хэрэглэж болдог. `net.inet.ip.rtexpire`, `rtminexpire`, болон `rtmaxcache sysctl` параметрүүдийг үзнэ үү. Дурын эхлэл IP хаягийг ашигласан хууран мэхэлсэн пакетийн халдлага нь чиглүүлэлтийн хүснэгтэд түр зуур кэш хийгдсэн чиглүүлэлтийг цөмөөр үүсгүүлэхэд хүргэдэг бөгөөд энэ нь `netstat -rna | fgrep W3` тушаалаар харагддаг. Эдгээр чиглүүлэлтүүд нь ихэвчлэн 1600 секунд орчим хугацааны дотор дуусдаг. Хэрэв цөм кэш хийгдсэн чиглүүлэлтийн хүснэгт хэтэрхий том болсныг илрүүлэх юм бол `rtexpire` динамикаар багасгадаг боловч `rtminexpire`-с бага болтол хэзээ ч багасгадаггүй. Хоёр асуудал байдаг:

1. Бага ачаалагдсан сервер гэнэт халдлагад өртөхөд цөм хангалттай хурдан хариу үйлдэл хийдэггүй.
2. `rtminexpire` хувьсагч нь үргэлжилсэн халдлагыг цөм дааж чадахаар хангалттай бага байдаггүй.

Хэрэв таны серверүүд Интернэтэд Т3 эсвэл илүү хурдаар холбогдсон бол sysctl(8)-оор `rtexpire` болон `rtminexpire` хувьсагчуудыг хоёуланг гараар дарж бичихдээ хянамгай байх хэрэгтэй. Аль ч параметрийг (машиныг сүйрүүлэхийг та хүсээгүй л бол) хэзээ ч битгий 0 болгоорой. Эдгээр параметрүүдийг хоёуланг нь 2 секунд болгох нь чиглүүлэлтийн хүснэгтийг халдлагаас хамгаалахад хангалттай байх ёстой.

### 15.3.9. Kerberos болон SSH-тэй холбоотой хандалтын асуудлууд

Хэрэв та Kerberos болон ssh-г хоёуланг ашиглахаар бол цөөн хэдэн асуудлуудыг дурдах хэрэгтэй. Kerberos 5 нь жинхэнийг шалгах маш сайн нэвтрэлтийн протокол боловч түүнийг ашигласан telnet болон rlogin-д байдаг алдаанууд нь энэ хоёр програмыг хоёртын урсгалтай ажиллахад тохиромжгүй болгодог. Мөн `-x` тохируулгыг ашиглахгүй л бол анхдагчаар Kerberos нь сессийг шифрлэдэггүй. ssh нь бүгдийг шифрлэдэг.

Ssh нь анхдагчаар шифрлэсэн түлхүүрүүдээ дамжуулдгаас бусад бүх л талаараа зэгсэн сайн ажилладаг. Энэ нь юу гэсэн үг вэ гэхээр та хэрэв системийн бусад хэсэгт хандах боломж олгодог түлхүүрүүд бүхий аюулгүй ажлын компьютертай бөгөөд та аюултай машин уруу ssh хийвэл таны түлхүүрүүд ашиглагдах боломжтой гэсэн үг юм. Яг түлхүүрүүд нь өөрсдөө ил гардаггүй боловч ssh нь таны нэвтэрсэн хугацааны туршид зориулж дамжуулах порт суулгадаг бөгөөд хэрэв халдагч аюулгүй машин дээрх `root`-г эвдсэн бол тэрхүү портыг таны түлхүүрүүдийг ашиглахын тулд хэрэглэн таны түлхүүрээр тайлагдах өөр бусад машинуудад хандах боломжийг олж авах боломжтой юм.

Бид staff нэвтрэлтүүдийн хувьд аль болох ssh-г Kerberos-той цуг ашиглахыг зөвлөдөг. Ssh нь Kerberos-ийн дэмжлэгтэй эмхэтгэгдэж болдог. Энэ нь ил гарсан байж болзошгүй ssh түлхүүрүүдэд найдах таны найдварыг багасгахын хамт нууц үгүүдийг Kerberos-оор хамгаалдаг. Ssh түлхүүрүүд нь аюулгүй машинуудын автоматчилагдсан ажлуудад (Kerberos-оор хийхэд таарахгүй) зөвхөн хэрэглэгдэх ёстой. Мөн бид таныг ssh-ийн тохиргоондоо key-forwarding буюу түлхүүр дамжуулалтыг болиулах эсвэл ssh-ийн authorized_keys файлдаа зөвхөн тусгайлсан машинуудаас нэвтрэхэд түлхүүрийг ашиглаж болохоор болгож зөвшөөрдөг `from=IP/DOMAIN` тохируулгыг ашиглахыг зөвлөдөг.

# 15.4. DES, Blowfish, MD5, SHA256, SHA512 болон Crypt

UNIX® систем дээрх хэрэглэгч бүрийн хувьд нууц үг бүртгэлтэй нь холбоотой байдаг. Мэдээж эдгээр нууц үгүүд нь зөвхөн хэрэглэгч ба үйлдлийн системд мэдэгдэж байх ёстой. Эдгээр нууц үгүүдийг нууцлаг байлгахын тулд тэдгээрийг "one-way hash буюу үл буцах хэш" гэгддэг шифрлэхэд амархан боловч буцааж болдоггүй аргаар шифрлэдэг. Өөрөөр хэлбэл хормын өмнө мэдээж гэж хэлсэн бидний хэлсэн үг яг жинхэнэдээ үнэн биш юм: үйлдлийн систем өөрөө нууц үгийг *жинхэнэдээ* мэддэггүй. Энэ нь зөвхөн нууц үгийн *шифрлэсэн* хэлбэрийг мэддэг. "plain-text буюу ердийн уншигдах текст " хэлбэрийн нууц үгийг авах цорын ганц арга нь боломжит нууц үгүүдийн орон зайгаас балмадаар хүчлэн хайх явдал юм.

Харамсалтай нь UNIX® бий болсон тэр үед нууц үгийг аюулгүй аргаар шифрлэх цорын ганц арга нь DES, Data Encryption Standard буюу Өгөгдөл Шифрлэх Стандарт дээр үндэслэсэн байлаа. Энэ нь АНУ-д оршин сууж байсан хэрэглэгчдийн хувьд тийм ч асуудалтай биш байсан юм, гэхдээ DES-ийн эх код АНУ-аас гадагшаа экспорт хийгдэж болохгүй байсан учир FreeBSD нь АНУ-ын хуулийг дагахын хажуугаар DES-ийг ашигласан хэвээр байсан бусад бүх UNIX® төрлүүдтэй нийцтэй байх арга замыг хайж олоход хүрсэн юм.

Үүний шийдэл нь АНУ-ын хэрэглэгчид DES сангуудыг суулгаж ашиглах боломжтой мөртлөө олон улсын хэрэглэгчид гадагш экспорт хийгдэж болох шифрлэх аргатай бас байхаар шифрийн сангуудыг хуваасан явдал байлаа. Ингэж FreeBSD нь MD5-ийг өөрийн анхдагч шифрлэх аргаа болгон ашиглах болсон юм. MD5 нь DES-ээс илүү аюулгүй нууцлаг гэгддэг бөгөөд DES-ийг суулгах нь үндсэндээ нийцтэй байх шалтгаануудын улмаас зориулагдсан юм.

## 15.4.1. Өөрийн Crypt арга замыг таних нь

Одоогоор шифрийн сан DES, MD5, Blowfish, SHA256 болон SHA512 хэш функцуудыг дэмждэг. Анхдагчаар FreeBSD нь MD5 ашиглан нууц үгүүдийг шифрлэдэг.

FreeBSD аль шифрлэх аргыг тохируулж ашиглаж байгааг мэдэх хялбар байдаг. /etc/master.passwd файл дахь шифрлэсэн нууц үгийг шалгах нь нэг арга юм. MD5 хэшээр шифрлэгдсэн нууц үгүүд нь DES-р шифрлэгдсэнийгээ бодох юм бол урт бөгөөд `$1$` тэмдэгтээр бас эхэлдэг. `$2a$` тэмдэгтээр эхэлсэн нууц үгүүд Blowfish хэш функцаар шифрлэгдсэн байдаг. DES мөр нь ямар нэг тусгайлан таньж болох шинж тэмдэггүй байдаг боловч тэд MD5 нууц үгүүдээс богино бөгөөд `$` тэмдэгт ордоггүй 64 тэмдэгттэй цагаан толгойгоор кодчилогддог, тиймээс долларын тэмдэгтээр эхлээгүй харьцангуй богино мөр ихэвчлэн DES нууц үг байдаг. SHA256 болон SHA512 нь `$6$` тэмдэгтээр эхэлдэг.

Шинэ нууц үгүүдэд ашиглагдах нууц үгийн хэлбэр нь нэвтрэлтийн `passwd_format` боломжийн тусламжтай /etc/login.conf файлд хянагддаг бөгөөд энэ хувьсагч нь `des`, `md5 blf`, `sha256` эсвэл `sha512` утгуудыг авдаг. Нэвтрэлтийн боломжуудын талаар дэлгэрэнгүй мэдээллийг login.conf(5) гарын авлагын хуудаснаас үзнэ үү.

# 15.5. Нэг удаагийн нууц үгүүд

Анхдагчаар FreeBSD OPIE (One-time Passwords In Everything буюу Бүхэнд зориулсан нэг

удаагийн нууц үгүүд) дэмжлэгтэй байдаг бөгөөд энэ нь MD5 хэшийг анхдагчаар ашигладаг.

Бид гурван өөр төрлийн нууц үгийг доор хэлэлцэх болно. Эхнийх нь таны ердийн UNIX® загварын эсвэл Kerberos нууц үг юм; бид үүнийг "UNIX® нууц үг" гэж нэрлэх болно. Хоёр дахь төрөл нь OPIE opiekey(1) програмаар үүсгэгдэж opiepasswd(1) програм болон нэвтрэлт хүлээх мөр хүлээн авах нэг удаагийн нууц үг юм; бид үүнийг "нэг удаагийн нууц үг" гэх болно. Сүүлийн төрөл нууц үг бол `opiekey` програмд (заримдаа `opiepasswd` програмууд) өгдөг нууцлаг нууц үг бөгөөд үүнийг ашиглан дээрх програмууд нэг удаагийн нууц үг үүсгэдэг; бид үүнийг "нууцлаг нууц үг" гэх буюу эсвэл зүгээр л шалгагдаагүй "нууц үг" гэх болно.

Нууцлаг нууц үг нь таны UNIX® нууц үгтэй ямар ч холбоогүй юм; тэдгээр нь адил байж болох боловч ингэхийг зөвлөдөггүй. OPIE нууцлаг нууц үгүүд нь хуучин UNIX® нууц үгүүд шиг 8 тэмдэгтэд хязгаарлагддаггүй бөгөөд таны хүссэн хэмжээний урттай байж болдог. Зургаа эсвэл долоон үг бүхий өгүүлбэрээс тогтох нууц үгүүд нэлээн элбэг байдаг. Ихэнх хэсгийн хувьд OPIE систем UNIX®-ийн нууц үгийн системээс бүр мөсөн ангид ажилладаг.

Нууц үгээс гадна OPIE-д чухал өгөгдлийн өөр хоёр хэсэг байдаг. Нэг нь "seed буюу үр" эсвэл "key буюу түлхүүр" гэгддэг бөгөөд 2 үсэг болон таван тооноос тогтдог. Нөгөөдөх нь "давталтын тоо" буюу 1-ээс 100 хүртэлх тоо юм. OPIE нэг удаагийн нууц үгийг үр болон нууцлаг нууц үгийг нийлүүлэн MD5 хэшийг давталтын тоогоор ашиглан үүсгэж үр дүнг нь зургаан богино Англи үг болгодог. Эдгээр зургаан Англи үг нь таны нэг удаагийн нууц үг юм. Нэвтрэлт шалгах систем (үндсэндээ PAM) ашигласан хамгийн сүүлийн нэг удаагийн нууц үгийг хадгалж байдаг бөгөөд хэрэглэгчийн өгсөн нууц үгийн хэш өмнөх нууц үгтэй таарч байвал хэрэглэгчийг нэвтрүүлдэг. Үл буцах хэш ашиглагддаг болохоор хэрэв амжилттайгаар ашиглагдсан нууц үгийг олж авсан бол дараа дараагийн нэг удаагийн нууц үгүүдийг үүсгэх боломжгүй байдаг; хэрэглэгч болон нэвтрэлтийн програмыг хамгийн сүүлийн хэлбэрт адилхан байлгаж байхын тулд давталтын тоо амжилттай нэвтрэлт хийгдэх бүрийн дараа багасаж байдаг. Давталтын тоо 1 хүрэх үед OPIE дахин хийгдэх хэрэгтэй болно.

Систем болгоны хувьд хэдэн програмууд байдаг бөгөөд тэдгээрийг бид энд хэлэлцэх болно. `opiekey` програм давталтын тоо, үр болон нууцлаг нууц үгийг хүлээн авч нэг удаагийн нууц үг эсвэл нэг удаагийн нууц үгүүдийн үргэлжилсэн жагсаалтыг үүсгэдэг. `opiepasswd` програмыг OPIE-г эхлүүлэх болон нууц үг, давталтын тоо эсвэл үр өөрчлөхөд ашигладаг; энэ нь нууцлаг нэвтрэх үгс аль эсвэл давталтын тоо, үр болон нэг удаагийн нууц үгийг авдаг. `opieinfo` програм тохирох итгэмжлэлүүдийн файлуудыг (/etc/opiekeys) шалгаж ажиллуулсан хэрэглэгчийн одоогийн давталтын тоо болон үрийг дэлгэцэд гаргадаг.

Бид дөрвөн өөр төрлийн үйлдлийн талаар хэлэлцэх болно. Эхнийх нь аюулгүй холболтоор `opiepasswd` ашиглаж нэг удаагийн нууц үгүүдийг эхний удаа тохируулах эсвэл өөрийн нууц үг эсвэл үрийг өөрчлөх үйлдэл юм. Хоёр дахь үйлдэл нь `opiepasswd`-г аюултай холболтоор, `opiekey` тушаалыг аюулгүй холболтоор ашиглаж адил үйлдлийг хийх явдал юм. Гурав дахь нь `opiekey`-г аюултай холболтоор ашиглан нэвтрэн орох үйлдэл юм. Дөрөв дэх нь `opiekey`-г ашиглан хэд хэдэн түлхүүрүүд үүсгэх үйлдэл бөгөөд гадагшаа аюулгүй холболтуудгүй газрууд уруу явахдаа тэдгээр түлхүүрүүдийг бичин авч эсвэл хэвлэн аваад өөртөө авч явж болох юм.

### 15.5.1. Аюулгүй холболт эхлүүлэх

OPIE-г эхний удаа эхлүүлэхдээ `opiepasswd` тушаалыг ажиллуул:

```
% opiepasswd -c
[grimreaper] ~ $ opiepasswd -f -c
Adding unfurl:
Only use this method from the console; NEVER from remote. If you are using
telnet, xterm, or a dial-in, type ^C now or exit with no password.
Then run opiepasswd without the -c parameter.
Using MD5 to compute responses.
Enter new secret pass phrase:
Again new secret pass phrase:

ID unfurl OTP key is 499 to4268
MOS MALL GOAT ARM AVID COED
```

`Enter new secret pass phrase:` эсвэл `Enter secret password:` мөрүүд дээр та нууц үг эсвэл өгүүлбэр оруулах ёстой. Энэ нь таны нэвтрэхдээ ашиглах нууц үг биш гэдгийг санах хэрэгтэй, үүнийг ашиглаж таны нэг удаагийн нэвтрэх түлхүүрийг үүсгэдэг. "ID" мөр таны тухайн үеийн параметрүүд болох таны нэвтрэх нэр, давталтын тоо болон үрийг өгдөг. Нэвтрэн орох үед систем эдгээр параметрүүдийг санаж танд тэдгээрийг санах шаардлагагүйгээр буцаан үзүүлдэг. Сүүлийн мөр нь тэдгээр параметрүүд болон таны нууцлаг нууц үгт харгалзах нэг удаагийн нууц үгийг өгдөг; хэрэв та нэн даруй дахин нэвтэрвэл энэ нэг удаагийн нууц үг нь таны ашиглах тэр нууц үг юм.

### 15.5.2. Аюултай холболт эхлүүлэх

Өөрийн нууцлаг нууц үгийг аюултай холболтоор эхэлж өгөхдөө эсвэл өөрчлөхдөө `opiekey` ажиллуулж болох тийм газар уруу аюулгүй холболттой байж байх шаардлагатай; энэ нь таны итгэж байгаа машин дээр бүрхүүлийн тушаал хүлээх мөр хэлбэрээр байж болно. Та бас давталтын тоог (100 боломжийн утга байж болох юм) бодож өгөх хэрэгтэй бөгөөд та өөрөө үр бодож олох эсвэл дурын үүсгэснийг ашиглах хэрэгтэй. Аюултай холболтоор (таны эхлүүлж байгаа машин уруу) `opiepasswd` тушаалыг ашигла:

```
% opiepasswd

Updating unfurl:
You need the response from an OTP generator.
Old secret pass phrase:
        otp-md5 498 to4268 ext
        Response: GAME GAG WELT OUT DOWN CHAT
New secret pass phrase:
        otp-md5 499 to4269
        Response: LINE PAP MILK NELL BUOY TROY

ID mark OTP key is 499 gr4269
LINE PAP MILK NELL BUOY TROY
```

Анхдагч үрийг хүлээж авах бол `Return` дар. Дараа нь хандах нууц үгийг оруулахын өмнө аюулгүй холболт уруугаа орж адил параметрүүдийг өгөөрэй:

```
% opiekey 498 to4268
Using the MD5 algorithm to compute response.
Reminder: Don't use opiekey from telnet or dial-in sessions.
Enter secret pass phrase:
GAME GAG WELT OUT DOWN CHAT
```

Одоо аюултай холболт уруугаа шилжиж үүсгэсэн нэг удаагийн нууц үгээ тохирох програм уруу хуулаарай.

### 15.5.3. Нэг удаагийн нууц үг ганцыг үүсгэх нь

OPIE-г эхлүүлэн тохируулж нэвтэрсний дараа танд иймэрхүү тушаал хүлээх мөр харуулагдана:

```
% telnet example.com
Trying 10.0.0.1...
Connected to example.com
Escape character is '^]'.

FreeBSD/i386 (example.com) (ttypa)

login: <username>
otp-md5 498 gr4269 ext
Password:
```

Энэ дашрамд тэмдэглэн хэлэхэд OPIE тушаал хүлээх мөрүүд ашигтай боломжтой байдаг: хэрэв та нууц үг хүлээх мөр дээр `Return` дарвал хүлээх мөр цуурайг идэвхжүүлж таны юу бичиж байгааг танд харуулдаг. Та хэвлэсэн зүйлээсээ харж магадгүй нууц үгийг гараараа бичиж оруулахыг оролдож байгаа бол энэ маш ашигтай байж болох юм.

Энэ үед нэвтрэлт хүлээх мөрөнд хариулахын тулд та өөрийн нэг удаагийн нууц үгийг үүсгэх хэрэгтэй болно. Үүнийг `opiekey` тушаал итгэн ажиллуулж чадах тийм систем дээрээ хийх хэрэгтэй. (DOS, Windows® болон Mac OS®-д зориулсан эдгээрийн хувилбарууд байдаг) Эдгээрт давталтын тоо болон үр тушаалын мөрийн тохируулга хэлбэрээр хэрэгтэй байдаг. Та нэвтрэн орж байгаа машиныхаа нэвтрэлт хүлээх мөрөөс эдгээрийг шууд хуулан тавьж болох юм.

Итгэсэн систем дээрээ:

```
% opiekey 498 to4268
Using the MD5 algorithm to compute response.
Reminder: Don't use opiekey from telnet or dial-in sessions.
Enter secret pass phrase:
GAME GAG WELT OUT DOWN CHAT
```

Одоо та өөрийн нэг удаагийн нууц үгтэй болсон болохоор нэвтрэлтээ үргэлжлүүлж болно.

### 15.5.4. Нэг удаагийн нууц үг олныг үүсгэх нь

Заримдаа та итгэсэн машин эсвэл аюулгүй холболт уруу хандах боломжгүй тийм газар очих хэрэгтэй болдог. Энэ тохиолдолд `opiekey` тушаал ашиглаж хэд хэдэн нэг удаагийн нууц үгүүдийг урьдчилан үүсгэж хэвлэн биедээ авч явах боломжтой юм. Жишээ нь:

```
% opiekey -n 5 30 zz99999
Using the MD5 algorithm to compute response.
Reminder: Don't use opiekey from telnet or dial-in sessions.
Enter secret pass phrase: <secret password>
26: JOAN BORE FOSS DES NAY QUIT
27: LATE BIAS SLAY FOLK MUCH TRIG
28: SALT TIN ANTI LOON NEAL USE
29: RIO ODIN GO BYE FURY TIC
30: GREW JIVE SAN GIRD BOIL PHI
```

`-n 5` нь дараалсан таван түлхүүрийг үүсгэхийг, `30` нь сүүлийн давталтын тоог хэд байх ёстойг зааж өгч байгаа юм. Эдгээр нь ашиглах бололцоотойг *урвуу* дарааллаар дэлгэцэнд харуулдгийг тэмдэглэх нь зүйтэй. Хэрэв та хэт санаа зовниж байгаа бол та үр дүнг гараар бичиж авахыг хүсэж болох юм; эсвэл `lpr` уруу хуулан авч тавьж болох юм. Мөр бүр давталтын тоо болон нэг удаагийн нууц үгийг харуулж байгааг анхаараарай; та нууц үгүүдийг хэрэглэх бүртээ тэдгээрийг арилгаж энэ хэвлэсэн арга тань ашигтай хэвээр болохыг мэдэж болох юм.

### 15.5.5. UNIX® нууц үгүүдийг ашиглахыг хязгаарлах нь

OPIE нь UNIX® нууц үгүүдийн ашиглалтыг нэвтрэлтийн сессийн IP хаяг дээр тулгуурлан хязгаарлаж чаддаг. Тохирох файл нь /etc/opieaccess бөгөөд энэ файл нь анхдагчаар байдаг. Энэ файлын талаар болон үүнийг ашигласнаар та аюулгүй байдлын ямар зүйлсүүдийг бодолцож анхаарах ёстой талаар дэлгэрэнгүй мэдээллийг opieaccess(5)-с шалгана уу.

Энд жишээ opieaccess файл байна:

```
permit 192.168.0.0 255.255.0.0
```

Энэ мөр нь UNIX® нууц үгүүдийг ямар ч үед ашиглахын тулд эхлэл IP хаягийг (хууран мэхлэхэд хүрч болох тийм эмзэг) заагдсан утга болон багтай тааруулах боломжийг хэрэглэгчдэд олгодог.

opieaccess дахь аль ч дүрэм таарахгүй байгаа бол анхдагчаар OPIE биш нэвтрэлтүүдийг хааж үгүйсгэдэг.

## 15.6. TCP Гүйцэтгэлийг хялбаршуулагчид

inetd(8)-г мэддэг хэн бүхэн TCP Гүйцэтгэлийг хялбаршуулагчдын талаар заримдаа сонссон байх. Гэхдээ цөөн хүмүүс энэ боломжийн сүлжээний орчин дахь ашигтай талыг бүрэн ойлгодог юм шиг санагддаг. Хүн бүхэн сүлжээний холболтууд зохицуулах галт хана

суулгахыг хүсдэг юм шиг санагддаг. Галт хана олон төрлийн хэрэглээтэй боловч холболт үүсгэгч уруу текст илгээх зэрэг зарим зүйлсийг галт хана хийж чаддаггүй. Энд дурдсан TCP Гүйцэтгэлийг хялбаршуулагчид энэ мэтийг болон үүнээс илүүг хийдэг. Дараагийн хэдэн хэсэгт TCP Гүйцэтгэлийг хялбаршуулагчдын олон боломжуудыг хэлэлцэх бөгөөд боломжтой үед нь жишээ тохиргооны мөрийг үзүүлэх болно.

TCP Гүйцэтгэлийг хялбаршуулагчид програм хангамж нь inetd-ийн чадваруудыг сервер бүрийн хувьд түүний доор хянагдаж болохоор дэмжин өргөтгөдөг. Энэ аргыг ашиглан бүртгэл хөтлөх дэмжлэг нэмэх, холболтууд уруу мэдэгдэл буцаах, дэмонд зөвхөн дотоод холболтуудыг хүлээн авахыг зөвшөөрөх гэх мэт үйлдлүүдийг хийх боломжтой. Эдгээр боломжуудын заримыг галт хана суулган тохируулж хийж болох боловч энэ нь зөвхөн хамгаалалтын нэмэлт давхарга болохоос гадна галт ханын үзүүлж чаддагаас илүү хяналтыг олгодог юм.

TCP Гүйцэтгэлийг хялбаршуулагчдын ийнхүү нэмэгдсэн ажиллагаа нь сайн галт ханыг солихоор зүйл гэж ойлгогдох ёсгүй юм. TCP Гүйцэтгэлийг хялбаршуулагчид нь галт хана эсвэл өөр бусад аюулгүй байдлыг нэмэгдүүлэгч програмуудын хамтаар ашиглагдаж системийн хувьд хамгаалалтын нэмэлт давхарга болон аятайхан үйлчлэх боломжтой юм.

Энэ нь inetd-ийн тохиргооны өргөтгөл болохоор энэхүү баримтыг уншигч таныг inetd тохиргоо хэсгийг уншсан гэдэгт найдаж байна.

inetd(8)-ээр ажиллуулагдсан програмууд яг жинхэнээрээ "дэмонууд" биш боловч тэдгээрийг уламжлалаар дэмонууд гэдэг. Энэ ухагдахууныг бид энэ хэсэгт бас ашиглах болно.

### 15.6.1. Эхний тохиргоо

TCP Гүйцэтгэлийг хялбаршуулагчдыг FreeBSD-д ашиглахад байх цорын ганц шаардлага нь inetd серверийг rc.conf файлаас `-Ww` тохируулгатай ажиллуулсан эсэхийг шалгах явдал юм; энэ нь анхдагч тохиргоо юм. Мэдээж /etc/hosts.allow файлын зөв тохиргоо бас байгааг хүлээж байдаг боловч эдгээр тохиолдлуудад syslogd(8) системийн бүртгэлүүдэд мэдэгдлүүд шиддэг.

Бусад TCP Гүйцэтгэлийг хялбаршуулагчдын шийдлүүдтэй харьцуулах юм бол hosts.deny файлыг хэрэглэхээ больсон. Тохиргооны бүх сонголтууд /etc/hosts.allow файлд байх шаардлагатай.

Хамгийн амархан тохиргоогоороо бол дэмоны холболтын бодлогууд зөвшөөрөгдсөн эсвэл хаагдсаны аль нэгээр /etc/hosts.allow файл дахь тохируулгуудаас хамааран тохируулагддаг. FreeBSD дээрх анхдагч тохиргоо нь inetd-ээр эхэлсэн дэмон бүр уруу хийгдэх холболтыг зөвшөөрдөг. Үүнийг өөрчлөх талаар зөвхөн үндсэн тохиргооны тухай дурдсаны дараа хэлэлцэх болно.

Үндсэн тохиргоо ихэвчлэн `дэмон : хаяг : үйлдэл` хэлбэрийг авдаг. Энд байгаа `дэмон` нь `inetd` -ийн эхлүүлсэн дэмоны нэр юм. `Хаяг` нь зөв хостын нэр, `address` хаяг эсвэл дөрвөлжин хаалтан ([ ]) доторх IPv6 хаяг байж болно. `action` буюу үйлдлийн талбар нь `allow` буюу зөвшөөрөх эсвэл `deny` буюу эрхийг хориглох эсвэл хандалтыг хаахын аль нэг байна. Тохиргоо эхний тохирсон дүрэм журмын дагуу ажилладаг гэдгийг санах хэрэгтэй, энэ нь тохирох

дүрмийг тохиргооны файлаас өсөх дарааллаар хайна гэсэн үг юм. Тохирох дүрэм олдвол тэр дүрэм ашиглагдаж хайх процесс зогсоно.

Бусад хэд хэдэн тохируулгууд байдаг боловч тэдгээрийг энэ хэсгийн сүүлд тайлбарлах болно. Хялбар тохиргооны мөр ганцхан тэр мэдээллийн дагуу амархнаар хийгдэж болно. Жишээ нь mail/qpopper дэмоноор дамжин хийгдэж болох POP3 холболтуудыг зөвшөөрөхийн тулд дараах мөрүүд hosts.allow файлд нэмж хийгдэх хэрэгтэй:

```
# This line is required for POP3 connections:
qpopper : ALL : allow
```

Энэ мөрийг нэмснийхээ дараа inetd-г service(8) ашиглан дахин эхлүүлэх хэрэгтэй:

```
# service inetd restart
```

## 15.6.2. Дэвшилтэт тохиргоо

TCP Гүйцэтгэлийг хялбаршуулагчид нь бас дэвшилтэт тохируулгуудтай байдаг; тэдгээр нь холболтуудтай хэрхэн ажиллахыг илүүтэйгээр хянах боломжийг олгодог. Зарим тохиолдолд тодорхой хостууд эсвэл дэмон холболтууд уруу тайлбар буцаах нь зүйтэй санаа байж болох юм. Бусад тохиолдолд магадгүй бүртгэлийн файл бичигдэх ёстой эсвэл цахим захидал администратор уруу илгээгдэж болох юм. Бусад тохиолдлууд үйлчилгээг зөвхөн дотоод холболтууддаа ашиглахыг шаардаж болох юм. Эдгээр нь бүгдээрээ `орлуулагддаг тэмдэгтүүд`, өргөтгөх тэмдэгтүүд болон гадаад тушаалыг ажиллуулах зэрэг тохиргооны сонголтуудын тусламжтай хийгдэх боломжтой юм. Дараагийн хоёр хэсэгт эдгээр тохиолдлуудын талаар бичсэн байгаа.

### 15.6.2.1. Гадаад тушаалууд

Холболтыг хааж түүнийг тогтоохыг оролдсон хүн уруу шалтгааныг нь илгээх тохиолдол гарчээ гэж бодъё. Үүнийг яаж хийх вэ? Энэ үйлдлийг `twist` тохируулга ашиглан хийх боломжтой. Холболт тогтоохоор оролдоход `twist` тохируулга бүрхүүлийн тушаал эсвэл скрипт ажилуулахаар дуудагддаг. hosts.allow файлд үүний жишээ аль хэдийн орсон байдаг:

```
# The rest of the daemons are protected.
ALL : ALL \
        : severity auth.info \
        : twist /bin/echo "You are not welcome to use %d from %h."
```

Энэ жишээ нь "You are not allowed to use `daemon` from `hostname`. "буюу" Та `дэмон`ыг `hostname`-с ашиглах зөвшөөрөлгүй." гэсэн мэдэгдлийг хандалтын файлд урьдаар тохируулагдаагүй дэмон бүрийн хувьд буцаадаг. Энэ нь тогтоогдсон холболт дөнгөж салсны дараа холболтыг эхлүүлэгч уруу хариултыг буцааж илгээхэд маш их ашигтай байдаг. Буцсан мэдэгдэл бүр `"` тэмдэгтүүд дотор заавал байх *шаардлагатай*; энэ дүрмэнд ямар нэг жич зөвшөөрөл байхгүй.

Хэрэв халдагч эсвэл бүлэг халдагчид эдгээр дэмонуудыг холболт хийх

> хүсэлтээр цутгаж чадах юм бол серверийн эсрэг үйлчилгээг зогсоох халдлага явуулах боломжтой байж болох юм.

Өөр нэг боломж нь эдгээр тохиолдлуудад `spawn` тохируулгыг ашиглах явдал юм. `twist` тохируулгын нэгэн адил `spawn` тохируулга нь холболтуудыг сохроор хааж гадаад бүрхүүлийн тушаалууд эсвэл скриптүүдийг ажиллуулахад ашиглагдаж болно. `twist` тохируулгаас ялгаатай тал нь `spawn` нь холболт тогтоосон хүн уруу хариулт буцааж илгээдэггүй. Жишээ нь дараах тохиргооны мөр байжээ гэж бодъё:

```
# We do not allow connections from example.com:
ALL : .example.com \
    : spawn (/bin/echo %a from %h attempted to access %d >> \
      /var/log/connections.log) \
    : deny
```

Энэ нь `*.example.com` домэйноос ирсэн бүх холболтын оролдлогуудаас татгалзахын зэрэгцээ хостын нэр, IP хаяг болон тэдний хандалт хийхийг оролдсон дэмонг /var/log/connections.log файл уруу бүртгэнэ.

Дээр тайлбарласан орлуулах тэмдэгтүүдээс гадна, өөрөөр хэлбэл `%a` тэмдэгтээс гадна бусад цөөн хэдэн тэмдэгтүүд бас байдаг. Бүрэн жагсаалтыг hosts_access(5) гарын авлагын хуудаснаас үзнэ үү.

#### 15.6.2.2. Орлуулагддаг тэмдэгтүүдийн тохиргоонууд

Энэ хүртэл `ALL` тохируулга бүх л жишээнүүдэд ашиглагдлаа. Ажиллагааг арай цаашлуулж өргөтгөх бусад тохируулгууд байдаг. Жишээ нь `ALL` нь дэмон, домэйн эсвэл IP хаягийн аль нэгтэй тааруулах зорилгоор ашиглагдаж болох юм. Өөр нэг орлуулагддаг тэмдэгт нь IP хаягаа өөрчлөн хуурсан байж болох дурын хостыг тааруулах `PARANOID` тохируулга юм. Өөрөөр хэлбэл `PARANOID` буюу хэт зовнил нь өөрийн хостын нэрээс өөр IP хаягтай машинаас холболт хийгдэх бүр түүнд тохирох үйлдлийг тодорхойлоход ашиглагдаж болох юм. Дараах жишээ энэ хэлэлцүүлэгт арай илүү ойлголт өгч магадгүй юм:

```
# Block possibly spoofed requests to sendmail:
sendmail : PARANOID : deny
```

Энэ жишээн дээр `sendmail` уруу хийгдэж байгаа өөрийнхөө хостын нэрээс өөр IP хаягтай холболтын бүх хүсэлтүүдээс татгалзан хааж байна.

> Хэрэв клиент эсвэл сервер эвдэрхий DNS суулгацтай бол `PARANOID` орлуулагддаг тэмдэгтийг ашиглах нь серверүүдийг ноцтойгоор зэрэмдэг болгож болох юм. Иймд администраторын зохион байгуулалт болон хуваарилалт хийхийг зөвлөж байна.

Орлуулагддаг тэмдэгтүүдийн талаар болон тэдэнтэй холбоотой ажиллагааны талаар дэлгэрэнгүйг hosts_access(5) гарын авлагын хуудаснаас үзээрэй.

Тусгай тохиргооны аль ч мөрүүдийн өмнө дээрх нь ажиллана, эхний тохиргооны мөр hosts.allow файлд тайлбар болгон хаагдах шаардлагатай. Үүнийг энэ хэсгийн эхэнд тэмдэглэж хэлсэн байгаа.

## 15.7. Kerberos5

Kerberos нь хэрэглэгчид өөрсдийгөө нууцлаг серверийн үйлчилгээнүүдийн тусламжтайгаар таниулан нэвтрэх боломжийг олгодог сүлжээний нэмэлт систем/протокол юм. Алсын нэвтрэлт, алсын хуулбар, нууцлаг систем хоорондох файл хуулбарлалт болон бусад аюул ихтэй үйлдлүүд зэрэг үйлчилгээнүүд харьцангуй аюулгүй хийгдэж илүү хяналт хийж болохоор болсон.

Kerberos нь хэн бэ гэдгийг шалгах прокси систем юм. Энэ нь бас итгэгдсэн гуравдагч нэвтрэлт таних систем гэж тайлбарлагдаж болно. Kerberos нь зөвхөн нэг функцыг хангадаг - сүлжээн дээр хэрэглэгчдэд өөрсдийгөө аюулгүйгээр таниулах боломжийг хангаж өгдөг. Энэ нь шалгаж таних функцууд (хэрэглэгчдийн хийхийг зөвшөөрдөг) эсвэл аудит функцуудын (тэдгээр хэрэглэгчид юу хийснийг) үүргийг гүйцэтгэдэггүй. Клиент болон сервер өөрийгөө таниулж батлахаар Kerberos-г ашигласны дараа тэд бизнесээ бодож өөрсдийн бүх холболтуудаа шифрлэж нууцлал болон бүрэн бүтэн байдлаа хадгалан баталгаажуулж болно.

Иймээс Kerberos-ийг нэвтрэлт танилт болон аудит үйлчилгээнүүдийг хангадаг бусад аюулгүй байдлын аргуудтай цуг ашиглахыг маш ихээр зөвлөдөг.

Дараах зааврууд FreeBSD-д зориулан түгээгдсэн Kerberos-ийг хэрхэн тохируулах гарын авлага болгон ашиглаж болно. Гэхдээ та тохирох гарын авлагын хуудаснуудаас бүрэн тайлбарын талаар лавлах хэрэгтэй.

Kerberos-ийн суулгацыг үзүүлэх зорилгоор төрөл бүрийн нэрийн талбарууд дараах байдлаар зохицуулагдана:

- DNS домэйн ("бүс") нь example.org байна.
- Kerberos хүрээ нь EXAMPLE.ORG байна.

Хэрэв та дотооддоо ажиллуулах бодолтой байсан ч гэсэн Kerberos-ийг суулгаж тохируулахдаа жинхэнэ домэйны нэрүүдийг ашиглана уу. Энэ нь DNS-ийн асуудлуудыг тойрон гарч бусад Kerberos хүрээнүүдтэй хийх хоорондын үйлдлийг баталгаажуулдаг.

### 15.7.1. Түүх

Kerberos-ийг MIT анх сүлжээний аюулгүй байдлын асуудлуудын шийдэл болгож хийсэн. Kerberos протокол нь хүчирхэг криптографыг ашигладаг бөгөөд клиент нь аюултай сүлжээний холболтоор өөрийгөө хэн бэ гэдгийг серверт (болон эсрэгээр) баталж чадах боломжийг олгодог.

Kerberos нь сүлжээний танин шалгах протоколын нэрээс гадна програмыг (жишээ нь Kerberos телнет) шийдвэрлэж байгаа програмуудыг тайлбарласан тайлбар бас болдог. Протоколын одоогийн хувилбар нь 5 бөгөөд RFC 1510-д тайлбарласан байдаг.

Өргөн хүрээний үйлдлийн системүүдийг хамарсан энэ протоколын хэд хэдэн чөлөөтэй шийдлүүд байдаг. Kerberos анх хөгжүүлэгдсэн Массачусетсийн Технологийн Институт (MIT) нь өөрийн Kerberos багцыг хөгжүүлсээр байна. Энэ багц нь US-д криптограф бүтээгдэхүүн болж нийтлэг хэрэглэгддэг бөгөөд энэ нь түүхээс авч үзэхэд US-ын экспортын дүрэм журмуудаас болсон юм. MITKerberos нь порт (security/krb5) хэлбэрээр байдаг. Heimdal Kerberos нь өөр шийдлийн 5-р хувилбар бөгөөд экспортын дүрэм журмуудыг тойрон гарах зорилгоор US-ээс гадна хамааралгүйгээр хөгжүүлэгдсэн ( бөгөөд ихэвчлэн арилжааны бус UNIX® төрлүүдэд орсон байдаг) юм. Heimdal Kerberos түгээлт нь порт (security/heimdal) хэлбэрээр байдаг бөгөөд үүний хамгийн бага суулгац үндсэн FreeBSD суулгацад орсон байдаг.

Аль болох олон үзэгчдийг хамрахын тулд эдгээр зааврууд нь FreeBSD-д орсон Heimdal түгээлтийг ашиглаж байна гэж тооцдог.

### 15.7.2. Heimdal KDC суулгаж тохируулах

Түлхүүр Түгээх Төв (KDC) нь Kerberos-ийн хангадаг төвлөрсөн нэвтрэлт таних үйлчилгээ юм - энэ нь Kerberos тасалбарууд өгдөг компьютер юм. KDC нь Kerberos хүрээний бусад бүх компьютеруудад "итгэгдсэн" гэж тооцогддог бөгөөд аюулгүй байдлын санаа зовнилыг дээшлүүлдэг.

Kerberos серверийг ажиллуулж байхад маш цөөн тооцооллын эх үүсвэрийг шаарддаг боловч аюулгүй байдлын шалтгаанаас болоод зөвхөн KDC болон ажиллах тусдаа зориулагдсан машинтай байхыг зөвлөдгийг санаарай.

KDC-г тохируулж эхлэхдээ таны /etc/rc.conf файлд KDC болж ажиллах зөв тохиргоо хийгдсэн эсэхийг шалгаарай (өөрийн системийн хувьд та замуудыг өөрчлөх хэрэгтэй байж болох юм):

```
kerberos5_server_enable="YES"
kadmind5_server_enable="YES"
```

Дараа нь бид таны Kerberos тохиргооны файл /etc/krb5.conf-г тохируулна:

```
[libdefaults]
    default_realm = EXAMPLE.ORG
[realms]
    EXAMPLE.ORG = {
        kdc = kerberos.example.org
        admin_server = kerberos.example.org
    }
[domain_realm]
    .example.org = EXAMPLE.ORG
```

Энэ /etc/krb5.conf файл нь таны KDC нь бүрэн баталгаажсан хостын нэр `kerberos.example.org` -тэй байна гэж үзэж байгааг санаарай. Хэрэв таны KDC өөр хостын нэртэй бол та өөрийн бүсийн файлдаа CNAME (alias)-ийг нэмэх хэрэгтэй.

Зөв тохируулсан BINDDNS сервер бүхий том сүлжээнүүдэд өмнөх жишээ нь:

```
[libdefaults]
      default_realm = EXAMPLE.ORG
```

болж дараах мөрүүдийг `example.org` бүсийн файлд нэмж цэгцэлж болно:

```
_kerberos._udp      IN  SRV     01 00 88 kerberos.example.org.
_kerberos._tcp      IN  SRV     01 00 88 kerberos.example.org.
_kpasswd._udp       IN  SRV     01 00 464 kerberos.example.org.
_kerberos-adm._tcp  IN  SRV     01 00 749 kerberos.example.org.
_kerberos           IN  TXT     EXAMPLE.ORG
```

Kerberos үйлчилгээнүүдийг хэрэглэгчдэд хүртээмжтэй болгохын тулд та эсвэл бүрэн тохируулсан /etc/krb5.conf файлтай эсвэл хамгийн багаар тохируулсан /etc/krb5.conf файл *болон* зөв тохируулсан DNS сервертэй байх *ёстой*.

Дараа нь бид Kerberos мэдээллийн бааз үүсгэнэ. Энэ мэдээллийн бааз нь мастер нууц үгээр шифрлэсэн бүх удирдагчдын түлхүүрүүдийг агуулдаг. Та энэ нууц үгийг тогтоох шаардлагагүй, энэ нь файлд (/var/heimdal/m-key) хадгалагдах болно. Мастер түлхүүр үүсгэхийн тулд `kstash` тушаалыг ажиллуулж нууц үгээ оруулаарай.

Мастер түлхүүр үүсгэгдсэний дараа та мэдээллийн баазыг `kadmin` програмыг `-l` тохируулгатай ("локал" гэсэн утгатай) ашиглан эхлүүлж болно. Энэ тохируулга нь `kadmin`-д мэдээллийн баазын файлыг `kadmind` сүлжээний үйлчилгээгээр дамжилгүйгээр шууд өөрчлөхийг заадаг. Энэ нь мэдээллийн бааз үүсэхээс өмнө түүн уруу хандахыг оролдох асуудлыг (яг л өндөг, тахианы аль нь түрүүлж гарсан гэж маргадаг тэр асуудлын адил) зохицуулдаг. `kadmin` хүлээх мөртэй болсныхоо дараа та өөрийн хүрээнүүдийн эхний мэдээллийн санг `init` тушаал ашиглан үүсгээрэй.

Эцэст нь `kadmin`-ы горимд байхдаа өөрийн эхний удирдагчийг `add` тушаал ашиглан үүсгээрэй. Одоохондоо удирдагчийн хувьд анхдагч тохируулгуудыг сонгоорой, та тэдгээрийг сүүлд нь `modify` тушаал ашиглан өөрчилж чадна. Та аль ч тушаал хүлээх мөрөнд `?` тушаал ашиглаж байгаа боломжит тохируулгуудыг харж болохыг санаарай.

Мэдээллийн сан үүсгэлтийн жишээ сесс доор байна:

```
# kstash
Master key: xxxxxxxx
Verifying password - Master key: xxxxxxxx

# kadmin -l
kadmin> init EXAMPLE.ORG
Realm max ticket life [unlimited]:
kadmin> add tillman
```

```
Max ticket life [unlimited]:
Max renewable life [unlimited]:
Attributes []:
Password: xxxxxxxx
Verifying password - Password: xxxxxxxx
```

Одоо KDC үйлчилгээнүүдийг эхлүүлэх цаг болжээ. Үйлчилгээнүүдийг эхлүүлэхдээ `service kerberos start` болон `service kadmind start` тушаалуудыг ажиллуулна. Энэ үед танд ямар ч kerberos хийгдсэн дэмон байхгүйг санаарай, гэхдээ та KDC-ийн өөрийнх нь тушаалын мөрөөс үүсгэсэн удирдагчид (хэрэглэгч) зориулсан тасалбарыг авч жагсаан KDC-г ажиллаж байгаа гэдгийг та баталж чадаж байх ёстой:

```
% kinit tillman
tillman@EXAMPLE.ORG's Password:

% klist
Credentials cache: FILE:/tmp/krb5cc_500
    Principal: tillman@EXAMPLE.ORG

  Issued           Expires          Principal
Aug 27 15:37:58  Aug 28 01:37:58  krbtgt/EXAMPLE.ORG@EXAMPLE.ORG
```

Та дууссаныхаа дараа тасалбарыг буцааж болно:

```
% kdestroy
```

### 15.7.3. Серверийг Kerberos хийн Heimdal үйлчилгээнүүдтэй идэвхжүүлэх

Эхлээд бидэнд Kerberos-ийн тохиргооны файл /etc/krb5.conf-ийн хуулбар хэрэг болно. Ингэхийн тулд KDC-ээс түүнийг аюулгүй аргаар (scp(1) зэрэг сүлжээний хэрэгслүүд эсвэл физикээр уян диск ашиглан) клиент компьютер уруу ердөө л хуулах хэрэгтэй.

Дараа нь танд /etc/krb5.keytab файл хэрэгтэй. Энэ нь Kerberos хийгдсэн дэмонууд бүхий сервер болон ажлын станц хоёрын гол ялгаа юм - сервер нь keytab файлтай байх шаардлагатай. Энэ файл нь өөрийг нь зөвшөөрдөг серверийн хост түлхүүр болон өөрсдийнхөө нэрийг (identity) шалгах KDC-г агуулдаг. Хэрэв түлхүүр нь нийтэд мэдэгдвэл серверийн аюулгүй байдал эвдэрч болох учир энэ нь сервер уруу аюулгүйн үүднээс дамжуулагдах ёстой. Энэ нь шууд утгаараа FTP зэрэг цэвэр текст сувгаар дамжуулах нь маш буруу гэсэн үг юм.

Ихэвчлэн сервер уруу keytab файлыг `kadmin` тушаал ашиглан дамжуулдаг. Энэ нь тохиромжтой байдаг бөгөөд учир нь та бас хостын удирдагчийг (krb5.keytab файлын KDC төгсгөл) `kadmin` тушаал ашиглан үүсгэх хэрэгтэй болдог.

Та тасалбарыг аль хэдийн авсан байх ёстой бөгөөд энэ тасалбар нь kadmind.acl файлын `kadmin` интерфэйсийг ашиглаж болохоор зөвшөөрөгдсөн байх ёстойг санаарай. Heimdal-ийн

мэдээллийн хуудаснуудын (`info heimdal`) "Алсын удирдлага" гэсэн гарчигтай хэсгээс хандалт хянах жагсаалтуудыг дизайн хийх талаар дэлгэрэнгүйг үзнэ үү. Хэрэв та алсын `kadmin` хандалтыг идэвхжүүлэхийг хүсэхгүй байгаа бол та KDC уруу ердөө л аюулгүйгээр холбогдож (локал консолоор, ssh(1) эсвэл Kerberos telnet(1)) удирдлагыг локалаар өөр дээрээсээ `kadmin -l` тушаал ашиглан хийж болно.

/etc/krb5.conf файлыг суулгасны дараа та Kerberos серверээс `kadmin` тушаалыг ашиглаж болно. `add --random-key` тушаал нь серверийн хост удирдагчийг нэмэх боломжийг танд олгох бөгөөд `ext` тушаал нь серверийн хост удирдагчийг өөрийн keytab уруу задлах боломжийг танд олгоно. Жишээ нь:

```
# kadmin
kadmin> add --random-key host/myserver.example.org
Max ticket life [unlimited]:
Max renewable life [unlimited]:
Attributes []:
kadmin> ext host/myserver.example.org
kadmin> exit
```

`ext` тушаал нь ("extract" гэдгийг богиноор илэрхийлнэ) задалсан түлхүүрийг анхдагчаар /etc/krb5.keytab файлд хадгалдаг.

Хэрэв таны хувьд KDC дээр `kadmind` ажиллахгүй байгаа бөгөөд (магадгүй аюулгүй байдлын шалтгаануудаас болоод) тэгээд `kadmin` уруу алсаас хандах боломжгүй бол та хост удирдагчийг (`host/myserver.EXAMPLE.ORG`) шууд KDC дээр нэмж дараа нь доор дурдсантай адилаар түүнийг түр зуурын файл уруу (KDC дээрх /etc/krb5.keytab файлыг дарж бичихээс сэргийлж) задалж болно:

```
# kadmin
kadmin> ext --keytab=/tmp/example.keytab host/myserver.example.org
kadmin> exit
```

Та дараа нь keytab-ийг аюулгүйгээр (жишээ нь `scp` эсвэл уян диск ашиглан) сервер компьютер уруу хуулж болно. KDC дээрх keytab-ийг дарж бичихээс сэргийлж keytab нэрийг анхдагч бишээр зааж өгсөн эсэхээ шалгаарай.

Энэ мөчид хүрэх үед таны сервер KDC-тэй (krb5.conf файлтай учраас) холбогдож чадах бөгөөд (krb5.keytab файлтай учраас) өөрийгөө таниулан баталж чадна. Одоо та зарим нэг Kerberos үйлчилгээнүүдийг идэвхжүүлэхэд бэлэн болжээ. Энэ жишээн дээр бид `telnet` үйлчилгээг /etc/inetd.conf файлд доор дурдсантай төстэй мөрийг оруулан идэвхжүүлж дараа нь inetd(8) үйлчилгээг `service inetd restart` тушаалын тусламжтай дахин ачаалах болно:

```
telnet    stream  tcp     nowait  root    /usr/libexec/telnetd  telnetd -a user
```

Хамгийн чухал нь `-a` төрөл (нэвтрэлт танихад) хэрэглэгчид тохируулагдсан. Илүү дэлгэрэнгүйг telnetd(8) гарын авлагын хуудаснаас лавлана уу.

### 15.7.4. Клиентийг Kerberos хийн Heimdal үйлчилгээтэйгээр идэвхжүүлэх

Клиент компьютерийг тохируулах нь маш амархан. Kerberos тохиргоо хийгдсэний дараа танд зөвхөн /etc/krb5.conf-д байрлах Kerberos тохиргооны файл хэрэгтэй. Үүнийг ердөө л аюулгүйгээр клиент компьютер уруу KDC-ээс хуулна.

Клиентээсээ `kinit`, `klist`, болон `kdestroy` тушаалуудыг үүсгэсэн удирдагчийнхаа хувьд тасалбар олж авах, үзүүлэх, болон дараа нь устгахад ашиглахыг оролдон клиент компьютераа тест хийгээрэй. Та Kerberos програмуудыг ашиглан Kerberos хийгдсэн серверүүд уруу холбогдож чадах ёстой бөгөөд хэрэв ингэж ажиллаж болохгүй байгаа бөгөөд тасалбар олж авах нь асуудалтай байгаа бол энэ нь клиент эсвэл KDC-тэй холбоотой биш сервертэй холбоотой асуудал юм.

`telnet` зэрэг програмыг тест хийж байх үед таны нууц үг цэвэр текстээр бишээр илгээгдэж байгааг шалгахын тулд пакет шиншлэгч (tcpdump(1) зэрэг) ашиглаад үзээрэй. `telnet`-ийг бүх өгөгдлийн урсгалыг шифрлэдэг (`ssh`-тэй адил) `-x` тохируулгатай ашиглахыг оролдоорой.

Төрөл бүрийн гол биш Kerberos клиент програмууд нь бас анхдагчаар суудаг. Энэ нь үндсэн Heimdal суулгацын "хамгийн бага" мөн чанар юм: `telnet` нь цорын ганц Kerberos хийгдсэн үйлчилгээ юм.

Heimdal порт нь зарим нэг дутуу програмуудыг нэмдэг: `ftp`, `rsh`, `rcp`, `rlogin` болон бусад цөөн хэдэн нийтлэг биш програмуудын Kerberos хийгдсэн хувилбаруудыг нэмдэг. MIT порт нь бас Kerberos клиент програмуудын бүрэн цуглуулгыг агуулдаг.

### 15.7.5. Хэрэглэгчийн тохиргооны файлууд: .k5login болон .k5users

Хүрээн дэх хэрэглэгчийн хувьд ихэнхдээ өөрсдийнх нь Kerberos удирдагчийг (`tillman@EXAMPLE.ORG` зэрэг) локал хэрэглэгчийн бүртгэлд (`tillman` зэрэг локал бүртгэл) харгалзуулж өгсөн байдаг. `telnet` зэрэг клиент програмууд ихэвчлэн хэрэглэгчийн нэр эсвэл удирдагчийг шаарддаггүй.

Гэхдээ хаяа нэг та харгалзах Kerberos удирдагчгүй хэн нэгэнд зориулж локал хэрэглэгчийн бүртгэлд хандах хандалтыг өгөхийг хүсэж болох юм. Жишээ нь `tillman@EXAMPLE.ORG` магадгүй локал хэрэглэгчийн бүртгэл `webdevelopers`-д хандах хандалт хэрэгтэй байж болох юм. Бусад удирдагчид бас энэ локал бүртгэлд хандах хэрэгтэй байж болох юм.

.k5login болон .k5users файлууд нь хэрэглэгчдийн гэрийн сангуудад байрладаг бөгөөд .hosts болон .rhosts файлуудын хүчирхэг хослолын нэгэн адилаар энэ асуудлыг шийдэн ашиглагдаж болох юм. Жишээ нь хэрэв .k5login нь дараах агуулгатайгаар:

```
tillman@example.org
jdoe@example.org
```

локал хэрэглэгч `webdevelopers`-ийн гэр санд байрлаж байвал энд жагсаагдсан хоёр удирдагч хоёулаа хуваалцсан нууц үгийн шаардлагагүйгээр тэр бүртгэл уруу хандах хандалттай болох юм.

Эдгээр тушаалуудын гарын авлагын хуудаснуудыг уншихыг зөвлөж байна. `ksu` гарын авлагын хуудас .k5users файлын тухай тайлбарладгийг тэмдэглэх нь зүйтэй юм.

### 15.7.6. Kerberos-той холбоотой арга, зальнууд болон алдааг олж засварлах

- Heimdal эсвэл MITKerberos портууд ашиглах үед таны `PATH` орчны хувьсагч клиентийн програмуудын Kerberos хувилбаруудыг системийн хувилбаруудаас өмнө жагсаасан байхыг шаарддаг.
- Таны хүрээний бүх компьютерууд цагийн тохиргоонуудаа адилаар тохируулсан уу? Хэрэв үгүй бол нэвтрэлт танилт амжилтгүй болж болох юм. ntpd-р Цаг Тааруулах нь нь NTP ашиглан цагийг хамгийн сүүлийн хэлбэрт аваачиж адил болгож тохируулах талаар тайлбарладаг.
- MIT болон Heimdal нь хоорондоо сайн ажилладаг. `kadmin`-аас бусад талаараа сайн ажилладаг, учир нь энэ програмын протокол стандартчилагдаагүй.
- Та хэрэв өөрийн хостын нэрийг өөрчилбөл бас өөрийн `host/` удирдагчийг өөрчилж өөрийн keytab-ийг шинэчлэх хэрэгтэй. Энэ нь бас Апачигийн www/mod_auth_kerb-д хэрэглэгддэг `www/` удирдагч зэрэг тусгай keytab оруулгуудад хамаатай юм.
- Таны хүрээний бүх хостууд DNS-д (эсвэл хамгийн багадаа /etc/hosts-ийн хувьд) танигдаж (урагш болон эсрэгээр танигдаж) байх ёстой. CNAME-үүд ажиллах боловч A болон PTR бичлэгүүд зөв бөгөөд байрандаа байж байх ёстой. Алдааны мэдэгдэл нь тийм ч ойлгогдохоор байдаггүй, жишээ нь: `Kerberos5 refuses authentication because Read req failed: Key table entry not found буюу орчуулбал Унших Req амжилтгүй болсон болохоор Kerberos5 нь нэвтрэлт танилтаас татгалзаж байна`.
- Таны KDC-ийн хувьд магадгүй клиент маягаар харьцаж байгаа зарим үйлдлийн системүүд setuid `root` болохын тулд `ksu` тушаалд зөвшөөрлүүдийг тохируулдаггүй. Энэ нь `ksu` ажиллахгүй гэсэн үг бөгөөд аюулгүй байдлын хувьд сайн боловч залхаамаар байдаг. Энэ нь KDC-ийн алдаа биш юм.
- MITKerberos-той байхад хэрэв та анхдагч 10 цагаас арай урт амьдрах хугацаа бүхий тасалбартай удирдагчийг зөвшөөрөхийг хүсвэл `kadmin` дээр `modify_principal` тушаал ашиглан өөрчлөхийг хүссэн удирдагч болон `krbtgt` удирдагчийн maxlife-ийг өөрчлөх шаардлагатай. Дараа нь удирдагч `-l` тохируулгыг `kinit`-тай ашиглаж илүү урт амьдрах хугацаатай тасалбарыг авах хүсэлт илгээж болох юм.

Хэрэв та өөрийн KDC дээр алдааг олж засварлахын тулд пакет шиншлэгч ажиллуулж дараа нь ажлын станцаасаа `kinit`-ийг ажиллуулахад `kinit`-ийг ажилласан даруй таны TGT илгээгдэхийг - таныг бүр нууц үгээ бичихээс өмнө та харах болно! Үүний тайлбар нь Kerberos сервер чөлөөтэйгээр TGT-ийг (Ticket Granting Ticket буюу Тасалбар Баталгаажуулах Тасалбар) ямар ч танигдаагүй хүсэлтэд дамжуулдаг; гэхдээ TGT бүр хэрэглэгчийн нууц үгээс гарсан түлхүүр болон шифрлэгдсэн байдаг. Тийм болохоор хэрэглэгч өөрсдийн нууц үгийг бичихэд тэр нь KDC уруу илгээгддэггүй бөгөөд харин `kinit`-ийн аль хэдийн олж авсан TGT-г буцааж шифрлэхэд (decrypt) ашиглагддаг. Хэрэв буцааж шифрлэх процесс хүчинтэй хугацаа бүхий хүчинтэй тасалбарыг гаргаж авбал хэрэглэгч хүчинтэй Kerberos

> итгэмжлэлүүдтэй байна. Эдгээр итгэмжлэлүүд нь ирээдүйд Kerberos сервертэй аюулгүй холболтууд хийхэд зориулагдсан сессийн түлхүүр болон бас Kerberos серверийн өөрийнх нь түлхүүрээр шифрлэгдсэн тасалбар-баталгаажуулах тасалбарыг агуулдаг. Шифрлэлтийн хоёр дахь давхарга нь хэрэглэгчид мэдэгддэггүй, гэхдээ энэ нь TGT бүрийн жинхэнийг шалгахыг Kerberos серверт зөвшөөрч байгаа тэр зүйл юм.

- Хэрэв та урт амьдрах хугацаатай (жишээ нь долоо хоног) тасалбар ашиглахыг хүсэж байгаа бөгөөд та тасалбар хадгалагдаж байгаа машин уруу OpenSSH ашиглан холбогдож байгаа бол Kerberos `TicketCleanup` тохируулга `no` гэж sshd_config тохиргооны файлд байгаа эсэхийг шалгаарай, тэгэхгүй бол таны тасалбарууд таныг гарах үед устгагдах болно.
- Хостын удирдагчид илүү урт амьдрах хугацаатай тасалбартай бас байж болно гэдгийг санаарай. Хэрэв таны хэрэглэгчийн удирдагч долоо хоног амьдрах хугацаатай бөгөөд гэхдээ таны холбогдож байгаа хост 9 цаг амьдрах хугацаатай бол та кэшдээ хугацаа нь дууссан хостын удирдагчтай болж тасалбарын кэш хүссэнээр ажиллахгүй болох болно.
- Тусгайлсан муу нууц үгүүдийг ашиглуулахгүйн тулд (`kadmind` тушаалын гарын авлагын хуудас үүнийг товчхон тайлбарладаг) krb5.dict файлыг тохируулахдаа нууц үгийн бодлого тавигдсан удирдагчдад энэ нь зөвхөн хамаатайг санах хэрэгтэй. krb5.dict файлуудын хэлбэр хялбар байдаг: нэг мөрт нэг үг (string) байна. /usr/shared/dict/words симболын холбоос үүсгэх нь ашигтай байж болох юм.

### 15.7.7. MIT портоос ялгаатай талууд

MIT болон Heimdal суулгацуудын гол ялгаа нь өөр (гэхдээ орлуулж болох) тушаалууд болон өөр протоколууд ашигладаг `kadmin` програмтай холбоотой юм. Хэрэв таны KDC нь MIT бол та Heimdal `kadmin` програмыг ашиглаж өөрийн KDC-г алсаас (эсвэл эсрэг чиглэлд энэ зорилгоор) удирдаж чадахгүй болдог учир энэ нь их хамаатай юм.

Клиент програмууд нь бас шал өөр өөр тушаалын мөрийн тохируулгууд авч адил үүргийг гүйцэтгэж болох юм. MITKerberos вэб сайт (http://web.mit.edu/Kerberos/www/) дээрх зааврууды дагахыг зөвлөж байна. Замын асуудлуудаас болгоомжлоорой: MIT порт нь анхдагчаар /usr/local/ уруу суудаг бөгөөд хэрэв таны `PATH` орчны хувьсагч системийн сангуудыг эхлээд жагсаадаг бол "жирийн" системийн програмууд MIT-ийн оронд ажиллаж болохыг санаарай.

> `telnetd` болон `klogind`-ээр нэвтрэх нэвтрэлтүүд нэг л хачин байдаг тэр шалтгааныг ойлгохыг хүсвэл FreeBSD-ийн хангадаг MITsecurity/krb5 портын суулгасан /usr/local/shared/doc/krb5/README.FreeBSD файлыг унших хэрэгтэй. Хамгийн чухал нь "кэш файл дахь буруу зөвшөөрлүүд"ийг зөв болгох нь дамжуулагдсан итгэмжлүүдийн эзэмшилтийг зөвөөр солих `login.krb5` хоёртын файлыг нэвтрэлт танилтад ашиглахыг шаарддаг.

rc.conf файл дараах тохиргоог агуулж засварлагдсан байх бас шаардлагатай:

```
kerberos5_server="/usr/local/sbin/krb5kdc"
kadmind5_server="/usr/local/sbin/kadmind"
kerberos5_server_enable="YES"
```

```
kadmind5_server_enable="YES"
```

MIT керберосд зориулсан програмууд /usr/local санд хоёртын файлуудыг суулгадаг болохоор ингэж хийгддэг.

## 15.7.8. Kerberos дахь хязгаарууд багасгах

### 15.7.8.1. Kerberos нь бүгдийг эсвэл юуг ч биш гэсэн арга юм

Сүлжээнд идэвхжүүлэгдсэн үйлчилгээ бүр Kerberos-тэй ажиллахаар засварлагдсан (эсвэл сүлжээний халдлагуудын эрэг аюулгүй байдлыг хангасан) байх шаардлагатай, тэгэхгүй бол хэрэглэгчдийн итгэмжлэлүүд хулгайлагдаж дахин ашиглагдаж болох юм. Үүний нэг жишээ нь бүх алсын бүрхүүлүүдийг (жишээ нь `rsh` болон `telnet`) Kerberos хийн идэвхжүүлсэн мөртлөө нууц үгүүдийг цэвэр текстээр илгээдэг POP3 захидлын серверийг тэгж хувиргахгүй байх явдал юм.

### 15.7.8.2. Kerberos нь ганц хэрэглэгчийн ажлын станцуудад зориулагдсан

Олон хэрэглэгчийн орчинд Kerberos нь тийм ч аюулгүй биш юм. Энэ нь тасалбаруудыг бүх хэрэглэгчийн хувьд уншигдаж болох /tmp санд хадгалдаг учраас тэр юм. Хэрэв хэрэглэгч компьютераа хэд хэдэн бусад хүмүүстэй зэрэг харилцан хуваалцаж байвал (өөрөө хэлбэл олон-хэрэглэгч) хэрэглэгчийн тасалбаруудыг өөр хэрэглэгч хулгайлах (хуулан авах) боломжтой юм.

Үүнийг `-c` файлын нэрийн тушаалын мөрийн тохируулгатай эсвэл (илүү зохимжтой) `KRB5CCNAME` орчны хувьсагчтайгаар даван гарч болох юм, гэхдээ ингэх нь их ховор байдаг. Зарчмын хувьд тасалбарыг хэрэглэгчдийн гэр санд хадгалж хялбар файлын зөвшөөрлүүдийг ашиглах нь энэ асуудлыг багасгадаг.

### 15.7.8.3. KDC нь бүтэлгүйтлийн ганц цэг

Дизайнаараа бол KDC нь мастер нууц үгийн мэдээллийн баазаас тогтох бөгөөд түүний нэгэн адил аюулгүй байх ёстой. KDC нь үүн дээр өөр ямар ч үйлчилгээнүүд ажиллуулсан байх ёсгүй бөгөөд физикээр аюулгүй байдлыг нь хангасан байх шаардлагатай. Kerberos нь ижил түлхүүрээр ("мастер" түлхүүр) шифрлэгдсэн бүх нууц үгүүдийг хадгалдаг бөгөөд тэр ижил түлхүүр нь эргээд KDC дээр файл маягаар хадгалагддаг учраас аюул өндөртэй байдаг.

Тэмдэглэн хэлэхэд булаан эзлэгдсэн мастер түлхүүр нь хэн нэг нь айхаар тийм ч муу биш юм. Түлхүүр үг нь зөвхөн Kerberos мэдээллийн баазыг шифрлэхэд болон санамсаргүй тоо үүсгэгчийн үр болон хэрэглэгддэг. Таны KDC уруу хандахад аюулгүй л байж байвал халдагч мастер түлхүүрээр их юм хийж чадахгүй.

Мөн нэмж хэлэхэд хэрэв KDC нь боломжгүй байвал (магадгүй үйлчилгээ зогсоох халдлага эсвэл сүлжээний асуудлуудаас болоод) сүлжээний үйлчилгээнүүд нь нэвтрэлт танилтыг хийж болохгүй болохоор хэрэглэгдэх боломжгүй болох бөгөөд нэг ёсны үйлчилгээ зогсоох халдлагын рецепт болох юм. Үүнийг олон KDC-тэй (нэг мастер болон нэг буюу хэд хэдэн боолууд) болон хоёрдогч эсвэл нэмэлт, эцсийн нэвтрэлт таних (PAM нь энэнд маш сайн) болгоомжтой шийдлийн тусламжтайгаар даван гарч болох юм.

#### 15.7.8.4. Kerberos-ийн дутагдлууд

Kerberos нь хэрэглэгчид, хостууд болон үйлчилгээнүүдэд өөр хоорондоо бие биенээ таниулах боломжийг олгодог. Гэхдээ энэ нь KDC-г хэрэглэгчид, хостууд эсвэл үйлчилгээнүүдэд таниулах аргагүй юм. Энэ нь троян хийгдсэн `kinit` (жишээ нь) тушаал бүх хэрэглэгчийн нэрс болон нууц үгүүдийг бүртгэн бичиж авч болно гэсэн үг юм. security/tripwire ч юм уу эсвэл өөр бусад файлын системийн бүрэн бүтэн байдлыг шалгах хэрэгслүүд үүнийг арилгаж чадна.

### 15.7.9. Эх сурвалжууд болон нэмэлт мэдээллүүд

- Kerberos-ийн FAQ
- Танин шалгах системийг дизайн хийх нь: Дөрвөн үзэгдэл дэх харилцан яриа (диалог)
- RFC 1510, Kerberos Сүлжээний Танин Шалгах Үйлчилгээ (V5)
- MIT Kerberos-ийн гэр хуудас
- Heimdal Kerberos-ийн гэр хуудас

## 15.8. OpenSSL

Олон хэрэглэгчдийн хайдаг нэг боломж нь FreeBSD-д байдаг OpenSSL багаж юм. OpenSSL нь ердийн холбооны давхарга дээр шифрлэлт дамжуулах давхаргыг хангаж өгдөг; ингэснээр түүнийг сүлжээний програмууд болон үйлчилгээнүүдтэй холбож өгөх боломжийг олгодог.

OpenSSL-ийн зарим нэг хэрэглээнд захидлын клиентүүдийн шифрлэсэн нэвтрэлт, кредит картаар хийх төлбөрүүд гэх мэт вэб дээр тулгуурласан шилжүүлгүүд зэрэг олныг дурдаж болно. www/apache22 болон mail/claws-mail зэрэг олон портууд нь OpenSSL-тэй бүтээх эмхэтгэлийн дэмжлэгийг санал болгодог.

Ихэнх тохиолдолд Портуудын Цуглуулга нь make хувьсагч `WITH_OPENSSL_BASE` -ийг "yes" гэж заагаагүй тохиолдолд security/openssl портыг бүтээхийг оролддог.

FreeBSD-д орсон OpenSSL-ийн хувилбар нь Secure Sockets Layer v2/v3 (SSLv2/SSLv3) буюу Аюулгүй Сокетуудын Давхаргын v2/v3 хувилбарууд, Transport Layer Security v1 (TLSv1) буюу Тээврийн Давхаргын Аюулгүй байдлын v1 хувилбарын сүлжээний аюулгүй байдлын протоколуудыг дэмждэг бөгөөд ерөнхий криптограф сан болон ашиглагдаж болох юм.

OpenSSL нь IDEA алгоритмийг дэмждэг боловч Нэгдсэн Улсын патентуудаас болоод анхдагчаар хаалттай байдаг. Үүнийг ашиглахын тулд лицензийг шалгасан байх ёстой бөгөөд хэрэв хязгаарлалтуудыг хүлээн авах боломжтой бол `MAKE_IDEA` хувьсагчийг make.conf файлд заагж өгөх ёстой байдаг.

OpenSSL-ийн хамгийн түгээмэл хэрэглээний нэг бол програм хангамжуудад зориулан ашиглах сертификатуудыг бэлдэх явдал юм. Эдгээр сертификатууд нь компани болон хувь хүмүүсийн итгэмжлэлүүдийг хүчинтэй бөгөөд луйврын биш гэдгийг баталгаажуулдаг. Хэрэв асуудалтай сертификат хэд хэдэн "Certificate Authorities" эсвэл CA-ууд буюу Сертификатын

Эрх мэдэлтнүүдээр шалгагдаагүй бол ихэвчлэн анхааруулга үзүүлдэг. Сертификатын Эрх мэдэлтэн нь VeriSign зэрэг компани байдаг бөгөөд компаниуд эсвэл хувь хүмүүсийн итгэмжлэлүүдийг хүчин төгөлдөр болгохын тулд сертификатуудыг баталгаажуулж өгдөг. Энэ процесс нь өртөгтэй бөгөөд сертификатууд ашиглахад заавал ч үгүй шаардлага болдоггүй; гэхдээ энэ нь паранойд буюу хэт зовнисон хэрэглэгчдийн заримын санааг тайвшруулж болох юм.

### 15.8.1. Сертификатуудыг үүсгэх нь

Сертификат үүсгэхийн тулд дараах тушаал байдаг:

```
# openssl req -new -nodes -out req.pem -keyout cert.pem
Generating a 1024 bit RSA private key
..............++++++
.....................................++++++
writing new private key to 'cert.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:PA
Locality Name (eg, city) []:Pittsburgh
Organization Name (eg, company) [Internet Widgits Pty Ltd]:My Company
Organizational Unit Name (eg, section) []:Systems Administrator
Common Name (eg, YOUR name) []:localhost.example.org
Email Address []:trhodes@FreeBSD.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:SOME PASSWORD
An optional company name []:Another Name
```

"Common Name" хүлээх мөрийн дараах хариу домэйны нэрийг харуулж байгааг анзаараарай. Энэ мөр нь шалгалт хийх зорилгоор серверийн нэрийг оруулахыг шаарддаг; домэйн нэрээс бусдыг байрлуулах нь ашиггүй сертификат үүсэхэд хүргэдэг. Бусал тохируулгууд, жишээ нь дуусах хугацаа, өөр шифрлэх алгоритмууд гэх мэт тохируулгууд байдаг. Бүрэн гүйцэд жагсаалтыг openssl(1) гарын авлагын хуудсыг үзэн авч болно.

Дээрх тушаалын ажилласан санд хоёр файл одоо байж байх ёстой. Сертификатын хүсэлт req.pem нь таны оруулсан итгэмжлэлүүдийг хүчин төгөлдөр болгож хүсэлтийг баталгаажуулан сертификатыг танд буцаах сертификатын эрх мэдэлтэн уруу илгээгдэж болно. Үүсгэгдсэн хоёр дахь файл нь cert.pem гэж нэрлэгдэн сертификатын хувийн түлхүүр болох бөгөөд ямар ч байсан гэсэн хамгаалагдсан байх ёстой; хэрэв энэ нь бусдын гарт орох юм бол таны (эсвэл таны серверийн) дүрд тоглон ашиглагдаж болох юм.

CA-с гарын үсэг шаарддаггүй тохиолдолд өөрөө зурсан сертификатыг үүсгэж болно. Эхлээд RSA түрхүүр үүсгэх хэрэгтэй:

```
# openssl dsaparam -rand -genkey -out myRSA.key 1024
```

Дараа нь CA түлхүүр үүсгэ:

```
# openssl gendsa -des3 -out myca.key myRSA.key
```

Сертификат үүсгэхийн тулд энэ түлхүүрийг ашигла :

```
# openssl req -new -x509 -days 365 -key myca.key -out new.crt
```

Санд хоёр шинэ файл үүсэх ёстой: сертификатын эрх мэдэлтний гарын үсгийн файл myca.key болон сертификат өөрөө new.crt байна. Эдгээрийг зөвхөн `root` унших эрхтэй /etc санд байрлуулах шаардлагатай. Үүнд 0700 зөвшөөрөл байж болох бөгөөд түүнийг `chmod` хэрэгсэл ашиглан тохируулж болно.

### 15.8.2. Сертификатуудыг ашиглах нь, жишээ

Тэгэхээр эдгээр файлууд нь юу хийж чадах вэ? Сайн хэрэглээ болох нэг жишээ нь SendmailMTA уруу хийгдэх холболтуудыг шифрлэх байж болно. Энэ нь локал MTA ашиглан захидал илгээх хэрэглэгчдийн цэвэр текст нэвтрэлтийн хэрэглээг болиулах юм.

Зарим MUA-ууд нь хэрэв хэрэглэгчид дотроо сертификат суулгаагүй бол тэдэнд алдааг харуулдаг болохоор энэ нь ертөнц дээрх хамгийн шилдэг хэрэглээ биш юм. Сертификат суулгах тухай илүү мэдээллийг програм хангамжтай цуг ирсэн баримтаас үзэх хэрэгтэй.

Дотоод .mc файл дотор дараах мөрүүдийг байрлуулах хэрэгтэй:

```
dnl SSL Options
define(`confCACERT_PATH',`/etc/certs')dnl
define(`confCACERT',`/etc/certs/new.crt')dnl
define(`confSERVER_CERT',`/etc/certs/new.crt')dnl
define(`confSERVER_KEY',`/etc/certs/myca.key')dnl
define(`confTLS_SRV_OPTIONS', `V')dnl
```

Дээрх /etc/certs/ нь сертификат болон түлхүүр файлуудыг дотооддоо хадгалах сан юм. Сүүлийн хэдэн шаардлагууд нь дотоод .cf файлын дахин бүтээлт юм. Үүнийг /etc/mail сан дотроос `make install` тушаал бичин хийж болно. Ингэсний дараа `make restart` тушаалыг ажиллуулаарай, энэ нь Sendmail дэмонг эхлүүлэх ёстой.

Хэрэв бүгд зүгээр болж өнгөрвөл /var/log/maillog файлд ямар ч алдаа бичигдэхгүй бөгөөд Sendmail процессийн жагсаалтад харуулагдана.

Хялбар тест хийхийн тулд telnet(1) хэрэгсэл ашиглан захидлын серверт холбогдох хэрэгтэй:

```
# telnet example.com 25
Trying 192.0.34.166...
Connected to example.com.
Escape character is '^]'.
220 example.com ESMTP Sendmail 8.12.10/8.12.10; Tue, 31 Aug 2004 03:41:22 -0400 (EDT)
ehlo example.com
250-example.com Hello example.com [192.0.34.166], pleased to meet you
250-ENHANCEDSTATUSCODES
250-PIPELINING
250-8BITMIME
250-SIZE
250-DSN
250-ETRN
250-AUTH LOGIN PLAIN
250-STARTTLS
250-DELIVERBY
250 HELP
quit
221 2.0.0 example.com closing connection
Connection closed by foreign host.
```

Хэрэв "STARTTLS" мөр гарч ирвэл бүгд зөв ажиллаж байна.

# 15.9. IPsec дээгүүр VPN хийх

FreeBSD гарц машинуудыг ашиглан Интернэтээр тусгаарлагдсан хоёр сүлжээний хооронд VPN үүсгэх.

## 15.9.1. IPsec-ийг ойлгох нь

Энэ хэсэг нь IPsec-ийг тохируулах процессийг тайлбарлах болно. IPsec-ийг тохируулахын тулд та өөрчлөн тохируулсан цөм бүтээх ухагдахууныг мэдсэн байх шаардлагатай (FreeBSD цөмийг тохируулах нь-г үзнэ үү).

*IPsec* нь Интернэт Протокол (IP) давхаргын дээр суудаг протокол юм. Энэ нь хоёр буюу хэд хэдэн хостуудыг аюулгүй байдлаар (нэрээс нь харах юм бол) холбох боломжийг олгодог. FreeBSD IPsec "сүлжээний стек" нь IPv4 болон IPv6 протоколуудыг хоёуланг дэмждэг KAME шийдэл дээр үндэслэсэн.

IPsec нь хоёр дэд протоколоос тогтоно:

- *Encapsulated Security Payload (ESP) буюу Хайрцаглагдсан Аюулгүй байдлын ачаа* нь гуравдагчийн нөлөөллөөс тэгш хэмт криптограф алгоритмийг (Blowfish, 3DES-тэй адил) ашиглан агуулгыг нь шифрлэж IP пакетийн өгөгдлийг хамгаалдаг.
- *Authentication Header (AH) буюу Нэвтрэлт Танилтын Толгой* нь аюулгүй хэш хийх функцаар IP пакетийн толгойн талбаруудыг хэш хийн криптограф хянах нийлбэрийг

тооцоолон гуравдагч этгээдийн нөлөөлөл болон хууран мэхлэлтээс IP пакетийн толгойг хамгаалдаг. Үүний дараа пакет дахь мэдээллийг таниулахыг зөвшөөрөх хэшийг агуулсан нэмэлт толгой байдаг.

ESP болон AH нь орчноосоо хамааранд хоёулаа цуг эсвэл тусдаа ашиглагдаж болно.

IPsec нь хоёр хостын хоорондох урсгалыг шууд шифрлэх (*Transport Mode буюу Тээвэрлэх Горим* гэгддэг) буюу эсвэл хоёр корпорацийн сүлжээний хооронд аюулгүй холбоонд ашиглагдаж болох "виртуал туннелиуд" (*Tunnel Mode буюу Туннелийн Горим* гэгддэг) бүтээхэд хэрэглэгдэж болох юм. Сүүлийнх нь ерөнхийдөө *Виртуал Хувийн Сүлжээ (VPN)* гэгддэг. FreeBSD-ийн IPsec дэд системийн талаар дэлгэрэнгүй мэдээллийг ipsec(4) гарын авлагын хуудаснаас лавлах хэрэгтэй.

Өөрийн цөмдөө IPsec дэмжлэгийг нэмэхийн тулд та дараах тохируулгуудыг цөмийн тохиргоондоо нэмээрэй:

```
options   IPSEC        IP security
device    crypto
```

Хэрэв IPsec дибаг хийх дэмжлэг заавал хэрэгтэй бол дараах цөмийн тохируулга бас нэмэгдсэн байх шаардлагатай:

```
options   IPSEC_DEBUG  debug for IP security
```

### 15.9.2. Асуудал

VPN-ийг байгуулахад ямар нэг стандарт байхгүй. VPN-үүд нь өөр өөрийн давуу болон сул талуудтай төрөл бүрийн технологиудыг ашиглан хийгдэж болно. Энэ хэсэг нь нэг тохиолдлын загвар үзүүлэх бөгөөд энэ тохиолдол дахь VPN-ийг хийхэд хэрэглэгдэх стратегиудыг харуулах болно.

### 15.9.3. Тохиолдол: Хоёр сүлжээ, нэг нь гэрийн нэг нь ажлын. Хоёулаа Интернэтэд холбогдсон бөгөөд энэ VPN-ээр нэг юм шиг ажиллах сүлжээ.

Угтвар нөхцөл дараах маягийн байна:

- Та хамгийн багадаа хоёр сайттай байна
- Хоёр сайт хоёулаа IP-г дотооддоо ашигладаг
- FreeBSD дээр нь ажилладаг гарц компьютераар хоёр сайт хоёулаа Интернэтэд холбогдсон.
- Хоёр сүлжээний гарц компьютер бүр хамгийн багаар бодоход нэг нийтийн IP хаягтай.
- Хоёр сүлжээний дотоод хаягууд нь нийтийн эсвэл хувийн IP хаягууд байж болох юм, энэ нь хамаагүй. Тэдгээр нь давхцахгүй байх ёстой, өөрөөр хэлбэл хоёулаа `192.168.1.x`-г ашиглаж болохгүй юм.

### 15.9.4. IPsec-ийг FreeBSD дээр тохируулах нь

Эхлээд security/ipsec-tools портын цуглуулгаас суусан байх шаардлагатай. Энэ гуравдагч талын програм хангамжийн багц нь тохиргоог дэмжихэд туслах хэд хэдэн програмуудыг агуулдаг.

Дараагийн шаардлага нь пакетуудыг тунель хийх болон хоёр сүлжээг зөв холбогдоход ашиглагдах хоёр gif(4) псевдо төхөөрөмжийг үүсгэх явдал юм. `root` хэрэглэгчээр *internal* болон *external* гэсэн утгуудыг жинхэнэ дотоод болон гадаад гарцуудаар өөрчлөн дараах тушаалыг ажиллуулна:

```
# ifconfig gif0 create
```

```
# ifconfig gif0 internal1 internal2
```

```
# ifconfig gif0 tunnel external1 external2
```

Жишээ нь ажлын LAN-ий нийтийн IP нь `172.16.5.4` бөгөөд хувийн IP нь `10.246.38.1` байна. Гэрийн LAN-ий нийтийн IP нь `192.168.1.12` бөгөөд дотоод хувийн IP нь `10.0.0.5` байна.

Энэ нь толгой эргэмээр санагдаж болох бөгөөд ifconfig(8) тушаалын дараах жишээ үр дүнгээс харна уу:

```
Gateway 1:

gif0: flags=8051 mtu 1280
tunnel inet 172.16.5.4 --> 192.168.1.12
inet6 fe80::2e0:81ff:fe02:5881%gif0 prefixlen 64 scopeid 0x6
inet 10.246.38.1 --> 10.0.0.5 netmask 0xffffff00

Gateway 2:

gif0: flags=8051 mtu 1280
tunnel inet 192.168.1.12 --> 172.16.5.4
inet 10.0.0.5 --> 10.246.38.1 netmask 0xffffff00
inet6 fe80::250:bfff:fe3a:c1f%gif0 prefixlen 64 scopeid 0x4
```

Хийгдэж дууссаны дараа хоёр хувийн IP-д ping(8) тушаал ашиглан дараах үр дүнд харуулсан шиг хүрэх боломжтой байх ёстой:

```
priv-net# ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=64 time=42.786 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=64 time=19.255 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=64 time=20.440 ms
```

```
64 bytes from 10.0.0.5: icmp_seq=3 ttl=64 time=21.036 ms
--- 10.0.0.5 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 19.255/25.879/42.786/9.782 ms

corp-net# ping 10.246.38.1
PING 10.246.38.1 (10.246.38.1): 56 data bytes
64 bytes from 10.246.38.1: icmp_seq=0 ttl=64 time=28.106 ms
64 bytes from 10.246.38.1: icmp_seq=1 ttl=64 time=42.917 ms
64 bytes from 10.246.38.1: icmp_seq=2 ttl=64 time=127.525 ms
64 bytes from 10.246.38.1: icmp_seq=3 ttl=64 time=119.896 ms
64 bytes from 10.246.38.1: icmp_seq=4 ttl=64 time=154.524 ms
--- 10.246.38.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 28.106/94.594/154.524/49.814 ms
```

Хүсэн хүлээж байсны дагуу хоёр тал хоёулаа хувийн тохируулсан хаягаасаа ICMP пакетуудыг илгээх болон хүлээн авах боломжтой байна. Дараа нь аль аль сүлжээнээс урсгалыг зөв илгээдэг байхын тулд хоёр гарцад хоёуланд нь пакетуудыг хэрхэн яаж чиглүүлэхийг зааж өгөх ёстой. Энэ зорилгод дараах тушаал хүрнэ:

```
# corp-net# route add 10.0.0.0 10.0.0.5 255.255.255.0
```

```
# corp-net# route add net 10.0.0.0: gateway 10.0.0.5
```

```
# priv-net# route add 10.246.38.0 10.246.38.1 255.255.255.0
```

```
# priv-net# route add host 10.246.38.0: gateway 10.246.38.1
```

Энэ үе хүрэхэд дотоод машинууд нь аль аль гарц болон гарцын цаана байгаа машинуудаас хүрэх боломжтой байх ёстой. Үүнийг хялбараар дараах жишээнээс тодорхойлж болно:

```
corp-net# ping 10.0.0.8
PING 10.0.0.8 (10.0.0.8): 56 data bytes
64 bytes from 10.0.0.8: icmp_seq=0 ttl=63 time=92.391 ms
64 bytes from 10.0.0.8: icmp_seq=1 ttl=63 time=21.870 ms
64 bytes from 10.0.0.8: icmp_seq=2 ttl=63 time=198.022 ms
64 bytes from 10.0.0.8: icmp_seq=3 ttl=63 time=22.241 ms
64 bytes from 10.0.0.8: icmp_seq=4 ttl=63 time=174.705 ms
--- 10.0.0.8 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 21.870/101.846/198.022/74.001 ms

priv-net# ping 10.246.38.107
```

```
PING 10.246.38.1 (10.246.38.107): 56 data bytes
64 bytes from 10.246.38.107: icmp_seq=0 ttl=64 time=53.491 ms
64 bytes from 10.246.38.107: icmp_seq=1 ttl=64 time=23.395 ms
64 bytes from 10.246.38.107: icmp_seq=2 ttl=64 time=23.865 ms
64 bytes from 10.246.38.107: icmp_seq=3 ttl=64 time=21.145 ms
64 bytes from 10.246.38.107: icmp_seq=4 ttl=64 time=36.708 ms
--- 10.246.38.107 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 21.145/31.721/53.491/12.179 ms
```

Тунелиуд үүсгэж тохируулах нь хялбар хэсэг юм. Аюулгүй холбоосыг тохируулах нь илүү гүнзгий процесс юм. Дараах тохиргоо нь урьдчилан хуваалцсан (PSK) RSA түлхүүрүүдийг ашиглаж байна. IP хаягаас гадна хоёр /usr/local/etc/racoon/racoon.conf файл хоёулаа адил бөгөөд доорхтой төстэй байна.

```
path    pre_shared_key  "/usr/local/etc/racoon/psk.txt"; #location of pre-shared key
file
log     debug;  #log verbosity setting: set to 'notify' when testing and debugging is
complete

padding # options are not to be changed
{
        maximum_length  20;
        randomize       off;
        strict_check    off;
        exclusive_tail  off;
}

timer   # timing options. change as needed
{
        counter         5;
        interval        20 sec;
        persend         1;
#       natt_keepalive  15 sec;
        phase1          30 sec;
        phase2          15 sec;
}

listen  # address [port] that racoon will listening on
{
        isakmp          172.16.5.4 [500];
        isakmp_natt     172.16.5.4 [4500];
}

remote  192.168.1.12 [500]
{
        exchange_mode   main,aggressive;
        doi             ipsec_doi;
        situation       identity_only;
        my_identifier   address 172.16.5.4;
```

```
        peers_identifier        address 192.168.1.12;
        lifetime        time 8 hour;
        passive         off;
        proposal_check  obey;
#       nat_traversal   off;
        generate_policy off;

                        proposal {
                                encryption_algorithm    blowfish;
                                hash_algorithm          md5;
                                authentication_method   pre_shared_key;
                                lifetime time           30 sec;
                                dh_group                1;
                        }
}

sainfo  (address 10.246.38.0/24 any address 10.0.0.0/24 any)    # address
$network/$netmask $type address $network/$netmask $type ( $type being any or esp)
{                               # $network must be the two internal networks you are
joining.
        pfs_group       1;
        lifetime        time    36000 sec;
        encryption_algorithm    blowfish,3des,des;
        authentication_algorithm        hmac_md5,hmac_sha1;
        compression_algorithm   deflate;
}
```

Тохируулга бүрийг энэ жишээн дээр жагсаагдсантай нь тайлбарлах нь энэ баримтын хүрээнээс гадуур юм. racoon-ий тохиргооны гарын авлагын хуудсанд холбогдох мэдээлэл олон бий.

FreeBSD болон racoon нь хостуудын хооронд сүлжээний урсгалыг нууцлах болон буцааж задалж чаддаг байхын тулд SPD бодлогуудыг тохируулсан байх ёстой.

Энэ үйлдлийг дараах ажлын гарц дээрх шиг энгийн бүрхүүлийн скриптээр шийдэж болно. Энэ файлыг системийг эхлүүлэх үед ашиглах бөгөөд /usr/local/etc/racoon/setkey.conf гэж хадгалах ёстой.

```
flush;
spdflush;
# To the home network
spdadd 10.246.38.0/24 10.0.0.0/24 any -P out ipsec esp/tunnel/172.16.5.4-
192.168.1.12/use;
spdadd 10.0.0.0/24 10.246.38.0/24 any -P in ipsec esp/tunnel/192.168.1.12-
172.16.5.4/use;
```

Ингэсний дараа racoon-г хоёр гарц дээр дараах тушаал ашиглан эхлүүлнэ:

```
# /usr/local/sbin/racoon -F -f /usr/local/etc/racoon/racoon.conf -l
/var/log/racoon.log
```

Гарах үр дүнд нь доорхтой төстэй байна:

```
corp-net# /usr/local/sbin/racoon -F -f /usr/local/etc/racoon/racoon.conf
Foreground mode.
2006-01-30 01:35:47: INFO: begin Identity Protection mode.
2006-01-30 01:35:48: INFO: received Vendor ID: KAME/racoon
2006-01-30 01:35:55: INFO: received Vendor ID: KAME/racoon
2006-01-30 01:36:04: INFO: ISAKMP-SA established 172.16.5.4[500]-192.168.1.12[500]
spi:623b9b3bd2492452:7deab82d54ff704a
2006-01-30 01:36:05: INFO: initiate new phase 2 negotiation:
172.16.5.4[0]192.168.1.12[0]
2006-01-30 01:36:09: INFO: IPsec-SA established: ESP/Tunnel 192.168.1.12[0]-
>172.16.5.4[0] spi=28496098(0x1b2d0e2)
2006-01-30 01:36:09: INFO: IPsec-SA established: ESP/Tunnel 172.16.5.4[0]-
>192.168.1.12[0] spi=47784998(0x2d92426)
2006-01-30 01:36:13: INFO: respond new phase 2 negotiation:
172.16.5.4[0]192.168.1.12[0]
2006-01-30 01:36:18: INFO: IPsec-SA established: ESP/Tunnel 192.168.1.12[0]-
>172.16.5.4[0] spi=124397467(0x76a279b)
2006-01-30 01:36:18: INFO: IPsec-SA established: ESP/Tunnel 172.16.5.4[0]-
>192.168.1.12[0] spi=175852902(0xa7b4d66)
```

Тунель зөв ажиллаж байгааг шалгахын тулд нөгөө консол руу шилжиж сүлжээний урсгалыг харахын тулд tcpdump(1) ашиглан дараах тушаалыг хэрэглэнэ. em0-ийг сүлжээний интерфэйс картаараа шаардлагатай бол солиорой.

```
# tcpdump -i em0 host 172.16.5.4 and dst 192.168.1.12
```

Доорхтой төстэй өгөгдөл консол дээр гарах ёстой. Хэрэв үгүй бол асуудалтай гэсэн үг бөгөөд буцаасан өгөгдлийг дибаг хийх шаардлагатай.

```
01:47:32.021683 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com:
ESP(spi=0x02acbf9f,seq=0xa)
01:47:33.022442 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com:
ESP(spi=0x02acbf9f,seq=0xb)
01:47:34.024218 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com:
ESP(spi=0x02acbf9f,seq=0xc)
```

Энд хүрэхэд хоёр сүлжээ хүрэх боломжтой байх бөгөөд нэг сүлжээний хэсэг юм шиг харагдах болно. Хоёр сүлжээ нь аль аль нь галт ханаар хамгаалагдсан байж болох бөгөөд ингэх ч ёстой юм. Тэдгээрийн хооронд урсгалыг зөвшөөрөхийн тулд пакетуудыг нааш цааш дамжуулах дүрмүүдийг нэмэх шаардлагатай. ipfw(8) галт ханын хувьд галт ханын тохиргооны файлдаа дараах дүрмүүдийг нэмээрэй:

```
ipfw add 00201 allow log esp from any to any
ipfw add 00202 allow log ah from any to any
ipfw add 00203 allow log ipencap from any to any
ipfw add 00204 allow log udp from any 500 to any
```

Дүрмийн дугааруудыг тухайн хостын тохиргооноос хамаарч өөрчлөх шаардлагатай байж болох юм.

pf(4) эсвэл ipf(8),-ийн хэрэглэгчдийн хувьд дараах дүрмүүд үүнийг хийх болно:

```
pass in quick proto esp from any to any
pass in quick proto ah from any to any
pass in quick proto ipencap from any to any
pass in quick proto udp from any port = 500 to any port = 500
pass in quick on gif0 from any to any
pass out quick proto esp from any to any
pass out quick proto ah from any to any
pass out quick proto ipencap from any to any
pass out quick proto udp from any port = 500 to any port = 500
pass out quick on gif0 from any to any
```

Төгсгөлд нь системийг эхлүүлэх явцад VPN-ийг машин дэмжин ажиллаж эхэлдэг байлгахын тулд дараах мөрүүдийг /etc/rc.conf файлд нэмэх хэрэгтэй:

```
ipsec_enable="YES"
ipsec_program="/usr/local/sbin/setkey"
ipsec_file="/usr/local/etc/racoon/setkey.conf" # allows setting up spd policies on
boot
racoon_enable="yes"
```

# 15.10. OpenSSH

OpenSSH нь алсын машинуудад аюулгүйгээр хандах сүлжээний холболтын хэрэгслүүдийн олонлог юм. `rlogin`, `rsh`, `rcp`, болон `telnet`-ийг энэ програмаар шууд орлуулан ашиглаж болно. Мөн TCP/IP холболтууд аюулгүйгээр SSH-ээр туннель хийгдэж/дамжуулагдаж болдог. OpenSSH нь сэм чагналт, холболт булаан авалт, болон бусад сүлжээний түвшний халдлагуудыг үр дүнтэйгээр устгаж бүх трафикийг шифрлэдэг.

OpenSSH-г OpenBSD төсөл дэмжиж байдаг бөгөөд бүх сүүлийн үеийн алдааны засварууд болон шинэчлэлтүүд бүхий SSH v1.2.12 дээр тулгуурласан байдаг. Энэ програм нь SSH протокол 1 болон 2-той хоёулантай нь нийцтэй.

## 15.10.1. OpenSSH-ийг ашиглах давуу тал

telnet(1) эсвэл rlogin(1) ашиглаж байх үед сүлжээгээр илгээгдэж байгаа өгөгдөл цэвэр,

шифрлэгдээгүй хэлбэрээр байдаг. Сүлжээний шиншлэгчид клиент болон серверийн хооронд хаана ч байсан гэсэн таны хэрэглэгч/нууц үгийн мэдээлэл эсвэл таны сессээр дамжсан өгөгдлийг хулгайлж чадна. OpenSSH нь ийм асуудлаас хамгаалж төрөл бүрийн нэвтрэлт таних болон шифрлэх аргуудыг санал болгодог.

### 15.10.2. sshd-г идэвхжүүлэх

sshd нь стандарт FreeBSD суулгацын явцад харуулагдах тохируулга юм. sshd идэвхжсэн эсэхийг харахдаа rc.conf файлаас дараах мөрийг шалгаарай:

```
sshd_enable="YES"
```

Энэ нь дараагийн удаа таны систем эхлэхэд OpenSSH-д зориулсан sshd(8) дэмон програмыг дуудна. Мөн service(8) ашиглан OpenSSH-г эхлүүлэх боломжтой байдаг:

```
# service sshd start
```

### 15.10.3. SSH клиент

ssh(1) хэрэгсэл rlogin(1)-тэй адил ажилладаг.

```
# ssh user@example.com
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Host 'example.com' added to the list of known hosts.
user@example.com's password: *******
```

Нэвтрэлт нь `rlogin` эсвэл `telnet` ашиглан үүсгэгдсэн сесс шиг үргэлжлэх болно. SSH нь хэрэглэгч холбогдоход серверийн жинхэнэ эсэхийг шалгахын тулд түлхүүр хээ шалгах системийг хэрэглэдэг. Хэрэглэгч зөвхөн эхний удаа холбогдоход `yes` гэж оруулахыг шаардана. Дараа дараагийн нэвтрэлт оролдлогууд бүгд хадгалсан хээ шалгах түлхүүртэй харьцуулагдан шалгагддаг. Хэрэв хадгалсан хээ нь дараа дараагийн нэвтрэлтийн оролдлогуудаас хүлээн авсан хээнээс өөр бол SSH клиент нь танд түгшүүр өгнө. Хээнүүд ~/.ssh/known_hosts файлд эсвэл SSH v2-ийн хээнүүд ~/.ssh/known_hosts2 файлд хадгалагдана.

Анхдагчаар OpenSSH серверүүдийн сүүлийн үеийн хувилбарууд зөвхөн SSH v2 холболтуудыг хүлээн авдаг. Клиент нь хэрэв боломжтой бол 2-р хувилбарыг ашиглах бөгөөд боломжгүй бол 1-р хувилбарыг ашигладаг. `-1` эсвэл `-2` тохируулгуудыг 1-р эсвэл 2-р хувилбаруудад зориулан дамжуулан клиентэд зөвхөн аль нэгийг ашиглахыг хүчилж болно. 1-р хувилбарын нийцтэй байдал нь клиентэд хуучин хувилбаруудтай нийцтэй байх зорилгоор дэмжигдсэн байдаг.

### 15.10.4. Аюулгүй хуулбарлалт

scp(1) тушаал rcp(1)-тэй адил ажилладаг; энэ нь файлыг алсын машинаас эсвэл машин уруу, ялгаатай нь аюулгүйгээр хуулдаг.

```
#  scp user@example.com:/COPYRIGHT COPYRIGHT
user@example.com's password: *******
COPYRIGHT            100% |*****************************|  4735
00:00
#
```

Өмнөх жишээн дээр энэ хостын хувьд хээ нь аль хэдийн хадгалагдсан болохоор scp(1)-ийг энд ашиглах үед шалгагддаг.

scp(1)-ээр дамжуулсан нэмэлт өгөгдлүүд нь cp(1)-тэй адил бөгөөд эхний нэмэлт өгөгдөлд файл эсвэл файлууд, хоёр дахь дээр очих файлыг зааж өгдөг. Файл нь сүлжээгээр SSH-ээр татагддаг болохоор файлын нэг эсвэл хэд хэдэн нэмэлт өгөгдлүүд `user@host:<path_to_remote_file>` хэлбэрийг авдаг.

### 15.10.5. Тохиргоо

OpenSSH дэмон болон клиентийн системийн дагуух тохиргооны файлууд /etc/ssh санд байрладаг.

ssh_config клиентийн тохируулгуудыг тохируулдаг бөгөөд sshd_config нь дэмонг тохируулдаг.

Мөн `sshd_program` (анхдагчаар /usr/sbin/sshd) болон `sshd_flags`rc.conf тохируулгууд тохиргооны түвшнүүдийг илүүтэйгээр хангадаг.

### 15.10.6. ssh-keygen

Нууц үгүүдийг ашиглахын оронд ssh-keygen(1) нь хэрэглэгчийг шалгаж танихад DSA эсвэл RSA түлхүүрүүдийг үүсгэхэд хэрэглэгдэж болно:

```
% ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_dsa):
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_dsa.
Your public key has been saved in /home/user/.ssh/id_dsa.pub.
The key fingerprint is:
bb:48:db:f2:93:57:80:b6:aa:bc:f5:d5:ba:8f:79:17 user@host.example.com
```

ssh-keygen(1) нь шалгаж танихад хэрэглэгдэх нийтийн болон хувийн түлхүүр хослолыг үүсгэнэ. Хувийн түлхүүр ~/.ssh/id_dsa эсвэл ~/.ssh/id_rsa-д хадгалагдах бөгөөд харин нийтийн түлхүүр нь ~/.ssh/id_dsa.pub эсвэл ~/.ssh/id_rsa.pub-д DSA болон RSA түлхүүрийн төрлүүдэд зориулагдан хадгалагддаг. Тохируулга нь ажиллахын тулд нийтийн түлхүүр нь алсын машины ~/.ssh/authorized_keys файлд DSA болон RSA түлхүүрүүдийн хоёулангийнх нь хувьд хийгдэх ёстой байдаг. Үүнтэй адилаар нийтийн түлхүүрүүдийн RSA хувилбар нь

~/.ssh/authorized_keys файлд бас хийгдэх ёстой.

Энэ нь нууц үгүүдийн оронд SSH түлхүүрүүдийг ашиглан алсын машин уруу холбогдохыг зөвшөөрөх болно.

Хэрэв нэвтрэх үгнүүд ssh-keygen(1)-д ашиглагдаж байгаа бол хувийн түлхүүрийг хэрэглэхийн тулд хэрэглэгчээс нууц үгийг нэвтрэх болгонд асуудаг. ssh-agent(1) нь урт нэвтрэх үгнүүдийг дахин дахин оруулах тэр зовлонг зөөллөж чадах бөгөөд ssh-agent болон ssh-add хэсэгт тайлбарлагдсан байгаа болно.

Төрөл бүрийн тохируулгууд болон файлууд нь таны систем дээр байгаа OpenSSH-ийн хувилбаруудаас шалтгаалан өөр өөр байдаг; асуудалтай учрахгүйн тулд та ssh-keygen(1) гарын авлагын хуудаснаас лавлах хэрэгтэй.

### 15.10.7. ssh-agent болон ssh-add

ssh-agent(1) болон ssh-add(1) хэрэгслүүд нь нэвтрэх үгнүүдийг дахин дахин бичүүлэлгүйгээр SSH түлхүүрүүдийг санах ойд дуудан ашиглаж болох аргуудаар хангадаг.

ssh-agent(1) хэрэгсэл нь түүн уруу дуудагдсан хувийн түлхүүр(үүд) ашиглан жинхэнэ эсэхийг шалгах танилтыг зохицуулна. ssh-agent(1) нь өөр програмыг ачаалахад хэрэглэгдэх ёстой. Хамгийн хялбартаа энэ нь бүрхүүл эсвэл илүү дэвшилттэйгээр ашиглавал цонхны удирдагч ажиллуулж болох юм.

ssh-agent(1)-ийг бүрхүүлд ашиглахын тулд үүнийг эхлээд бүрхүүлтэй цуг нэмэлт өгөгдөл маягаар ажиллуулах шаардлагатай. Хоёрдугаарт хэн бэ гэдэг мэдээллийг (identity) ssh-add(1) -г ажиллуулан нэмэх хэрэгтэй бөгөөд түүнд хувийн түлхүүрийн нэвтрэх үгнүүдийг өгөх хэрэгтэй. Эдгээр алхмууд хийгдсэний дараа хэрэглэгч харгалзах нийтийн түлхүүр суулгагдсан дурын хост уруу ssh(1) хийж чадах болно. Жишээ нь:

```
% ssh-agent csh
% ssh-add
Enter passphrase for /home/user/.ssh/id_dsa:
Identity added: /home/user/.ssh/id_dsa (/home/user/.ssh/id_dsa)
%
```

X11 дээр ssh-agent(1) хэрэглэхийн тулд ssh-agent(1)-ийн дуудлага ~/.xinitrc-д байх шаардлагатай. Ингэснээр X11-д ачаалагдсан бүх програмуудад ssh-agent(1)-ийн үйлчилгээнүүдийг үзүүлэх болно. Жишээ ~/.xinitrc файл иймэрхүү харагдах болно:

```
exec ssh-agent startxfce4
```

Энэ нь ssh-agent(1)-ийг ажиллуулах бөгөөд тэр нь эргээд X11 эхлэх бүрт XFCE-ийг ажиллуулна. Ингэж хийгдсэний дараа өөрчлөлтүүд нь үйлчлэхийн тулд X11 дахин эхэлсний хойно өөрийн SSH түлхүүрүүдийг бүгдийг ачаалахын тулд ердөө л ssh-add(1)-ийг ажиллуулаарай.

### 15.10.8. SSH туннель хийх

OpenSSH нь шифрлэгдсэн сессийн үед өөр протоколыг хайрцаглах туннель үүсгэх чадвартай байдаг.

Дараах тушаал telnet-д зориулж туннель үүсгэхийг ssh(1)-д хэлж өгнө:

```
% ssh -2 -N -f -L 5023:localhost:23 user@foo.example.com
%
```

`ssh` тушаал дараах тохируулгуудтай хэрэглэгдэнэ:

`-2`

: `ssh`-ийг протоколын 2-р хувилбарыг ашиглахыг зааж өгнө. (хэрэв та хуучин SSH серверүүдтэй ажиллаж байгаа бол үүнийг битгий ашиглаарай)

`-N`

: Тушаал байхгүй эсвэл зөвхөн туннель гэдгийг заана. Хэрэв үүнийг орхивол `ssh` ердийн сесс эхлүүлнэ.

`-f`

: `ssh`-ийг ард, далд ажиллуулахыг заана.

`-L`

: Локал туннелийг *localport:remotehost:remoteport* загвараар зааж өгнө.

`user@foo.example.com`

: Алсын SSH сервер.

SSH туннель нь сонсох сокетийг `localhost`-ийн заагдсан порт дээр үүсгэн ажилладаг. Дараа нь локал хост/порт дээр хүлээн авсан дурын холболтыг SSH-ээр дамжуулан заасан алсын хост болон порт уруу илгээдэг.

Жишээн дээр `localhost` дээрх *5023* порт нь алсын машины `localhost` дээрх *23* порт уруу дамжуулагдаж байна. *23* нь telnet учир энэ нь SSH туннелээр аюулгүй telnet сесс үүсгэнэ.

SMTP, POP3, FTP гэх зэрэг ямар ч аюултай TCP протоколуудын гүйцэтгэлийг хялбаршуулахад үүнийг ашиглаж болно.

*Жишээ 23. SMTP-д зориулан SSH ашиглан аюулгүй туннель үүсгэх*

```
% ssh -2 -N -f -L 5025:localhost:25 user@mailserver.example.com
user@mailserver.example.com's password: *****
% telnet localhost 5025
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 mailserver.example.com ESMTP
```

Үүнийг ssh-keygen(1) болон нэмэлт хэрэглэгчийн бүртгэлүүдтэй цуг илүү үл үзэгдэх/төвөггүй SSH туннель хийх орчин үүсгэхэд ашиглаж болно. Түлхүүрүүд нь нууц үг бичихийн оронд ашиглагдаж болох бөгөөд туннелиуд нь тусдаа хэрэглэгч маягаар ажиллаж чадна.

#### 15.10.8.1. SSH туннелийн практик жишээнүүд

##### 15.10.8.1.1. POP3 сервер уруу аюулгүй хандах

Ажил дээр чинь гаднаас холболтууд хүлээн авах SSH сервер байна. Бас тэр оффисийн сүлжээнд POP3 сервер ажиллуулж байгаа захидлын сервер байна. Таны гэр болон оффисийн хоорондын сүлжээ болон сүлжээний зам итгэж болохоор эсвэл итгэж болохооргүй байж магадгүй юм. Ийм учраас та өөрийн захидлыг аюулгүй аргаар шалгах хэрэгтэй юм. Үүний шийдэл нь өөрийн оффисийн SSH сервер уруу SSH холболт үүсгэж захидлын сервер уруу туннель хийх явдал юм.

```
% ssh -2 -N -f -L 2110:mail.example.com:110 user@ssh-server.example.com
user@ssh-server.example.com's password: ******
```

Туннель эхлэн ажилласны дараа та өөрийн захидлын клиентийнхээ POP3 хүсэлтүүдийг localhost-ийн 2110 порт уруу илгээхээр зааж өгч болно. Эндэх холболт туннелээр аюулгүйгээр дамжин mail.example.com уруу илгээгдэнэ.

##### 15.10.8.1.2. Хэцүү галт ханыг тойрон гарах

Зарим сүлжээний администраторууд хэтэрхий чанга галт ханын дүрэм ашиглан зөвхөн ирж байгаа холболтууд төдийгүй гарч байгаа холболтуудыг ч бас шүүдэг. Танд алсын машинуудад зөвхөн SSH болон вэбээр аялах 22 болон 80-р портуудад хандах боломжийг өгсөн байж болох юм.

Та хөгжим цацдаг Ogg Vorbis сервер зэрэг өөр (магадгүй ажилдаа холбоогүй) үйлчилгээ уруу хандахыг магадгүй хүсэж болох юм. Хэрэв энэ Ogg Vorbis сервер нь 22 эсвэл 80-аас бусад өөр порт дээр цацаж байгаа бол та түүнд хандаж чадахгүй юм.

Үүний шийдэл нь таны сүлжээний галт ханаас гаднах машин уруу SSH холболт үүсгэж үүнийг Ogg Vorbis сервер уруу туннель хийхэд ашиглах явдал юм.

```
% ssh -2 -N -f -L 8888:music.example.com:8000 user@unfirewalled-system.example.org
user@unfirewalled-system.example.org's password: *******
```

Таны урсгал хүлээн авах клиент одоо localhost-ийн 8888 порт уруу заагдах бөгөөд тэр цаашаагаа галт ханыг амжилттайгаар гэтлэн music.example.com уруу дамжуулагдана.

### 15.10.9. AllowUsers тохируулга

Ямар хэрэглэгчид хаанаас орохыг хязгаарлаж өгөх нь зүйтэй юм. AllowUsers тохируулга нь үүнд хүрэх сайн арга юм. Жишээ нь root хэрэглэгчийг зөвхөн 192.168.1.32-оос орохыг

зөвшөөрөхийн тулд доор дурдсантай адил тохируулгыг /etc/ssh/sshd_config файлд хийх нь зүйтэй юм:

```
AllowUsers root@192.168.1.32
```

admin хэрэглэгчийг хаанаас ч орохыг зөвшөөрөхийн тулд ердөө л хэрэглэгчийн нэрийг өөрийг нь жагсааж өгнө:

```
AllowUsers admin
```

Олон хэрэглэгчид нэг мөрөнд жагсаагдах шаардлагатай:

```
AllowUsers root@192.168.1.32 admin
```

Та энэ машин уруу нэвтрэх хэрэгцээтэй хэрэглэгч бүрийг жагсааж өгөх нь чухал юм, тэгэхгүй бол тэдгээр нь орж чадахгүй болно.

/etc/ssh/sshd_config-д өөрчлөлтүүд хийснийхээ дараа sshd(8)-д өөрийн тохиргооны файлуудыг дахин дуудахыг дараах тушаалыг ажиллуулж та хэлж өгөх ёстой:

```
# service sshd reload
```

### 15.10.10. Нэмэлт унших материалууд

OpenSSH

ssh(1) scp(1) ssh-keygen(1) ssh-agent(1) ssh-add(1) ssh_config(5)

sshd(8) sftp-server(8) sshd_config(5)

## 15.11. Файлын системийн хандалт хянах жагсаалтууд(ACL-үүд)

Хормын хувилбарууд зэрэг файлын системийн өргөжүүлэлтүүдийн хамтаар FreeBSD нь Файлын системийн хандалт хянах жагсаалтуудын (ACL-ууд) аюулгүй байдлыг санал болгодог.

Хандалт Хянах Жагсаалтууд нь стандарт UNIX® зөвшөөрлийн загварыг маш нийцтэй (POSIX®.1e) аргаар өргөтгөдөг. Энэ боломж нь администраторт илүү төвөгтэй аюулгүй байдлын загвар болон түүний давуу талыг ашиглахыг зөвшөөрдөг.

UFS файлын системүүдэд ACL дэмжлэгийг идэвхжүүлэхийн тулд дараах:

```
options UFS_ACL
```

тохируулгыг цөмд эмхэтгэх шаардлагатай. Хэрэв энэ тохируулга эмхэтгэгдээгүй бол ACL-ууд дэмжих файлын системийг холбохыг оролдоход анхааруулах мэдэгдэл дэлгэцэд гардаг. Энэ тохируулга GENERIC цөмд орсон байдаг. ACL-ууд нь файлын систем дээр өргөтгөсөн шинж чанаруудыг идэвхжүүлсэн дээр тулгуурладаг. Өргөтгөсөн шинж чанарууд нь дараа үеийн UNIX® файлын систем UFS2-д төрөлхийн дэмжигдсэн байдаг.

UFS1 дээр өргөтгөсөн шинж чанаруудыг тохируулахад UFS2 дээр тохируулахтай харьцуулбал илүү удирдлагын зардал шаардлагатай байдаг. UFS2 дээрх өргөтгөсөн шинж чанаруудын ажиллагаа нь бас бодитойгоор илүү байдаг. Иймээс UFS2-г UFS1-ийн оронд хандалт хянах жагсаалтуудад ашиглахыг ерөнхийдөө зөвлөдөг.

ACL-ууд нь /etc/fstab файлд нэмэгдэж өгч болох холбох үеийн удирдлагын `acls` тугаар идэвхтэй болдог. Файлын системийн толгой дахь супер блокийн ACL-ууд тугийг өөрчлөхийн тулд tunefs(8)-ийг ашиглан шургуу замаар холбох үеийн тугийг автоматаар зааж өгч болно. Ерөнхийдөө хэд хэдэн шалтгааны улмаас супер блокийн тугийг ашиглах нь дээр байдаг:

- Холбх үеийн ACL-ууд туг дахин холболтоор өөрчлөгддөггүй (mount(8) `-u`), зөвхөн бүрэн гүйцэд umount(8) хийгдэж шинэ mount(8) хийгдсэний дараа болно. Энэ нь бас файлын системийг ашиглаж байх үед дарааллыг нь өөрчилж болохгүй гэсэн үг юм.
- fstab-д мөр байхгүй байсан ч гэсэн эсвэл төхөөрөмжүүдийн дарааллал өөрчлөгдсөн ч гэсэн супер блокийн тугийг тохируулах нь файлын системийг үргэлж ACL-уудыг идэвхтэйгээр холбоход хүргэдэг. Энэ нь файлын системийг ACL-уудыг идэвхжүүлэлгүйгээр санамсаргүйгээр холбохоос хамгаалдаг бөгөөд ингэж санамсаргүй холбох нь ACL-уудыг буруугаар албадаж тэгснээр аюулгүй байдлын асуудлуудад хүргэж болох юм.

Бид шинэ mount(8) хийлгүйгээр туг идэвхжүүлдгийг зөвшөөрөхөөр ACL-уудын ажиллагааг өөрчилж болох юм, гэхдээ бид ACL-уудыг идэвхжүүлэлгүй санамсаргүйгээр холболт хийхийг болиулахыг хүсдэг бөгөөд учир нь хэрэв та ACL-уудыг идэвхжүүлээд дараа нь болиулаад өргөтгөсөн шинж чанаруудыг устгалгүйгээр дахин идэвхжүүлбэл та өөртөө нэлээн хэцүү асуудал учруулах зүйлийг хийх болно. Ерөнхийдөө та файлын систем дээр ACL-уудыг идэвхжүүлсний дараа файлын хамгаалалтууд нь системийн хэрэглэгчдэд зориулагдсан файлуудтай нийцгүй болж болох учир тэдгээрийг болиулж болохгүй бөгөөд ACL-уудыг дахин идэвхжүүлэх нь зөвшөөрлүүд нь өөрчлөгдсөн байж болох файлуудад өмнөх ACL-уудыг магадгүй дахин холбож өөр тааварлаж болшгүй ажиллагаанд хүргэж болох юм.

ACL-ууд идэвхжүүлсэн файлын системүүд өөрсдийн зөвшөөрлийн тохируулгууд дээрээ `+` (нэмэх) тэмдэг үзэх үед харуулдаг. Жишээ нь:

```
drwx------  2 robert  robert  512 Dec 27 11:54 private
drwxrwx---+ 2 robert  robert  512 Dec 23 10:57 directory1
drwxrwx---+ 2 robert  robert  512 Dec 22 10:20 directory2
```

```
drwxrwx---+ 2 robert  robert  512 Dec 27 11:57 directory3
drwxr-xr-x  2 robert  robert  512 Nov 10 11:54 public_html
```

Энд бид directory1, directory2, болон directory3 сангууд бүгд ACL-ууд-ийн давуу талыг авч байгааг харж байна. public_html сан тэгэхгүй байна.

### 15.11.1. ACL-уудыг ашиглах нь

Файлын системийн ACL-уудыг getfacl(1) хэрэгслээр харж болно. Жишээ нь test файл дээрх ACL тохируулгуудыг харахын тулд дараах тушаалыг ажиллуулах хэрэгтэй:

```
% getfacl test
    #file:test
    #owner:1001
    #group:1001
    user::rw-
    group::r--
    other::r--
```

Энэ файлын ACL тохируулгуудыг өөрчлөхийн тулд setfacl(1) хэрэгслийг ажиллуул. Ажиглаарай:

```
% setfacl -k test
```

`-k` туг нь тухайн үед тодорхойлогдсон бүх ACL-уудыг файл эсвэл файлын системээс арилгана. Илүү дээр арга бол ACL-уудыг ажиллуулахад шаардлагатай үндсэн талбаруудыг орхидог `-b` тугийг ашиглах явдал юм.

```
% setfacl -m u:trhodes:rwx,group:web:r--,o::--- test
```

Дээр дурдсан тушаал дээр `-m` тохируулга анхдагч ACL оруулгуудыг өөрчлөхөд хэрэглэгдсэн. Өмнөх тушаалаар устгагдсан болохоор урьдчилан тодорхойлсон оруулгууд байхгүй учир энэ нь анхдагч тохируулгуудыг сэргээж жагсаасан тохируулгуудаас зааж өгдөг. Хэрэв та систем дээр байхгүй хэрэглэгч эсвэл бүлэг нэмэх бол `Invalid argument` буюу Буруу нэмэлт өгөгдөл гэсэн алдаа stdout уруу хэвлэгдэнэ гэдгийг санаж байх хэрэгтэй.

## 15.12. Гуравдагч талын аюулгүй байдлын асуудлуудыг монитор хийх нь

Сүүлийн жилүүдэд эмзэг асуудлын үнэлгээ хэрхэн зохицуулагдаж байгаа тал дээр аюулгүй байдлын ертөнц олон сайжруулалт хийсэн. Одоогийн байгаа бүх л үйлдлийн системүүд дээр гуравдагч талын хэрэгслүүд суулгаж тохируулдгаас болж системийн халдлагын заналхийлэл ихэсдэг.

Эмзэг асуудлын үнэлгээ нь аюулгүй байдлын түлхүүр хүчин зүйл бөгөөд FreeBSD нь үндсэн

системд зориулан зөвлөгөөнүүдийг гаргадаг боловч гуравдагч талын хэрэгслүүд бүрийн хувьд хийх нь FreeBSD төслийн боломжоос гадуур юм. Мэдэгдэж байгаа асуудлуудыг администраторуудад анхааруулж гуравдагч талын эмзэг асуудлуудыг зөөлрүүлэх арга байдаг. FreeBSD-д нэмэлтээр Portaudit гэгддэг хэрэгсэл зөвхөн энэ зорилгоор байдаг.

ports-mgmt/portaudit порт нь FreeBSD-ийн аюулгүй байдлын баг болон портуудын хөгжүүлэгчдийн шинэчилж ажиллагааг нь хангаж байдаг мэдээллийн баазаас мэдэгдэж байгаа аюулгүй байдлын асуудлуудыг шалгадаг.

Portaudit-г ашиглаж эхлэхийн тулд Портуудын цуглуулгаас түүнийг суулгах хэрэгтэй:

```
# cd /usr/ports/ports-mgmt/portaudit && make install clean
```

Суулгах процессийн явцад өдөр бүрийн аюулгүй байдлыг шалгах ажиллагаанд Portaudit-н гаралтыг зөвшөөрч periodic(8)-д зориулсан тохиргооны файлуудыг шинэчилдэг. Өдөр тутмын аюулгүй байдлыг шалгах ажиллагаа `root`-ийн захидлын бүртгэл уруу цахим захидал явуулж түүнийг уг хэрэглэгч уншсан эсэхийг баталгаажуулах хэрэгтэй. Өөр ямар ч илүү тохиргоо энд хэрэггүй.

Суулгасны дараа администратор мэдээллийн баазыг шинэчлэх болон суулгасан багцуудад мэдэгдэж байгаа эмзэг асуудлуудыг үзэхдээ дараах тушаалыг ажиллуулна:

```
# portaudit -Fda
```

Мэдээллийн бааз periodic(8) ажиллах үед автоматаар шинэчлэгддэг; иймээс дээрх тушаал заавал шаардлагагүй юм. Энэ нь зөвхөн дараах жишээнүүдэд шаардлагатай.

Портуудын цуглуулгын хэсэг болгон суулгагдсан гуравдагч талын хэрэгслүүдийг ямар ч үед аудит хийхдээ администратор зөвхөн дараах тушаалыг ажиллуулах хэрэгтэй:

```
# portaudit -a
```

Portaudit эмзэг асуудалтай багцын хувьд доор дурдсантай адилыг гаргана:

```
Affected package: cups-base-1.1.22.0_1
Type of problem: cups-base -- HPGL buffer overflow vulnerability.
Reference: <http://www.FreeBSD.org/ports/portaudit/40a3bca2-6809-11d9-a9e7-
0001020eed82.html>

1 problem(s) in your installed packages found.

You are advised to update or deinstall the affected package(s) immediately.
```

Үзүүлсэн URL уруу вэб хөтчийг чиглүүлж администратор асуудалтай байгаа эмзэг асуудлын

талаар дэлгэрэнгүй мэдээллийг олж авч болно. Ийм мэдээлэл нь нөлөөлөх хувилбарууд болон FreeBSD-ийн портын хувилбар, аюулгүй байдлын зөвлөгөөнүүд байж болох өөр бусад вэб сайтуудыг агуулж болох юм.

Товчхондоо Portaudit нь хүчирхэг хэрэгсэл бөгөөд Portupgrade порттой цуг хэрэглэхэд маш ашигтай байдаг.

# 15.13. FreeBSD-ийн аюулгүй байдлын зөвлөгөөнүүд

Үйлдвэрлэлийн чанарыг хангасан үйлдлийн системүүдийн нэгэн адил FreeBSD "Аюулгүй байдлын зөвлөгөөнүүд" гаргадаг. Эдгээр зөвлөгөөнүүд нь ихэвчлэн аюулгүй байдлын жагсаалтууд уруу илгээгддэг бөгөөд зөвхөн тохирох хувилбаруудад засвар хийгдсэний дараа Errata буюу алдааны хуудсанд тэмдэглэгддэг. Энэ хэсэгт зөвлөгөө гэж юу болох, түүнийг хэрхэн ойлгох болон системд засвар хийхдээ ямар арга хэмжээнүүдийг авах талаар тайлбарлах болно.

## 15.13.1. Зөвлөгөө ямархуу харагдах вэ?

FreeBSD-ийн аюулгүй байдлын зөвлөгөөнүүд FreeBSD аюулгүй байдлын мэдэгдлүүд захидлын жагсаалт захидлын жагсаалтаас авсан доорх зөвлөгөөтэй адил харагдах болно.

```
=============================================================================
FreeBSD-SA-XX:XX.UTIL                                          Security Advisory
                                                               The FreeBSD Project

Topic:          denial of service due to some problem①

Category:       core②
Module:         sys③
Announced:      2003-09-23④
Credits:        Person⑤
Affects:        All releases of FreeBSD⑥
                FreeBSD 4-STABLE prior to the correction date
Corrected:      2003-09-23 16:42:59 UTC (RELENG_4, 4.9-PRERELEASE)
                2003-09-23 20:08:42 UTC (RELENG_5_1, 5.1-RELEASE-p6)
                2003-09-23 20:07:06 UTC (RELENG_5_0, 5.0-RELEASE-p15)
                2003-09-23 16:44:58 UTC (RELENG_4_8, 4.8-RELEASE-p8)
                2003-09-23 16:47:34 UTC (RELENG_4_7, 4.7-RELEASE-p18)
                2003-09-23 16:49:46 UTC (RELENG_4_6, 4.6-RELEASE-p21)
                2003-09-23 16:51:24 UTC (RELENG_4_5, 4.5-RELEASE-p33)
                2003-09-23 16:52:45 UTC (RELENG_4_4, 4.4-RELEASE-p43)
                2003-09-23 16:54:39 UTC (RELENG_4_3, 4.3-RELEASE-p39) ⑦
CVE Name:   CVE-XXXX-XXXX ⑧

For general information regarding FreeBSD Security Advisories,
including descriptions of the fields above, security branches, and the
following sections, please visit
http://www.FreeBSD.org/security/.
```

I. Background ⑨

II. Problem Description ⑩

III. Impact ⑪

IV. Workaround ⑫

V. Solution ⑬

VI. Correction details ⑭

VII. References ⑮

① `Topic` буюу сэдэв талбар асуудал юу болохыг яг заасан байдаг. Энэ нь үндсэндээ тухайн үеийн аюулгүй байдлын зөвлөгөөний танилцуулга бөгөөд эмзэг асуудалтай цуг хэрэгслийг тэмдэглэдэг.

② The `Category` буюу зэрэглэл талбар нь хамаарч байгаа системийн хэсгийг хэлдэг бөгөөд `core`, `contrib`, эсвэл `ports`-ийн аль нэг байж болно. `core` зэрэглэл нь эмзэг асуудал FreeBSD үйлдлийн системийн гол хэсэгт нөлөөлнө гэсэн үг юм. `contrib` зэрэглэл нь эмзэг асуудал sendmail зэрэг FreeBSD төсөлд хувь нэмэр болгон оруулсан програм хангамжуудад нөлөөлнө гэсэн үг юм. Эцэст нь `ports` зэрэглэл нь эмзэг асуудал портуудын цуглуулганд ордог нэмэлт програм хангамжуудад нөлөөлөхийг харуулдаг.

③ `Module` талбар нь бүрэлдэхүүн хэсгийн байрлалыг жишээ нь `sys` гэх зэргээр илэрхийлдэг. Энэ жишээн дээр `sys` модуль өртөхийг бид харж байгаа бөгөөд ийм учраас энэ эмзэг асуудал нь цөм дотор ашиглагдсан бүрэлдэхүүн хэсэгт нөлөөлөх юм.

④ `Announced` буюу зарласан талбар нь аюулгүй байдлын зөвлөгөө хэвлэгдсэн эсвэл ертөнцөд зарлагдсан огноог заадаг. Энэ нь аюулгүй байдлын баг асуудал байгааг шалгаж үүний засвар FreeBSD-ийн эх модны архивт итгэмжлэн оруулсныг тогтоосон гэсэн үг юм.

⑤ `Credits` буюу талархал талбар нь эмзэг асуудлыг мэдэж тайлагнасан хувь хүн болон байгууллагыг зааж талархдаг.

⑥ `Affects` буюу нөлөөлөх хувилбарын талбар нь энэ эмзэг асуудал нөлөөлөх FreeBSD-ийн хувилбаруудыг тайлбарладаг. Цөмийн хувьд уг нөлөөлсөн файлууд дээр ажиллуулсан `ident` тушаалын үр дүнг зэрвэс харж хувилбарыг тодорхойлж болно. Портуудын хувьд /var/db/pkg санд портын нэрийн дараа хувилбарын дугаар байдаг. Хэрэв систем нь FreeBSD-ийн Subversion архивтай адил хамгийн сүүлийн хэлбэрт орж өдөр тутам дахин бүтээгдээгүй бол энэ нь нөлөөлөлд орсон хэвээр байх магадлалтай юм.

⑦ `Corrected` буюу засварласан талбар нь огноо, цаг, цагийн бүс болон засварласан хувилбаруудыг заадаг. Common Vulnerabilities Database system буюу Нийтлэг Эмзэг асуудлуудын Мэдээллийн Баазын системээс эмзэг асуудлуудыг хайхад хэрэглэгдэх магадлалын мэдээлэлд нөөцлөгддөг.

⑧ `Background` талбар нь нөлөөлөлд яг ямар хэрэгсэл орсон талаар мэдээлэл өгдөг. Ихэнхдээ энэ нь FreeBSD-д яагаад тухайн хэрэгсэл байдаг, юунд хэрэглэгддэг болон хэрэгсэл хэрхэн бий болсон талаар байдаг.

⑨ `Problem Description` буюу асуудлын тайлбар талбар нь аюулгүй байдлын цоорхойг гүнзгий

тайлбарладаг. Энэ нь гажигтай кодын мэдээлэл эсвэл бүр хэрэгслийг хэрхэн хорлонтойгоор ашиглаж аюулгүй байдлын цоорхой нээдэг тухай мэдээллийг агуулдаг.

⑩ `Impact` буюу үйлчлэл талбар нь асуудал системд ямар төрлийн үйлчлэл үзүүлдгийг тайлбарладаг. Жишээ нь энэ нь үйлчилгээг зогсоох халдлагаас авахуулаад хэрэглэгчдэд өгч болох нэмэлт зөвшөөрлүүд эсвэл халдагчид супер хэрэглэгчийн хандалт өгөх зэрэг юу ч байж болно.

⑪ `Workaround` буюу тойрон гарах талбар нь боломжит тойрон гарах арга замыг системийг шинэчилж чадахгүй байж болох системийн администраторуудад олгодог. Энэ нь хугацааны шаардлагууд, сүлжээний боломж эсвэл өөр бусад олон шалтгаанаас болдог байж болох юм. Ямар ч байсан гэсэн аюулгүй байдлыг хөнгөнөөр авч үзэж болохгүй бөгөөд нөлөөлөлд орсон систем эсвэл засвар нөхөөс хийгдэх аль эсвэл аюулгүй байдлын цоорхойг тойрон гарах шийдэл хийгдэх шаардлагатай.

⑫ `Solution` буюу шийдэл талбар нь нөлөөлөлд орсон системийг засварлах зааврууд санал болгодог. Энэ нь системд засвар нөхөөс хийн аюулгүй ажиллуулах алхам алхмаар тест хийгдэж шалгагдсан арга юм.

⑬ `Correction Details` буюу засварын нарийн учир талбар нь Subversion салбар эсвэл хувилбарын нэрийн цэгүүдийг доогуур зураас тэмдэгтээр өөрчилж үзүүлдэг. Мөн энэ нь салбар болгон дахь нөлөөлөлд орсон файлуудын хувилбарын дугаарыг бас харуулдаг.

⑭ `References` буюу лавлагаа талбар нь ихэвчлэн бусад мэдээллийн эхүүдийг өгдөг. Энэ нь вэбийн URL-ууд, номнууд, захидлын жагсаалтууд болон мэдээний бүлгүүдийг агуулж болно.

# 15.14. Процессийн бүртгэл хөтлөх

Процессийн бүртгэл хөтлөх аюулгүй байдлын аргыг ашиглаж администраторууд системийн эх үүсвэрүүдийг ашигласан байдал болон тэдгээрийг хэрэглэгчдэд хэрхэн хуваарилсныг мэдэж болох бөгөөд энэ нь системийг монитор хийх боломжийг олгодог. Мөн энэ арга нь хэрэглэгчдийн тушаалуудыг туйлын багаар мөшгих боломжийг администраторуудад олгодог.

Энэ нь үнэн хэрэгтээ өөрийн эерэг болон сөрөг талуудтай. Эерэг талуудын нэг нь халдлагыг орсон цэг хүртэл нарийсган олох боломж юм. Сөрөг тал нь процессийн бүртгэл хөтлөлтөөр үүссэн бүртгэлүүд бөгөөд тэдгээр нь дискний зай шаардаж болох юм. Энэ хэсэг процессийн бүртгэл хөтлөлтийн үндсүүдийг администраторуудад таниулах болно.

## 15.14.1. Процессийн бүртгэл хөтлөлтийг идэвхжүүлж хэрэглэх нь

Процессийн бүртгэл хөтлөлтийг ашиглаж эхлэхээсээ өмнө үүнийг идэвхжүүлэх хэрэгтэй. Үүнийг хийхийн тулд дараах тушаалуудыг ажиллуул:

```
# touch /var/account/acct

# accton /var/account/acct

# echo 'accounting_enable="YES"' >> /etc/rc.conf
```

Идэвхтэй болгосны дараа бүртгэл хөтлөлт CPU статистикууд, тушаалууд гэх мэтийг даган мөшгиж эхэлнэ. Бүртгэлийн бүх бичлэгүүд уншиж болохооргүй хэлбэрээр байдаг бөгөөд тэдгээрийг sa(8) хэрэгсэл ашиглан үзэж болдог. Ямар нэг тохируулгагүйгээр ажиллуулбал `sa` тушаал нь хэрэглэгч болгоны дуудлагуудын тоо, нийт зарцуулсан хугацааг минутаар, нийт CPU болон хэрэглэгчийн хугацааг минутаар, дундаж I/O үйлдлүүдийн тоо гэх мэттэй холбоотой мэдээллийг дэлгэцэнд хэвлэн үзүүлдэг.

Тушаалуудыг ашигласан тухай мэдээллийг харахын тулд lastcomm(1) хэрэгслийг ашиглах хэрэгтэй. `lastcomm` тушаал нь тухайн ttys(5) дээр хэрэглэгчдийн ажиллуулсан тушаалуудыг үзүүлэхэд хэрэглэгдэж болно, жишээ нь:

```
# lastcomm ls
    trhodes ttyp1
```

Дээрх тушаал нь `ttyp1` терминал дээр `trhodes` хэрэглэгчийн `ls` тушаал ашигласан мэдэгдэж байгаа бүгдийг дэлгэцэд харуулах болно.

Өөр олон ашигтай тохируулгууд байдаг бөгөөд lastcomm(1), acct(5) болон sa(8) гарын авлагын хуудаснуудад тайлбарласан байдаг.

# Chapter 16. Jails буюу Шоронгууд

## 16.1. Ерөнхий агуулга

Энэ бүлэг нь FreeBSD-ийн шоронгууд гэж юу болох, тэдгээрийг хэрхэн ашиглах талаар тайлбарлах болно. Шоронгууд буюу заримдаа *chroot орчнуудын* өргөжүүлсэн орлуулалт гэгддэг энэ боломж нь системийн администраторуудад зориулагдсан маш хүчтэй хэрэгсэл боловч тэдгээрийн үндсэн хэрэглээ нь илүү дэвшилтэт хэрэглэгчдэд бас үр ашигтай байдаг.

Шоронгууд нь хүчирхэг хэрэгсэл боловч аюулгүй байдлын хувьд мундаг биш юм. Шоронд хийсэн процесс өөрөө оргох боломжгүй ч шоронгийн гаднах зөвшөөрөлгүй хэрэглэгч шоронд байгаа зөвшөөрөлтэй хэрэглэгчтэй холбогдож улмаар хост орчинд эрх дээшлүүлэх хэд хэдэн арга байдаг гэдгийг мэдэх нь чухал юм.

Эдгээр ихэнх халдлагуудыг хост орчин дахь зөвшөөрөлгүй хэрэглэгчдийн хувьд шоронгийн root руу хандах боломжгүй болгосноор шийдэж болох юм. Гэхдээ ерөнхийдөө итгэл олгогдоогүй бөгөөд шоронд зөвшөөрөлтэй хандах хэрэглэгчдэд хост орчинд хандах эрх өгөх ёсгүй юм.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Шорон гэж юу болох, FreeBSD-ийн суулгалтуудад ямар зорилгоор ашиглагдаж болох талаар.
- Шоронг хэрхэн бүтээх, эхлүүлэх, болон зогсоох талаар.
- Шоронгийн гадна болон дотор талаас хийгдэж болох удирдлагын үндсүүд.

Шоронгийн тухай ашигтай мэдээллийн өөр бусад эхүүдийг дурдвал:

- jail(8) гарын авлагын хуудас. Энэ нь `jail` хэрэгслийн бүрэн гүйцэд авлага юм - `jail` нь FreeBSD дээр FreeBSD шоронгуудыг эхлүүлэх, зогсоох, болон хянахад ашиглагдаж болох удирдлагын хэрэгсэл юм.
- Захидлын жагсаалтууд болон тэдгээрийн архивууд. FreeBSD ерөнхий асуултууд захидлын жагсаалт болон бусад захидлын жагсаалтуудын архивууд нь FreeBSD жагсаалтын сервер дээр байрладаг бөгөөд шоронгуудын талаар маш баялаг материалуудыг агуулсан байдаг. Архивуудаас хайх юм уу эсвэл freebsd-questions захидлын жагсаалт уруу шинэ асуултаа илгээх нь үргэлж сонирхолтой байдаг.

## 16.2. Шоронгуудтай холбоотой ойлголтууд

Шоронгуудтай холбоотой FreeBSD системийн хэсгүүд, тэдгээрийн дотоод хэсгүүд болон FreeBSD-ийн бусад хэсэгтэй хэрхэн харилцдаг арга замыг илүүтэй ойлгохыг хөнгөвчлөхийн тулд энэ бүлэгт дараах ойлголтуудыг ашиглах болно:

**chroot(8) (тушаал)**

Процесс болон түүний бүх үр удмуудын root санг өөрчилдөг FreeBSD-ийн chroot(2)

системийн дуудлагыг ашигладаг хэрэгсэл.

**chroot(2) (орчин)**

"chroot"-д ажиллаж байгаа процессуудын орчин. Үүнд харагдаж байгаа файлын системийн хэсэг, байгаа хэрэглэгч болон бүлэг, сүлжээний интерфэйсүүд болон бусад IPC арга замууд гэх мэт эх үүсвэрүүд ордог.

**jail(8) (тушаал)**

Шоронгийн орчин дотор процессуудыг ажиллуулах боломжийг олгох системийн удирдлагын хэрэгсэл.

**хост (систем, процесс, хэрэглэгч, гэх мэт.)**

Шоронгийн системийн хяналтын систем. Хост систем нь байгаа бүх тоног төхөөрөмжийн эх үүсвэрүүдэд хандах боломжтой байдаг бөгөөд шоронгийн орчны болон түүний гаднах процессуудыг хянаж чаддаг. Хост системийн шоронгоос ялгарах нэг чухал ялгаа нь шорон доторх супер хэрэглэгчийн процессуудад хамаарах хязгаарлалтууд хост системийн процессуудын хувьд үйлчилдэггүй явдал юм.

**хост хийгдсэн (систем, процесс, хэрэглэгч, гэх мэт.)**

FreeBSD шоронгоор эх үүсвэрүүдэд ханддаг хандалт нь хязгаарлагддаг процесс, хэрэглэгч эсвэл бусад зүйлс.

# 16.3. Танилцуулга

Системийн удирдлага нь хэцүү, самууруулмаар ажил болохоор администраторын амьдралыг хялбар болгох үүднээс олон хүчирхэг хэрэгслүүд хийгдэж хөгжүүлэгдсэн байдаг. Эдгээр хэрэгслүүд нь системийг суулгах, тохируулах, болон арчлахад нэгэн төрлийн өргөтгөлүүдийг ихэвчлэн хангаж өгдөг. Администраторуудын хийх ёстой эдгээр ажлуудын нэг хэсэг нь системийн аюулгүй байдлыг зөв тохируулах явдал юм. Ингэснээр аюулгүй байдлын зөрчлүүдгүйгээр систем өөрийн жинхэнэ зорилгоороо үйлчлэх болно.

FreeBSD системийн аюулгүй байдлыг сайжруулахад ашиглагдаж болох хэрэгслүүдийн нэг нь *jails* буюу *шоронгууд* юм. Шоронгуудыг FreeBSD 4.X дээр Poul-Henning Kamp <phk@FreeBSD.org> анх танилцуулсан юм. Гэхдээ тэдгээрийг хүчирхэг, уян хатан дэд систем болгохын тулд FreeBSD 5.X дээр илүү ихээр сайжруулсан билээ. Тэдгээрийн ашигтай тал, ажиллагаа болон найдвартай байдлыг өргөжүүлэн тэдгээрийн хөгжүүлэлт үргэлжилсэн хэвээр болой.

## 16.3.1. Шорон гэж юу вэ

BSD-тэй төстэй үйлдлийн системүүд нь 4.2BSD-ийн үеэс эхлэн chroot(2) боломжтой болсон билээ. chroot(8) хэрэгсэл нь процессуудын олонлогийн root санг өөрчлөхөд ашиглагдаж аюулгүй орчин үүсгэн системийн бусад хэсгээс тэдгээрийг тусгаарладаг. chroot хийгдсэн орчинд үүсгэгдсэн процессууд нь өөрийн орчноос гаднах файлууд болон эх үүсвэрүүдэд хандаж чаддаггүй. Энэ шалтгаанаар chroot хийгдсэн орчинд ажиллаж байгаа үйлчилгээг эвдэх нь халдагчид бүхэл системийг эвдэх боломжийг олгох ёсгүй юм. chroot(8) хэрэгсэл нь маш их уян хатан чанар эсвэл төвөгтэй, дэвшилтэт боломжуудыг шаарддаггүй хялбар ажлуудад сайн байдаг. Мөн chroot ойлголтын эхлэлээс эхлээд л chroot хийгдсэн орчноос

зуттах олон арга замууд олдсон бөгөөд хэдийгээр тэдгээр нь FreeBSD цөмийн орчин үеийн хувилбаруудад засагдсан боловч chroot(2) нь үйлчилгээнүүдийг аюулгүй болгоход зориулагдсан туйлын шийдэл биш нь тодорхой байсан юм. Үүнтэй холбоотой шинэ дэд систем хийгдэх шаардлагатай болсон байна.

Энэ нь *шоронгууд* яагаад хөгжүүлэгдсэн гол шалтгаануудын нэг юм.

Шоронгууд нь уламжлалт chroot(2) орчны ойлголтуудыг хэд хэдэн аргаар сайжруулдаг. Уламжлалт chroot(2) орчинд процессууд нь өөрийн хандаж болох файлын системийн нэг хэсэгт хязгаарлагдаж байдаг. Системийн бусад эх үүсвэрүүд (системийн хэрэглэгчид, ажиллаж байгаа процессууд, эсвэл сүлжээний дэд систем зэрэг) нь chroot хийгдсэн процессууд болон хост системийн процессуудын хооронд хуваалцан хэрэглэгддэг. Шоронгууд нь зөвхөн файлын систем уруу хандах хандалт биш бас хэрэглэгчид, FreeBSD цөмийн сүлжээний дэд систем болон бусад хэд хэдэн зүйлсүүдийг виртуалчлан энэ загварыг өргөтгөдөг байна. Шорон болгосон орчны хандалтыг тааруулахад зориулсан илүү бүрэн гүйцэд нарийн тааруулсан хяналтуудын олонлог байдаг нь Нарийн тааруулалт болон удирдлага хэсэгт тайлбарлагдсан байгаа.

Шорон дөрвөн элементээр тодорхойлогддог:

- Сангийн дэд мод - шоронгийн орж ирдэг эхлэл цэг. Шорон дотор орсны дараа процессийг энэ дэд модноос гадна зуттахыг зөвшөөрдөггүй. Анхдагч chroot(2) дизайныг зовоосон аюулгүй байдлын уламжлалт асуудлууд нь FreeBSD шоронгуудад байдаггүй.
- Хостын нэр - шорон дотор ашиглагдах хостын нэр. Шоронгууд нь сүлжээний үйлчилгээнүүдийг хост хийхэд (байрлуулах) ихэвчлэн ашиглагддаг болохоор шорон бүрийн хувьд тодорхойлсон нэртэй байх нь системийн администраторт ихээхэн тус болж чадах юм.
- IP хаяг - энэ нь шорон бүрт өгөгдөх бөгөөд шоронгийн оршин тогтнох хугацаанд ямар ч талаараа өөрчлөгдөх ёсгүй. Шоронгийн IP хаяг нь ихэвчлэн байгаа сүлжээний интерфэйсийн alias хаяг байх боловч заавал тийм байх шаардлагагүй юм.
- Тушаал - шорон дотор ажиллах програм/тушаалын зам. Энэ нь шоронгийн орчны root сантай харьцангуй байх бөгөөд шоронгийн тусгай очны төрлөөс хамаараад асар өөр өөр байж болох юм.

Эдгээрээс гадна шоронгууд нь өөрийн гэсэн хэрэглэгчид болон өөрийн `root` хэрэглэгчтэй байж болдог. Мэдээжийн хэрэг `root` хэрэглэгчийн хүч чадал шоронгийн орчин дотор хязгаарлагддаг бөгөөд хост системийн үүднээс авч үзвэл шоронгийн `root` хэрэглэгч нь бүхнийг чадагч хэрэглэгч биш юм. Мөн шоронгийн `root` хэрэглэгчид өөрийнх нь харгалзах jail(8) орчноос гадна осолтой үйлдлүүдийг систем дээр хийлгэхийг зөвшөөрдөггүй. `root` хэрэглэгчийн боломжууд болон хязгаарлалтуудын тухай дэлгэрэнгүй мэдээллийг Нарийн тааруулалт болон удирдлага хэсэгт доор хэлэлцэх болно.

# 16.4. Шоронг үүсгэж хянах нь

Зарим администраторууд шоронг дараах хоёр төрөлд хуваадаг: эдгээр нь жинхэнэ FreeBSD системтэй адил төстэй "бүрэн" шоронгууд болон нэг програм юм уу эсвэл үйлчилгээнд зориулагдсан, магадгүй зөвшөөрлүүдтэй ажиллах "үйлчилгээ" шоронгууд юм. Энэ нь зөвхөн

ухагдахууны хуваагдал бөгөөд шоронг бүтээх процесс үүнд хамаагүй юм. jail(8) гарын авлагын хуудас шоронг бүтээх аргачлалын талаар маш тодорхой зааварласан буй:

```
# setenv D /here/is/the/jail
# mkdir -p $D ①
# cd /usr/src
# make buildworld ②
# make installworld DESTDIR=$D ③
# make distribution DESTDIR=$D ④
# mount -t devfs devfs $D/dev ⑤
```

① Шоронгийн байрлалыг сонгох нь хамгийн шилдэг эхлэх цэг юм. Энэ нь шорон физикээр өөрийн хостын файлын систем дотор байрлах байрлал юм. Сайн сонголт нь /usr/jail/jailname байж болох бөгөөд энд байгаа *jailname* нь шоронг таниулж байгаа хостын нэр юм. /usr/ файлын систем нь шоронгийн файлын системийн хувьд ихэвчлэн хангалттай зайтай байдаг. Үндсэндээ "бүрэн" шоронгуудын хувьд энэ шоронгийн файлын систем нь үндсэн FreeBSD системийн андагч суулгацад байдаг бүх файлуудын хуулбар байдаг.

② Хэрэв та өөрийн хэрэглэгчийн талбарыг `make world` эсвэл `make buildworld` тушаалаар аль хэдийн бүтээсэн бол та энэ алхмыг алгасаж өөрийн хэрэглэгчийн талбарыг шинэ шорон руу суулгаж болно.

③ Энэ тушаал нь шоронгийн физик байрлал болгон сонгосон сангийн дэд модыг файлын систем дээр шаардлагатай хоёртын файлууд, сангууд, гарын авлагын хуудаснууд гэх зэргүүдийг тараан байрлуулах болно.

④ make тушаалд зориулагдсан `distribution` тохируулга нь бүх шаардлагатай тохиргооны файлыг суулгана, өөрөөр хэлбэл энэ нь /usr/src/etc/ сангийн бүх суулгаж болох файлуудыг шоронгийн орчны /etc сан болох $D/etc/ руу хуулдаг.

⑤ Шорон дотор devfs(8) файлын системийг холбох шаардлагагүй. Нөгөө талаас авч үзвэл дурын, бараг бүх програм өөрийн зорилгоосоо хамааран хамгийн багаар бодоход ядаж ганц төхөөрөмжид хандах шаардлагатай байдаг. Шорон дотроос төхөөрөмжид хандах хандалтыг хянах нь маш чухал байдаг. Учир нь буруу тохируулгууд халдагчид шорон дотор муухай зүйлс хийх боломжийг олгож болох юм. devfs(8) дээрх хяналтыг devfs(8) болон devfs.conf(5) гарын авлагын хуудаснуудад тайлбарласан дүрмийн олонлогуудаар удирддаг.

Шорон суулгагдсаны дараа jail(8) хэрэгсэл ашиглан түүнийг эхлүүлж болно. jail(8) хэрэгсэл дөрвөн зайлшгүй шаардлагатай нэмэлт өгөгдлийг авдаг бөгөөд эдгээр нь Шорон гэж юу вэ хэсэгт тайлбарлагдсан байгаа болно. Өөр бусад нэмэлт өгөгдлүүдийг бас зааж өгч болох бөгөөд өөрөөр хэлбэл шорон хийгдсэн процессийг тухайн нэг хэрэглэгчийн итгэмжлэлүүдтэй ажиллуулж болох юм. `command`-н нэмэлт өгөгдөл нь шоронгийн төрлөөс хамаарна; *виртуал системийн* хувьд /etc/rc нь боломжийн сонголт байна. Энэ нь жинхэнэ FreeBSD системийн эхлүүлэх дарааллыг хуулбарлах учраас тэр юм. *Үйлчилгээ* шоронгийн хувьд шорон дотор ажиллах үйлчилгээ эсвэл програмаас энэ нь хамаарна.

Шоронгууд нь ихэвчлэн ачаалах үед эхлүүлэгддэг бөгөөд FreeBSD rc арга зам нь үүнийг хийх хялбар аргаар хангадаг.

1. Ачаалах үед эхлэхээр идэвхжүүлэгдсэн шоронгуудын жагсаалтыг rc.conf(5) файлд нэмэх ёстой:

```
jail_enable="YES"   # Set to NO to disable starting of any jails
jail_list="www"     # Space separated list of names of jails
```

`jail_list` хувьсагчид өгөгдсөн байгаа шоронгийн нэрс зөвхөн үсэг, тооноос бүрдсэн байх ёстой.

2. Шорон бүрийг тайлбарласан rc.conf(5) тохируулгуудын бүлэг `jail_list`-д жагсаагдсан шорон бүрийн хувьд доор дурдсаныг нэмэх ёстой:

```
jail_www_rootdir="/usr/jail/www"     # jail's root directory
jail_www_hostname="www.example.org"   # jail's hostname
jail_www_ip="192.168.0.10"          # jail's IP address
jail_www_devfs_enable="YES"          # mount devfs in the jail
jail_www_devfs_ruleset="www_ruleset" # devfs ruleset to apply to jail
```

rc.conf(5)-д тохируулагдсан шоронгуудын анхдагч эхлүүлэлт нь шоронг бүрэн виртуал систем гэж тооцдог шоронгийн /etc/rc скриптийг ажиллуулах болно. Үйлчилгээний шоронгуудын хувьд `jail`jailname`exec_start` тохируулгыг зохистойгоор тохируулан шоронгийн анхдагч эхлүүлэх тушаалыг өөрчлөх ёстой.

Тохируулгуудын бүрэн жагсаалтыг rc.conf(5) гарын авлагын хуудаснаас үзнэ үү.

Шоронд зориулагдсан оруулга rc.conf файлд байгаа тохиолдолд service(8) шоронг гараар эхлүүлэх эсвэл зогсооход ашиглагдаж болох юм:

```
# service jail start www
# service jail stop www
```

Одоогоор jail(8)-г зогсоох цэвэр зам байхгүй байгаа. Цэвэр системийн зогсолтыг хийх тушаалуудыг шорон дотор ашиглах боломжгүй байдаг болохоор тэр юм. Шоронг зогсоох хамгийн шилдэг арга бол дараах тушаалыг шорон дотроос ажиллуулах эсвэл шоронгийн гадна jexec(8) хэрэгслийг ашиглах явдал юм:

```
# sh /etc/rc.shutdown
```

Үүний талаар дэлгэрэнгүй мэдээллийг jail(8) гарын авлагын хуудаснаас олж болно.

# 16.5. Нарийн тааруулалт болон удирдлага

Аль ч шоронд зориулж тохируулж болох хэд хэдэн тохируулгууд байдаг бөгөөд өндөр түвшний програмуудыг хийхийн тулд хост FreeBSD системийг шоронгуудтай цуг хослуулах төрөл бүрийн аргууд бас байдаг. Энэ хэсэг нь дараах зүйлсийг үзүүлнэ:

- Ажиллагаа болон шоронгийн суулгалтаар хийгдсэн аюулгүй байдлын хязгаарлалтуудыг тааруулахад зориулагдсан зарим тохируулгууд.
- FreeBSD-ийн портын цуглуулгад байх, шорон дээр суурилсан шийдлүүдийг хийхэд ашиглагдаж болох шорон удирдах зарим нэг өндөр түвшний програмууд.

## 16.5.1. FreeBSD дээр шорон тааруулах системийн хэрэгслүүд

Шоронгийн тохиргооны нарийн сайн тааруулалтыг sysctl(8) хувьсагчуудыг тохируулснаар ихэвчлэн хийдэг. Бүх хамаатай тохируулгуудыг зохион байгуулах үндэс болон sysctl-ийн тусгай дэд мод байдаг: энэ нь FreeBSD цөмийн тохируулгуудын `security.jail.*` шатлал юм. Энд шоронтой холбоотой гол sysctl-уудын жагсаалтыг тэдгээрийн анхдагч утгуудтайгаар харуулав. Нэрс нь өөрийгөө тайлбарласан байгаа, гэхдээ тэдгээрийн талаар илүү мэдээллийг jail(8) болон sysctl(8) гарын авлагын хуудаснуудаас лавлана уу.

- `security.jail.set_hostname_allowed: 1`
- `security.jail.socket_unixiproute_only: 1`
- `security.jail.sysvipc_allowed: 0`
- `security.jail.enforce_statfs: 2`
- `security.jail.allow_raw_sockets: 0`
- `security.jail.chflags_allowed: 0`
- `security.jail.jailed: 0`

`root` хэрэглэгчид анхдагчаар ноогдуулсан хязгаарлалтуудын заримыг нэмэх эсвэл хасахын тулд эдгээр хувьсагчуудыг *хост системийн* администратор ашиглаж болно. Зарим нэг хязгаарлалтуудыг хасаж болохгүйг тэмдэглэе. `root` хэрэглэгчид jail(8) дотор файлын системүүдийг холбох эсвэл салгахыг зөвшөөрдөггүй. Шорон доторх `root` хэрэглэгч devfs(8) дүрмийн олонлогуудыг дуудах эсвэл буцааж болиулах, галт ханын дүрмүүдийг тохируулах, эсвэл цөмийн `securelevel` хувьсагчийг тохируулах зэрэг цөм дэх өгөгдлийн өөрчлөлтүүдийг шаарддаг өөр олон бусад удирдлагын ажлуудыг хийж чадахгүй байж болох юм.

FreeBSD-ийн үндсэн систем нь идэвхтэй шоронгуудын тухай мэдээллийг үзүүлэх болон удирдлагын тушаалуудыг ажиллуулахын тулд шоронд залгагдаж болох хялбар хэрэгслүүдийн цуглуулгыг агуулдаг. jls(8) болон jexec(8) тушаалууд нь FreeBSD-ийн үндсэн системийн хэсэг бөгөөд дараах хялбар ажлуудыг хийж гүйцэтгэхэд ашиглагдаж болно:

- Идэвхтэй байгаа шоронгуудын жагсаалт болон тэдгээрийн харгалзах шорон танигч (JID), IP хаяг, хостын нэр болон замыг үзүүлнэ.
- Өөрийнх нь хост системээс ажиллаж байгаа шоронд залгагдаж шорон дотор тушаал ажиллуулах юм уу эсвэл шоронгийн удирдлагын ажлуудыг шорон дотор ажиллуулна. `root` хэрэглэгч шоронг цэвэрхэн зогсоож унтраахыг хүсэх үед энэ нь ялангуяа ашигтай

байдаг. Шорон дотор удирдлага хийхийн тулд түүн дотор бүрхүүл эхлүүлэхэд jexec(8) хэрэгсэл бас ашиглагдаж болдог; жишээ нь:

```
# jexec 1 tcsh
```

### 16.5.2. FreeBSD-ийн портын цуглуулга дахь өндөр түвшний удирдлагын хэрэгслүүд

Шорон удирдлагад зориулагдсан гуравдагч талуудын олон хэрэгслүүдийн дундаас хамгийн бүрэн гүйцэд, ашигтай нь sysutils/jailutils юм. Энэ нь jail(8)-ийн удирдлагад хувь нэмэр болсон жижиг програмуудын цуглуулга юм. Дэлгэрэнгүй мэдээллийн талаар түүний вэб хуудсанд хандана уу.

## 16.6. Шоронгийн хэрэглээ

### 16.6.1. Үйлчилгээ шоронгууд

Энэхүү хэсэг нь Simon L. B. Nielsen <simon@FreeBSD.org> хөгжүүлэгчийн http://simon.nitro.dk/service-jails.html хуудас болон Кен Том locals@gmail.com-ийн бичсэн шинэчилсэн нийтлэл дээр гаргасан санаа дээр тулгуурласан юм. Энэ хэсэг нь jail(8) боломжийг ашиглан аюулгүй байдлын нэмэлт давхарга бүхий FreeBSD систем хэрхэн тохируулах талаар тайлбарлах болно. Өгөгдсөн систем нь ядаж RELENG_6_0 бөгөөд энэ бүлгийн өмнө дурдсан мэдээллийг уншигч авхай сайн ойлгосон гэж тооцдог.

#### 16.6.1.1. Шийдэл

Шоронгуудын гол асуудлуудын нэг нь тэдгээрийн шинэчлэлтийн процессийн удирдлага юм. Шорон бүр нь шинэчлэгдэх болгондоо дахин бүр эхнээсээ бүтээгдэх хэрэгтэй болдог учраас тэр юм. Нэг шоронгийн хувьд энэ нь ихэвчлэн асуудал болдоггүй, шинэчлэлтийн процесс их хялбар байдаг боловч олон шоронгууд үүсгэсэн бол их хугацаа шаардсан, төвөгтэй ажиллагаа байдаг.

Энэхүү тохиргоо нь FreeBSD-ийн маш сайн туршлага болон түүний боломжуудын хэрэглээг шаарддаг. Хэрэв доор үзүүлсэн алхмууд нь хэтэрхий төвөгтэй санагдвал FreeBSD шоронгуудыг удирдах илүү хялбар боломжийг олгодог бөгөөд энэ тохиргооных шиг төвөгтэй биш sysutils/qjail эсвэл sysutils/ezjail зэрэг хялбар системийг үзэхийг зөвлөж байна.

Гол санаа нь иймэрхүү асуудлуудыг шоронгууд хооронд аль болох ихээр аюулгүй аргаар хуваалцах замаар шийдэх явдал юм - шинэчлэлт хялбар байхаар зөвхөн уншигдах mount_nullfs(8) холболтуудыг ашиглах болон ганц үйлчилгээнүүдийг тусдаа шоронд хийх нь илүүтэй болох юм. Мөн энэ нь шоронгуудыг нэмэх эсвэл устгах болон тэдгээрийг шинэчлэх хялбар боломжийг олгодог юм.

Ийм зорилгоор ашиглаж болох үйлчилгээнүүдийг дурдвал: HTTP сервер, DNS сервер, SMTP сервер гэх мэт байж болох юм.

Энэ хэсэгт тайлбарласан тохиргооны зорилгуудыг дурдвал:

- Шорongийн хялбар, ойлгоход амархан бүтцийг үүсгэх. Энэ нь шорон бүрийн хувьд болон тэдгээр дээр бүрэн хэмжээний installworld үйлдлийг *ажиллуулахгүй* байх гэсэн үг юм.
- Шинэ шорongууд нэмэх эсвэл байгааг нь устгах процессийг хялбар болгох.
- Байгаа шорongуудыг шинэчлэх эсвэл сайжруулах процессийг хялбар болгох.
- Өөрчлөн тохируулсан FreeBSD салбарыг ажиллуулах боломжтой болгох.
- Нэвтрэн орох, эвдлэн орох боломжийг аль болох ихээр багасгаж аюулгүй байдлын хувьд паранойд байх.
- Зай болон inode-уудыг аль болох ихээр хэмнэх.

Урьд нь дурдагдсаны адил энэ шийдэл нь шорон бүрт зөвхөн уншигдахаар (nullfs гэгддэг) холбогдох ганц мастер загвар болон шорон бүрийн хувьд нэг уншигдах, бичигдэх төхөөрөмжтэй байх бүтэц дээр үндсэндээ тулгуурласан юм. Төхөөрөмж нь тусдаа физик диск, хуваалт, эсвэл vnode дээр тулгуурласан md(4) төхөөрөмж байж болох юм. Энэ жишээн дээр бид уншигдах, бичигдэх nullfs холболтуудыг ашиглах болно.

Файлын системийн дүр зураг доор дурдсан хэсэгт тайлбарласан буй:

- Шорон бүр /home/j сангийн доор холбогдох болно.
- /home/j/mroot нь шорон бүрийн хувьд загвар ба бүх шорongуудын хувьд зөвхөн уншигдах хуваалт юм.
- /home/j сангийн доор шорон бүрийн хувьд хоосон сан үүсгэгдэнэ.
- Шорон бүр системийн уншигдах, бичигдэх хэсэг уруу холбогдох /s сантай байна.
- Шорон бүр /home/j/skel дээр тулгуурласан өөрийн уншигдах, бичигдэх системтэй байх болно.
- Шорongийн талбар бүр (шорон бүрийн уншигдах, бичигдэх хэсэг) /home/js-д үүсгэгдэх болно.

Энэ нь шорongууд /home хуваалтын доор үндэслэсэн гэж үзнэ. Үүнийг мэдээж өөрчилж болох боловч ингэх тохиолдолд доор дурдсан жишээ бүрийн хувьд өөрчлөгдөх ёстой болно.

#### 16.6.1.2. Загвар үүсгэх нь

Энэ хэсэг нь шоронд зориулагдан ашиглагдах, зөвхөн уншигдах хэсэг болох мастер загварыг үүсгэхэд хэрэгтэй алхмуудыг тайлбарлах болно.

FreeBSD системийг сүүлийн -RELEASE салбар уруу шинэчлэх нь үргэлж зөв санаа байдаг. Үүнийг хийхийн тулд гарын авлагын бүлгээс лавлах хэрэгтэй. Шинэчлэл хийх шаардлагагүй бол гүйцэтгэлийг гүйцээхийн тулд buildworld хийх шаардлагатай. Мөн sysutils/cpdup багц хэрэгтэй. FreeBSD-ийн портын цуглуулгыг татаж авахдаа бид portsnap(8) хэрэгслийг ашиглах болно. Эхлэн суралцагчид гарын авлагын Portsnap бүлгийг унших нь зүйтэй юм.

1. Эхлээд бидний шоронгуудад зориулсан FreeBSD-ийн хоёртын файлуудыг агуулах зөвхөн уншигдах файлын системийн сангийн бүтцийг үүсгэх хэрэгтэй бөгөөд дараа нь FreeBSD-ийн эх модны сан уруу сангаа сольж ороод зөвхөн уншигдах файлын системийг шоронгийн загвар уруу суулгах хэрэгтэй:

```
# mkdir /home/j /home/j/mroot
# cd /usr/src
# make installworld DESTDIR=/home/j/mroot
```

2. Дараа нь шоронгуудад зориулж FreeBSD-ийн портын цуглуулга болон mergemaster-т шаардлагатай, FreeBSD-ийн эх модыг бэлдэх хэрэгтэй:

```
# cd /home/j/mroot
# mkdir usr/ports
# portsnap -p /home/j/mroot/usr/ports fetch extract
# cpdup /usr/src /home/j/mroot/usr/src
```

3. Системийн уншигдах, бичигдэх хэсэгт зориулж араг ясыг үүсгэх хэрэгтэй:

```
# mkdir /home/j/skel /home/j/skel/home /home/j/skel/usr-X11R6
/home/j/skel/distfiles
# mv etc /home/j/skel
# mv usr/local /home/j/skel/usr-local
# mv tmp /home/j/skel
# mv var /home/j/skel
# mv root /home/j/skel
```

4. Байхгүй байгаа тохиргооны файлуудыг суулгахын тулд mergemaster-г ашиглах хэрэгтэй. Дараа нь mergemaster-ийн үүсгэсэн илүү сангуудыг арилгах хэрэгтэй:

```
# mergemaster -t /home/j/skel/var/tmp/temproot -D /home/j/skel -i
# cd /home/j/skel
# rm -R bin boot lib libexec mnt proc rescue sbin sys usr dev
```

5. Одоо уншигдах, бичигдэх файлын системийг зөвхөн уншигдах файлын систем уруу заасан симболын холбоос үүсгэх хэрэгтэй. Симболын холбоосууд нь зөв s/ байрлалуудад үүсгэгдсэн эсэхийг шалгаарай. Жинхэнэ сангууд юм уу эсвэл сангуудын үүсгэлт буруу байрлалуудад хийгдсэн бол суулгалт амжилтгүй болоход хүргэнэ.

```
# cd /home/j/mroot
# mkdir s
# ln -s s/etc etc
# ln -s s/home home
```

```
# ln -s s/root root
# ln -s ../s/usr-local usr/local
# ln -s ../s/usr-X11R6 usr/X11R6
# ln -s ../../s/distfiles usr/ports/distfiles
# ln -s s/tmp tmp
# ln -s s/var var
```

6. Сүүлийн шатанд доор дурдсан агуулга бүхий ерөнхий /home/j/skel/etc/make.conf файлыг үүсгэх хэрэгтэй:

```
WRKDIRPREFIX?=  /s/portbuild
```

WRKDIRPREFIX-г ийм байдлаар тохируулах нь шорон бүрт FreeBSD-ийн портуудыг эмхэтгэх боломжтой болгох юм. Портуудын сан нь зөвхөн уншигдах системийн хэсэг гэдгийг санаарай. WRKDIRPREFIX-д зориулсан өөр зам нь шорон бүрийн уншигдах, бичигдэх хэсэгт бүтээлтүүдийг хийх боломжийг олгох юм.

#### 16.6.1.3. Шорон үүсгэх нь

Одоо бид бүрэн гүйцэд FreeBSD-ийн шоронгийн загвартай болсон болохоор /etc/rc.conf файлд бид шоронгуудыг суулган тохируулах боломжтой болно. Энэ жишээ нь "NS", "MAIL" болон "WWW" гэсэн 3 шоронгийн үүсгэлтийг харуулж байна.

1. Доор дурдсан мөрүүдийг /etc/fstab файлд нэмэх хэрэгтэй. Ингэснээр шоронгуудад зориулсан зөвхөн уншигдах загвар болон уншигдах, бичигдэх зай тохирох шоронгуудад ашиглах боломжтой болох юм:

```
/home/j/mroot   /home/j/ns     nullfs  ro  0   0
/home/j/mroot   /home/j/mail   nullfs  ro  0   0
/home/j/mroot   /home/j/www    nullfs  ro  0   0
/home/js/ns     /home/j/ns/s   nullfs  rw  0   0
/home/js/mail   /home/j/mail/s nullfs  rw  0   0
/home/js/www    /home/j/www/s  nullfs  rw  0   0
```

0 pass буюу өнгөрөх дугаараар тэмдэглэгдсэн хуваалтууд нь ачаалах үед fsck(8) хэрэгслээр шалгагддаггүй бөгөөд 0 dump дугаараар тэмдэглэгдсэн хуваалтууд нь dump(8) хэрэгслээр нөөцлөгддөггүй. Бид fsck хэрэгслээр nullfs холболтуудыг шалгах эсвэл dump хэрэгслээр шоронгуудын зөвхөн уншигдах nullfs холболтуудыг нөөцлөхийг хүсэхгүй байгаа билээ. Дээр дурдсан fstab оруулга бүрийн сүүлийн хоёр багана "0 0" гэж тэмдэглэгдсэн учир нь энэ юм.

2. Шоронгуудыг /etc/rc.conf-д тохируулах хэрэгтэй:

```
jail_enable="YES"
```

```
jail_set_hostname_allow="NO"
jail_list="ns mail www"
jail_ns_hostname="ns.example.org"
jail_ns_ip="192.168.3.17"
jail_ns_rootdir="/usr/home/j/ns"
jail_ns_devfs_enable="YES"
jail_mail_hostname="mail.example.org"
jail_mail_ip="192.168.3.18"
jail_mail_rootdir="/usr/home/j/mail"
jail_mail_devfs_enable="YES"
jail_www_hostname="www.example.org"
jail_www_ip="62.123.43.14"
jail_www_rootdir="/usr/home/j/www"
jail_www_devfs_enable="YES"
```

`jail`name`rootdir` хувьсагчийн утга /home-ийн оронд /usr/home гэсэн шалтгаан нь FreeBSD-ийн үндсэн суулгац дээр /home сангийн физик зам нь /usr/home гэж байдаг оршиж байгаа юм. `jail`name`rootdir` хувьсагчийн утга симбол холбоос бүхий зам байхаар тохируулагдсан байх *ёсгүй* бөгөөд хэрэв ингэвэл шоронгууд ажиллаж эхлэхгүй байх болно. Энэ хувьсагчийн утгад тавьж болох утгыг олохдоо realpath(1) -г ашиглах хэрэгтэй. Дэлгэрэнгүй мэдээллийг FreeBSD-SA-07:01.jail Аюулгүй байдлын Зөвлөгөөнөөс үзнэ үү.

3. Шорон бүрийн зөвхөн уншигдах файлын системд зориулсан, шаардлагатай холбох цэгүүдийг үүсгэнэ:

```
# mkdir /home/j/ns /home/j/mail /home/j/www
```

4. Шорон бүрт уншигдах, бичигдэх загварыг суулгах хэрэгтэй. sysutils/cpdup хэрэгслийн хэрэглээг энд тэмдэглэх нь зүйтэй юм. Энэ нь сан бүрийн зөв хуулбарыг хийхэд тусалдаг:

```
# mkdir /home/js
# cpdup /home/j/skel /home/js/ns
# cpdup /home/j/skel /home/js/mail
# cpdup /home/j/skel /home/js/www
```

5. Энэ үед шоронгууд нь бүтээгдэж ажиллахад бэлтгэгдсэн байна. Эхлээд шорон бүрийн хувьд шаардлагатай файлын системийг холбож дараа нь тэдгээрийг jail rc скрипт ашиглан эхлүүлэх хэрэгтэй:

```
# mount -a
# service jail start
```

Шоронгууд нь одоо ажиллаж байх ёстой. Тэдгээрийг зөв эхэлсэн эсэхийг шалгахын тулд jls(8) тушаалыг ашиглана. Үүний гаралт доор дурдсантай төстэй байх ёстой:

```
# jls
   JID  IP Address      Hostname                      Path
     3  192.168.3.17    ns.example.org                /home/j/ns
     2  192.168.3.18    mail.example.org              /home/j/mail
     1  62.123.43.14    www.example.org               /home/j/www
```

Энэ үед шорон бүр рүү нэвтэрч, шинэ хэрэглэгчид нэмэх эсвэл дэмонуудыг тохируулах боломжтой болсон байх ёстой. `JID` багана нь ажиллаж байгаа шорон бүрийн шорон таниулах дугаарыг илэрхийлдэг. `JID` нь 3 бүхий шорон дотор удирдлагын ажлуудыг гүйцэтгэхийн тулд дараах тушаалыг ашиглах хэрэгтэй:

```
# jexec 3 tcsh
```

#### 16.6.1.4. Шинэчлэх нь

Аюулгүй байдлын асуудлаас болоод эсвэл одоо байгаа шоронгуудад ашигтай шинэ боломжууд хийгдсэнээс болоод системээ FreeBSD-ийн шинэ хувилбар уруу шинэчлэх шаардлага заримдаа гардаг. Энэ тохиргооны дизайн нь байгаа шоронгуудыг хялбар аргаар шинэчлэх боломжийг олгодог. Мөн шоронгуудыг сүүлийн минутанд зогсоодог болохоор энэ нь тэдгээрийн зогсох хугацааг багасгадаг. Бас ямар нэг асуудал гарахад энэ нь хуучин хувилбар уруугаа шилжих боломжийг олгодог.

1. Эхний алхам нь хост системийг журмын дагуу шинэчлэх явдал юм. Дараа шинэ, түр зуурын, зөвхөн уншигдах загварыг /home/j/mroot2-д үүсгэх хэрэгтэй.

   ```
   # mkdir /home/j/mroot2
   # cd /usr/src
   # make installworld DESTDIR=/home/j/mroot2
   # cd /home/j/mroot2
   # cpdup /usr/src usr/src
   # mkdir s
   ```

   `installworld` ажиллахдаа цөөн хэрэггүй сангуудыг үүсгэдэг бөгөөд эдгээрийг устгах хэрэгтэй:

   ```
   # chflags -R 0 var
   # rm -R etc var root usr/local tmp
   ```

2. Мастер файлын системд зориулж уншигдах, бичигдэх симболын холбоосуудыг дахин үүсгэх хэрэгтэй:

```
# ln -s s/etc etc
# ln -s s/root root
# ln -s s/home home
# ln -s ../s/usr-local usr/local
# ln -s ../s/usr-X11R6 usr/X11R6
# ln -s s/tmp tmp
# ln -s s/var var
```

3. Шоронгуудыг зогсоох зөв үе нь одоо байна:

```
# service jail stop
```

4. Эх файлын системүүдийг салгах хэрэгтэй:

```
# umount /home/j/ns/s
# umount /home/j/ns
# umount /home/j/mail/s
# umount /home/j/mail
# umount /home/j/www/s
# umount /home/j/www
```

Уншигдах, бичигдэх системүүд нь зөвхөн уншигдах системд залгагдсан (/s) бөгөөд эхлээд салгагдах ёстой.

5. Хуучин зөвхөн уншигдах файлын системийг шилжүүлж шинээр сольно. Ямар нэг юм болохоо байхад энэ нь хуучин, зөвхөн уншигдах файлын системийн нөөц болон архив маягаар ашиглагдах юм. Энд ашиглагдсан нэрлэх аргачлал нь шинэ, зөвхөн уншигдах файлын систем үүсгэгдэх үеийнхтэй тохирдог. Зай болон inode-уудыг хэмнэхийн тулд FreeBSD-ийн эх портын цуглуулгыг шинэ файлын систем уруу шилжүүлэх хэрэгтэй:

```
# cd /home/j
# mv mroot mroot.20060601
# mv mroot2 mroot
# mv mroot.20060601/usr/ports mroot/usr
```

6. Энэ үед шинэ, зөвхөн уншигдах загвар бэлэн болох бөгөөд үлдсэн цорын ганц ажил нь файлын системүүдийг дахин холбож шоронгуудыг эхлүүлэх явдал юм:

```
# mount -a
# service jail start
```

Шоронгууд зөв эхэлсэн эсэхийг шалгахын тулд jls(8)-г ашиглана. Шорон бүрт mergemaster-г

ажиллуулахаа мартуузай. Тохиргооны файлууд болон rc.d скриптүүдийг шинэчлэх хэрэгтэй болно.

# Chapter 17. Mandatory Access Control буюу Албадмал Хандалтын хяналт

## 17.1. Ерөнхий агуулга

FreeBSD 5.X нь POSIX®.1e ноорог дээр тулгуурласан TrustedBSD төслийн аюулгүй байдлын шинэ өргөтгөлүүдийг танилцуулсан. Хамгийн чухал аюулгүй байдлын шинэ арга замуудын хоёр нь файлын системийн Access Control Lists буюу Хандалтын Хяналтын Жагсаалтууд (ACL-үүд) болон Mandatory Access Control (MAC) буюу Албадмал Хандалтын Хяналт боломжууд юм. Албадмал Хандалтын Хяналт нь аюулгүй байдлын шинэ бодлогуудыг бий болгож хандалтын хяналтын модулиудыг ачаалах боломжийг олгодог. Зарим нь тухайн үйлчилгээг хатуужуулж системийн нарийн дэд олонлогуудын хамгаалалтуудыг хангадаг. Бусад нь хаяглагдсан, олон талын аюулгүй байдлыг бүх субьект болон обьектуудын хувьд хангадаг байна. Тодорхойлолтын албадмал буюу зайлшгүй шаардлагатай гэж хэлсэн хэсэг нь хяналтуудын албадлагыг администраторууд болон систем хийдэг бөгөөд discretionary access control (DAC, FreeBSD дээрх стандарт файл болон System V IPC зөвшөөрлүүд) буюу тусдаа байх хандалтын хяналтаар хийгддэг шиг хэрэглэгчээр өөрөөр нь хийлгэдэггүй гэсэн үг юм.

Энэ бүлэг Mandatory Access Control Framework (MAC Framework) буюу Албадмал Хандалтын Хяналт Тогтолцоо болон залгагдаж болох аюулгүй байдлын бодлогын модулиудын олонлогт анхаарлаа төвлөрүүлж төрөл бүрийн аюулгүй байдлын арга замуудыг идэвхжүүлэх болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Одоогоор FreeBSD-д ямар ямар аюулгүй байдлын MAC бодлогын модулиуд орсон болон тэдгээртэй холбоотой арга замуудын талаар.
- Аюулгүй байдлын MAC бодлогын модулиуд юу шийддэг болон хаяглагдсан болон хаяглагдаагүй бодлогын хоорондын ялгааны талаар.
- Системийг хэрхэн үр ашигтайгаар MAC тогтолцоог ашиглахаар тохируулах талаар.
- MAC тогтолцоонд орсон аюулгүй байдлын өөр өөр бодлогын модулиудыг хэрхэн тохируулах талаар.
- MAC тогтолцоо болон үзүүлсэн жишээнүүдийг ашиглан илүү аюулгүй орчинг хэрхэн бий болгох талаар.
- Тогтолцоо зөв хийгдсэнийг шалгахын тулд MAC тохиргоог хэрхэн тест хийх талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- UNIX® болон FreeBSD-ийн үндсүүдийг ойлгосон байх (Юниксийн үндэс).
- Цөмийн тохиргоо/эмхэтгэлийн (FreeBSD цөмийг тохируулах нь) үндсүүдтэй танилцсан байх.
- Аюулгүй байдалтай танилцаж энэ нь FreeBSD-д хэрхэн хамааралтай болохыг мэдэх (Аюулгүй байдал).

Энд байгаа мэдээллийг буруу ашиглавал системд хандаж чадахгүй болгох,

> хэрэглэгчдийн доройтол эсвэл X11-ийн хангадаг боломжуудад хандаж чадахгүйд хүргэж болох юм. Хамгийн чухал нь MAC нь системийг бүр мөсөн аюулгүй болгоно гэж найдаж болохгүй юм. MAC тогтолцоо нь байгаа аюулгүй байдлын бодлогыг зөвхөн сайжруулдаг; аюулгүй байдлын сайн практикгүй, байнгын аюулгүй байдлын шалгалтгүйгээр систем хэзээ ч бүрэн аюулгүй байж чадахгүй.
>
> Мөн энэ бүлгийн хүрээнд байгаа жишээнүүд нь зөвхөн жишээнүүд гэдгийг тэмдэглэх ёстой юм. Ялангуяа эдгээр тухайлсан тохиргоонуудыг жинхэнэ систем дээр хэрэглэхийг зөвлөдөггүй. Төрөл бүрийн аюулгүй байдлын бодлогын модулиудыг бүтээх нь ихээхэн бодолт болон тест хийхийг шаарддаг. Бүгд хэрхэн яаж ажилладгийг бүрэн ойлгоогүй хүнийн хувьд бүхэл системийг дахин үзэж олон файлууд эсвэл сангуудыг дахин тохируулахад хүргэж болох юм.

### 17.1.1. Юуг хэлэлцэхгүй вэ

Энэ бүлэг нь MAC тогтолцоотой холбоотой өргөн хүрээний аюулгүй байдлын асуудлуудыг хамардаг. Шинэ MAC аюулгүй байдлын бодлогын модулиудыг хөгжүүлэх талаар хэлэлцэхгүй болно. MAC тогтолцоонд орсон хэд хэдэн аюулгүй байдлын бодлогын модулиуд нь тусгай онцлогуудтай бөгөөд эдгээр нь тест хийх болон шинэ модуль хөгжүүлэхэд зориулагдсан юм. Эдгээрт mac_test(4), mac_stub(4) болон mac_none(4) орно. Эдгээр аюулгүй байдлын бодлогын модулиудын талаар болон тэдгээрийн хангадаг төрөл бүрийн арга замуудын талаар дэлгэрэнгүй мэдээллийг гарын авлагын хуудаснуудаас лавлана уу.

## 17.2. Энэ бүлэг дэх түлхүүр ухагдахуунууд

Энэ бүлгийг уншихаасаа өмнө хэд хэдэн түлхүүр ухагдахуунуудыг тайлбарлах ёстой. Энэ нь учирч болох ямар нэг эндүүрлийг цэгцэлж шинэ ухагдахуунууд болон мэдээллийн огцом танилцуулгаас зайлсхийх болно гэж найдаж байна.

- *compartment* буюу тасалгаа: Тасалгаа нь хэрэглэгчдэд системийн тусгай бүрэлдэхүүн хэсгүүдэд хандах хандалтыг өгдөг хуваагдах эсвэл тусгаарлагдах програмууд болон өгөгдлийн олонлог юм. Мөн тасалгаа нь ажлын групп, хэлтэс, төсөл эсвэл сэдэв зэрэг бүлэглэлийг илэрхийлдэг. Тасалгаануудыг ашиглан мэдэх хэрэгтэй аюулгүй байдлын бодлогыг хийж гүйцэтгэх боломжтой байдаг.
- *high water mark* буюу өндөр түвшин: Өндөр түвшин бодлого нь өндөр түвшний мэдээлэлд хандах зорилгоор аюулгүй байдлын түвшнүүдийг дээшлүүлэхийг зөвшөөрдөг бодлого юм. Ихэнх тохиолдолд процесс дууссаны дараа анхдагч түвшин сэргээгддэг. Одоогоор FreeBSD MAC тогтолцоо нь үүнд зориулсан бодлогогүй, гэхдээ бүрэн бүтэн байдлын үүднээс тодорхойлолт нь оржээ.
- *integrity* буюу бүрэн бүтэн байдал: Бүрэн бүтэн байдал нь түлхүүр ойлголт бөгөөд өгөгдөлд тавигдаж болох итгэмжлэлийн түвшин юм. Өгөгдлийн бүрэн бүтэн байдал дээшлэх тусам тэр өгөгдөлд итгэх чадвар бас дээшилдэг.
- *label* буюу хаяг/шошго: Хаяг/шошго нь файлууд, сангууд эсвэл систем дэх бусад зүйлсэд хамааруулж болох аюулгүй байдлын шинж чанар юм. Энэ нь итгэмжлэлийн тамга

гэгдэж болно; хаяг/шошго файлд тавигдсан бол тэр файлын аюулгүй байдлын өмчүүдийг тайлбарлах бөгөөд зөвхөн ижил аюулгүй байдлын тохиргоотой файлууд, хэрэглэгчид, эх үүсвэрүүд гэх зэргээс хандалтыг зөвшөөрөх болно. Хаяг/шошгоны утгуудын утга санаа болон тайлбар нь бодлогын тохиргооноос хамаардаг: зарим бодлогууд нь хаяг/шошгыг обьектийн бүрэн бүтэн байдал эсвэл нууцгай байдал гэж ойлгодог бол бусад бодлогууд хаяг/шошгыг хандалт хийхийн тулд дүрмүүдийг агуулахад ашиглаж болох юм.

- *level* буюу түвшин: Аюулгүй байдлын шинж чанарын ихэсгэсэн эсвэл багасгасан тохиргоо. Түвшин ихсэх тусам түүний аюулгүй байдал бас дээшилнэ гэж үздэг.
- *low water mark* буюу доод түвшин: Доод түвшин нь тийм ч аюулгүй биш мэдээлэлд хандахын тулд аюулгүй байдлын түвшингүүдийг доошлуулахыг зөвшөөрдөг бодлого юм. Ихэнх тохиолдолд процесс дууссаны дараа хэрэглэгчийн анхдагч аюулгүй байдлын түвшин сэргээгддэг. FreeBSD-д үүнийг ашигладаг цорын ганц аюулгүй байдлын бодлогын модуль бол mac_lomac(4) юм.
- *multilabel* буюу олон хаяг/шошго: `multilabel` өмч нь ганц хэрэглэгчийн горимд tunefs(8) хэрэгсэл, ачаалалтын үйлдлүүдийн үед эсвэл шинэ файлын систем үүсгэх үед fstab(5) файл ашиглан тохируулж болох файлын системийн тохируулга юм. Энэ тохируулга нь өөр өөр обьектуудад өөр өөр MAC хаяг/шошгонуудыг хамааруулахыг администраторт зөвшөөрөх болно. Энэ тохируулга нь хаяглалтыг дэмждэг аюулгүй байдлын бодлогын модулиудад зөвхөн хамаардаг.
- *object* буюу обьект: Обьект буюу системийн обьект нь *subject* буюу субьектийн удирдлагын доор мэдээлэл дамжин урсдаг тэр мөн чанар юм. Үүнд сангууд, файлууд, талбарууд, дэлгэцүүд, гарууд, санах ой, соронзон хадгалалт, хэвлэгчид эсвэл бусад дурын хадгалалт/хөдлөх төхөөрөмж ордог. Үндсэндээ обьект нь өгөгдлийн чингэлэг эсвэл системийн эх үүсвэр юм; _обьект_од хандах нь өгөгдөлд хандана гэсэн үг юм.
- *policy* буюу бодлого: Зорилгод хэрхэн хүрэхийг тодорхойлох дүрмүүдийн цуглуулга юм. *Бодлого* нь ихэвчлэн зарим нэг зүйлүүдтэй хэрхэн ажиллахыг баримтжуулдаг. Энэ бүлэг нь сэдэв дахь *бодлого* гэсэн энэ нэр томъёог *аюулгүй байдлын бодлого* гэж үзэх болно; өөрөөр хэлбэл өгөгдөл болон мэдээллийн урсгалыг хянах дүрмүүдийн цуглуулга гэж үзэх бөгөөд тэр өгөгдөл болон мэдээлэлд хэн хандалттай байхыг тодорхойлох болно.
- *sensitivity* буюу мэдрэмтгий байдал: MLS-ийг хэлэлцэж байх үед ихэвчлэн хэрэглэдэг. Мэдрэмтгий байдлын түвшин нь өгөгдөл ямар чухал эсвэл нууцлаг байх ёстой болохыг тайлбарлахад хэрэглэгддэг нэр томъёо юм. Мэдрэмтгий байдлын түвшин ихсэх тусам нууцгай байдлын чухал ач холбогдол эсвэл өгөгдлийн итгэмжлэгдсэн байдал бас ихэсдэг.
- *single label* буюу ганц хаяг/шошго: Ганц хаяг/шошго нь өгөгдлийн урсгалд хандалтын хяналт хийхийн тулд бүхэл файлын систем ганц хаяг/шошгыг хэрэглэх үе юм. `multilabel` тохируулгыг тохируулаагүй ямар ч үед файлын систем үүнийг тохируулсан байхад бүх файлууд нь ижил хаяг/шошгоны тохиргоог дагах болно.
- *subject* буюу субьект: субьект нь хэрэглэгч, хэрэглэгчийн процессор, системийн процесс гэх мэт _обьектууд_ийн хооронд мэдээллийг урсгах идэвхтэй мөн чанар юм. FreeBSD дээр энэ нь бараг үргэлж хэрэглэгчийн өмнөөс процессод үйлчилж байгаа thread буюу урсгал байдаг.

# 17.3. MAC-ийн тайлбар

Энэ бүх шинэ ухагдахуунуудыг санаад MAC тогтолцоо хэрхэн системийн аюулгүй байдлыг ерөнхийд нь нэмэгдүүлдэгийг эргэцүүлье. MAC тогтолцооны хангадаг төрөл бүрийн аюулгүй байдлын модулиуд нь сүлжээ болон файлын системүүдийг хамгаалах, зарим портууд болон сокетуудад хэрэглэгчид хандахыг хаах гэх зэрэгт ашиглагдаж болно. Магадгүй бодлогын модулиудыг ашиглах хамгийн шилдэг арга нь хэд хэдэн аюулгүй байдлын бодлогын модулиудыг нэг зэрэг олон давхаргажсан аюулгүй байдлын орчны хувьд дуудаж тэдгээрийг холих явдал байж болох юм. Олон давхаргажсан аюулгүй байдлын орчинд олон бодлогын модулиуд нь аюулгүй байдлыг шалгаж ажиллаж байдаг. Энэ нь зөвхөн тусгай зориулалтаар ашиглаж байгаа системийн элементүүдийг ихэвчлэн хатуужуулдаг чангатгах бодлогоос өөр юм. Цорын ганц сул тал нь олон файлын системийн хаяг/шошгонууд, сүлжээний хандалтын хяналтыг хэрэглэгч бүр дээр тохируулах гэх мэт тохиолдлуудад удирдлагын хувьд илүү ажилтай байдаг явдал юм.

Сул талууд нь тогтолцооны үйлчлэх нөлөөлөлтэй харьцуулахад бага зүйл юм. Жишээ нь тусгайлсан тохиргоонд ямар бодлогууд шаардлагатайг шилж сонгох чадвар нь ажиллагааны хувьд илүү ачааллыг багасгадаг. Хэрэгцээгүй бодлогуудын дэмжлэгийг багасгах нь системийн нийт ажиллагааг нэмэгдүүлэхээс гадна сонголтын уян хатан байдлыг санал болгодог. Сайн шийдэл нь аюулгүй байдлын ерөнхий шаардлагуудыг бодолцож энэ тогтолцооны санал болгодог төрөл бүрийн аюулгүй байдлын модулиудыг үр ашигтайгаар авч хэрэгжүүлдэг.

Тиймээс MAC боломжуудыг ашигладаг систем нь хэрэглэгчийн хүссэнээрээ аюулгүй байдлын шинж чанаруудыг өөрчлөх боломжийг хамгийн багаар бодоход зөвшөөрөхгүй байж баталгаажуулах ёстой юм. Хэрэглэгчийн бүх хэрэгслүүд, програмууд болон скриптүүд нь сонгосон аюулгүй байдлын бодлогын модулиудын хандалтын дүрмүүдийн шахалтын доор ажиллах ёстой бөгөөд MAC хандалтын дүрмүүдийн ерөнхий хяналт нь системийн администраторын гарт байдаг байна.

Аюулгүй байдлын бодлогын модулиудыг анхааралтай сонгох нь системийн администраторын цорын ганц үүрэг байдаг. Зарим орчнуудын хувьд сүлжээнд хандалтын хяналтыг хязгаарлах хэрэгтэй байдаг. Ийм тохиолдлуудад mac_portacl(4), mac_ifoff(4) болон бүр mac_biba(4) бодлогын модулиуд зөв эхлэл болж болох юм. Бусад тохиолдлуудад файлын системийн обьектуудын чанд нууцлал/итгэмжлэлийг шаардаж болох юм. Энэ зорилгоор mac_bsdextended(4) болон mac_mls(4) зэрэг бодлогын модулиуд байдаг.

Сүлжээний тохиргоон дээр үндэслэн бодлогын шийдвэрүүдийг хийдэг. Магадгүй сүлжээ эсвэл Интернэтэд хандахын тулд ssh(1)-ийн хангадаг боломжуудад зөвхөн зарим нэг хэрэглэгчдийг хандахыг зөвшөөрөх ёстой байж болох юм. Эдгээр тохиолдлуудад mac_portacl(4) нь сонгох бодлогын модуль болох юм. Гэхдээ файлын системүүдийн хувьд юу хийх ёстой вэ? Зарим нэг сангуудад бусад бүлгүүдээс эсвэл тусгай хэрэглэгчдээс хандах бүх хандалтыг чангаруулах ёстой юу? Эсвэл тусгай файлууд уруу хийх хэрэглэгчийн эсвэл хэрэгслийн хандалтыг зарим обьектуудыг нууц гэж тохируулан бид хязгаарлах ёстой юу?

Файлын системийн тохиолдолд обьектуудад хандах хандалт нь зарим хэрэглэгчдийн хувьд итгэмжлэгдсэн/нууц, бусдуудын хувьд үгүй байж болох юм. Жишээ нь хөгжүүлэх том багийг хэд хэдэн хөгжүүлэгчдээс тогтох жижиг бүлгүүдэд хувааж болох юм. B төсөл дэх хөгжүүлэгчдийн бичсэн обьектуудад A төсөл дэх хөгжүүлэгчид хандах ёсгүй. Бас тэд C төсөл

дэх хөгжүүлэгчдийн үүсгэсэн обьектуудад хандах хэрэгтэй байж болох юм. Ийм тохиолдол харин ч байж болох юм. MAC тогтолцооны өөр өөр аюулгүй байдлын бодлогын модулиудыг ашиглан хэрэглэгчдийг эдгээр бүлгүүдэд хувааж мэдээллийн алдагдлаас айлгүйгээр тохирох талбаруудад хандалтыг өгч болох юм.

Тиймээс аюулгүй байдлын бодлогын модуль бүр нь системийн ерөнхий аюулгүй байдлыг сайжруулах өвөрмөц аргатай байдаг. Модулийн сонголтыг хийхдээ аюулгүй байдлын бодлогын хувьд сайн бодож хийх хэрэгтэй. Ихэнх тохиолдлуудад ерөнхий бодлогыг дахин харж сайжруулан систем дээр дахин хэрэгжүүлэх хэрэгтэй байж болох юм. MAC тогтолцооны санал болгодог өөр өөр аюулгүй байдлын бодлогын модулиудыг ойлгох нь администраторуудад өөр өөрсдийн нөхцөлдөө тохируулан хамгийн шилдэг бодлогуудыг сонгоход туслах болно.

FreeBSD-ийн анхдагч цөм нь MAC тогтолцоонд зориулсан тохируулгагүй байдаг, тиймээс энэ бүлэгт байгаа жишээнүүд эсвэл мэдээллийг туршихаасаа өмнө дараах цөмийн тохируулгыг нэмэх ёстой:

```
options MAC
```

Тэгээд цөмийг дахин бүтээж суулгах шаардлагатай болно.

MAC бодлогын модулиудын төрөл бүрийн гарын авлагын хуудаснууд нь тэдгээрийг цөмд оруулан бүтээсэн гэж мэдэгддэг боловч системийг сүлжээнээс гаргаж түгжих зэрэг олон боломжтой байдаг. MAC-ийг хэрэгжүүлэх нь галт ханыг хэрэгжүүлэхтэй бараг адил бөгөөд системээс бүр мөсөн гарч түгжигдэхээс сэргийлэхийн тулд анхааралтай байх ёстой. Өмнөх тохиргоондоо эргэж буцааж болдог байх чадварыг бодолцох ёстой бөгөөд MAC шийдлийг алсаас хийхдээ маш болгоомжтой хийх хэрэгтэй юм.

# 17.4. MAC хаяг/шошгонуудыг ойлгох нь

MAC хаяг/шошго нь системийн турш нэлэнхүйд нь субьектууд болон обьектуудад өгч болох аюулгүй байдлын шинж чанар юм.

Хаяг/шошгыг тохируулах үед хэрэглэгч үүнийг яг юу болох, юу хийгдэхийг ойлгож чадаж байх ёстой. Обьект дээр байдаг шинж чанарууд нь бодлогын модуль дуудагдсан болон бодлогын модулиуд тэдгээрийн шинж чанаруудыг өөр аргаар ойлгуулдгаас хамаарна. Дутуу ойлгосноос эсвэл утга санаанууудыг нь ойлгох чадваргүй байдлаас болоод буруу тохируулсан бол үр дүн нь тааж болшгүй байх бөгөөд магадгүй системийн хүсээгүй ажиллагаанд хүргэж болох юм.

Обьект дээрх аюулгүй байдлын хаяг/шошго нь бодлогын гаргах аюулгүй байдлын хандалтын хяналтын шийдвэрийн хэсэг болон хэрэглэгддэг. Зарим бодлогуудад хаяг/шошго нь өөрөө шийдвэр гаргахад шаардлагатай бүх мэдээллийг агуулдаг; бусад загваруудад хаяг/шошгонууд нь илүү том дүрмийн олонлогийн хэсэг болон процесс хийгдэж болох юм. Гэх мэт олныг дурдаж болно.

Жишээ нь файл дээр `biba/low` гэж хаяг/шошгыг тохируулах нь Biba аюулгүй байдлын бодлогын модулиар хангагдаж байдаг хаяг/шошгыг "low" гэсэн утгатайгаар илэрхийлж байна гэсэн үг юм.

FreeBSD-д хаяглалтын боломжийг дэмждэг цөөн бодлогын модулиуд нь урьдчилан тодорхойлсон тусгай гурван хаяг/шошгыг санал болгодог. Эдгээр нь low буюу доод, high буюу өндөр болон equal буюу тэнцүү гэсэн хаяг/шошгууд юм. Тэдгээр нь хандалтын хяналтыг бодлогын модуль бүртэй өөр өөрөөр хийдэг боловч low хаяг/шошго нь хамгийн доод тохиргоо болох ба equal хаяг/шошго нь субьект эсвэл обьектийг хаах эсвэл хамаарахгүй гэж тохируулах бөгөөд high хаяг/шошго нь Biba болон MLS бодлогын модулиудад байх хамгийн дээд тохиргоог хийх болно.

Ганц хаяг/шошго бүхий файлын системийн орчинд обьектууд дээр зөвхөн нэг хаяг/шошго хэрэглэгдэх болно. Энэ нь хандалтын зөвшөөрлүүдийн нэг олонлогийг бүхэл бүтэн системийн дагуу ашиглах бөгөөд олон орчны хувьд энэ нь хангалттай байж болох юм. Файлын систем дэх обьектууд эсвэл субьектууд дээр олон хаяг/шошгонууд тавих цөөн тохиолдлууд байдаг. Ийм тохиолдолд `multilabel` тохируулгыг tunefs(8) уруу дамжуулж өгч болох юм.

Biba болон MLS-ийн хувьд тоон хаяг/шошгыг шаталсан хяналтын тодорхой түвшинг заахын тулд тохируулж болно. Энэ тоон түвшин нь мэдээллийг ангиллын өөр өөр бүлгүүдэд хуваах буюу эрэмбэлж тэр бүлэг эсвэл илүү өндөр бүлгийн түвшинд хандах хандалтыг зөвхөн зөвшөөрөхөд хэрэглэгддэг.

Ихэнх тохиолдлуудад администратор нь файлын системийн дагуу хэрэглэхийн тулд зөвхөн ганц хаяг/шошгыг тохируулдаг.

*Хөөе хүлээгээрэй, энэ нь DAC-тай адил юм байна! MAC нь хяналтыг зөвхөн администраторт өгдөг гэж бодсон.* Энэ өгүүлбэр нь зарим талаараа үнэн хэвээр байгаа, учир нь `root` хэрэглэгчид хяналт байгаа бөгөөд тэрээр хэрэглэгчдийг тохирох зэрэглэл/хандалтын түвшингүүдэд байрлуулахаар бодлогуудыг тохируулдаг. Харамсалтай нь бодлогын олон модулиуд нь `root` хэрэглэгчийг бас хязгаарлаж чадна. Обьектууд дээрх үндсэн хяналт нь тэгээд бүлэгт суллагдах боловч `root` нь тохиргоонуудыг ямар ч үед буцааж эсвэл өөрчилж болох юм. Энэ нь Biba болон MLS зэрэг бодлогуудын хамардаг шаталсан/цэвэрлэгээ загвар юм.

### 17.4.1. Хаяг/шошгоны тохиргоо

Хаяг/шошгоны бодлогын модулийн тохиргооны бараг л бүх зүйлсийг үндсэн системийн хэрэгслүүдийг ашиглан гүйцэтгэдэг. Эдгээр тушаалууд нь обьект эсвэл субьектийн тохиргоо эсвэл тохиргооны удирдлага болон шалгалтын хувьд энгийн интерфэйсээр хангадаг.

Бүх тохиргоог setfmac(8) болон setpmac(8) хэрэгслүүдийг ашиглан хийнэ. `setfmac` тушаал нь системийн обьектууд дээр MAC хаяг/шошгонуудыг тохируулахад хэрэглэгддэг бол `setpmac` тушаал нь системийн субьектууд дээр хаяг/шошгонуудыг тохируулахад хэрэглэгддэг. Дараах тушаалыг ажиглаарай:

```
# setfmac biba/high test
```

Дээрх тушаалыг ажиллуулсны дараа хэрэв ямар ч алдаа гараагүй бол хүлээх мөр буцаагдах болно. Эдгээр тушаалууд нь хөдөлгөөнгүй биш байх цорын ганц үе нь алдаа гарах үе юм; chmod(1) болон chown(8) тушаалуудтай адил юм. Зарим тохиолдолд энэ алдаа нь `Permission denied` гэсэн байж болох бөгөөд энэ нь ихэвчлэн хязгаарласан обьект дээр хаяг/шошгыг тохируулах буюу засах үед гардаг. Системийн администратор үүнийг давж гарахын тулд дараах тушаалуудыг ашиглаж болно:

```
# setfmac biba/high test
Permission denied
# setpmac biba/low setfmac biba/high test
# getfmac test
test: biba/high
```

Дээрхээс харахад ажиллуулсан процессод өөр хаяг/шошго зааж бодлогын модулийн тохиргоонуудыг өөрчлөхөд `setpmac` тушаалыг хэрэглэж болох юм байна. `getpmac` хэрэгсэл нь ихэвчлэн тухайн үед ажиллаж байгаа sendmail зэрэг процессуудад хэрэглэгддэг. Хэдийгээр энэ нь тушаалын оронд процессийн ID-г авдаг боловч логик нь туйлын төстэй юм. Хэрэв хэрэглэгчид өөрийн хандалтад байхгүй файлыг удирдахыг оролдвол дуудагдсан бодлогын модулиудын дүрмүүдээс болоод `Operation not permitted` алдаа `mac_set_link` функцээр харуулагдах болно.

#### 17.4.1.1. Нийтлэг хаяг/шошгоны төрлүүд

mac_biba(4), mac_mls(4) болон mac_lomac(4) бодлогын модулиудын хувьд энгийн хаяг/шошгонуудыг зааж өгөх боломж олгогдсон байдаг. Эдгээр нь high буюу өндөр/дээд, equal буюу тэнцүү болон low буюу доод гэсэн хэлбэрийг авах бөгөөд эдгээр хаяг/шошгонуудын юу хангадаг талаар товч тайлбарыг доор дурдав:

- `low` хаяг/шошго нь обьект эсвэл субьектийн авч болох хамгийн доод хаяг/шошгоны тохиргоо гэгддэг. Үүнийг обьектууд эсвэл субьектууд дээр тохируулах нь өндөр гэж тэмдэглэгдсэн обьектууд эсвэл субьектууд уруу хандах тэдгээрийн хандалтыг хаах болно.
- `equal` хаяг/шошго нь бодлогоос чөлөөлөгдөх обьектууд дээр зөвхөн тавигдах ёстой.
- `high` хаяг/шошго нь обьект эсвэл субьектэд хамгийн их боломжит тохиргоог зөвшөөрдөг.

Бодлогын модуль бүрийн хувьд тэдгээр тохиргоо бүр өөр өөр мэдээллийн урсгалын зааврыг хийх болно. Тохирох гарын авлагын хуудаснуудыг унших нь эдгээр ерөнхий хаяг/шошгоны тохиргоонуудын төрх байдлыг цаашид тайлбарлах болно.

##### 17.4.1.1.1. Хаяг/шошгоны илүү нарийн тохиргоо

Тоон зэргээр илэрхийлсэн хаяг/шошгонууд нь `comparison:compartment+compartment` буюу `харьцуулалт:тасалгаа+тасалгаа` гэсэнд зориулагдаж хэрэглэгддэг, тиймээс дараах нь:

```
biba/10:2+3+6(5:2+3-20:2+3+4+5+6)
```

Ингэж тайлбарлагдаж болно:

"Biba Бодлогын Хаяг/Шошго"/"Зэрэг 10" :"Тасалгаанууд 2, 3 болон 6": ("зэрэг 5 ...")

Энэ жишээн дээр эхний зэрэг нь "эффектив тасалгаанууд"тай "эффектив зэрэг" гэж тооцогддог, хоёр дахь зэрэг нь доод зэрэг бөгөөд хамгийн сүүлийнх нь өндөр зэрэг юм. Ихэнх тохиргоонуудад эдгээр тохируулгуудыг ашигладаггүй, харин тэдгээрийг илүү нарийн тохиргоонд зориулж санал болгодог.

Системийн обьектуудад хамааруулахад тэдгээр нь системийн субьектуудтай харьцуулах юм бол зөвхөн тухайн үеийн зэрэг/тасалгаануудтай байдаг. Системийн субьектууд нь систем болон сүлжээний интерфэйсүүдэд байгаа эрхүүдийн хүрээг тусгадаг. Сүлжээний интерфэйсүүд дээр хандалтын хяналтын хувьд хаяг/шошгонууд нь ашиглагддаг.

Субьект болон обьект хослол дахь зэрэг болон тасалгаанууд нь "давамгайлал" гэгддэг харилцааг бүтээхэд хэрэглэгддэг. Энэ харилцаанд субьект нь обьектийг давамгайлдаг, эсвэл обьект нь субьектийг давамгайлдаг, эсвэл аль нэг нь нөгөөгөө давамгайлахгүй, эсвэл хоёулаа нэг нэгнийгээ давамгайлдаг. "хоёулаа давамгайлах" тохиолдол нь хоёр хаяг/шошго тэнцүү байхад тохиолддог. Biba-ийн мэдээллийн урсгалын мөн чанараас болоод төсөлд тохирох "мэдэх хэрэгтэй" тасалгаануудын олонлогийн эрхүүд танд байдаг. Гэхдээ обьектууд нь бас тасалгаануудын олонлогтой байна. Хэрэглэгчид нь өөрсдөө хязгаарлалтгүй байдаг тасалгаа дахь обьектуудад хандахын тулд `su` эсвэл `setpmac` тушаалуудыг ашиглан өөрсдийнхөө эрхүүдийг дэд эрхүүд болгож болох юм.

#### 17.4.1.2. Хэрэглэгчид болон хаяг/шошгоны тохиргоонууд

Хэрэглэгчдийн өөрсдийнх нь файлууд болон процессууд систем дээр тодорхойлсон аюулгүй байдлын бодлоготой зөв харилцан ажилладаг байхын тулд хэрэглэгчид нь өөрсдөө хаяг/шошгонуудтай байх шаардлагатай байдаг. Үүнийг login.conf файлд нэвтрэлтийн ангиллуудыг ашиглан тохируулдаг. Хаяг/шошгонуудыг ашигладаг бодлогын модуль бүр хэрэглэгчийн ангиллын тохиргоог хийх болно.

Бодлогын модуль бүрийн тохиргоог агуулах жишээ оруулгыг доор үзүүлэв:

```
default:\
    :copyright=/etc/COPYRIGHT:\
    :welcome=/etc/motd:\
    :setenv=MAIL=/var/mail/$,BLOCKSIZE=K:\
    :path=~/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/usr/local/bin:\
    :manpath=/usr/shared/man /usr/local/man:\
    :nologin=/usr/sbin/nologin:\
    :cputime=1h30m:\
    :datasize=8M:\
    :vmemoryuse=100M:\
    :stacksize=2M:\
    :memorylocked=4M:\
    :memoryuse=8M:\
    :filesize=8M:\
    :coredumpsize=8M:\
    :openfiles=24:\
    :maxproc=32:\
    :priority=0:\
```

```
    :requirehome:\
    :passwordtime=91d:\
    :umask=022:\
    :ignoretime@:\
    :label=partition/13,mls/5,biba/10(5-15),lomac/10[2]:
```

label тохируулга нь хэрэглэгчийн ангиллын MAC-ийн үйлчлэх анхдагч хаяг/шошгыг тохируулахад хэрэглэгддэг. Хэрэглэгчид энэ уттыг өөрчлөх зөвшөөрөл хэзээ ч өгөгдөхгүй учраас энэ нь хэрэглэгчийн хувьд сонгох боломжгүй юм. Гэхдээ жинхэнэ тохиргоон дээр администратор нь бодлогын модуль бүрийг идэвхжүүлэхийг хэзээ ч хүсэхгүй. Энэ тохиргоонуудаас аль нэгийг нь хийж гүйцэтгэхээсээ өмнө энэ бүлгийн үлдсэнийг дахин шалгаж уншихыг зөвлөж байна.

Хэрэглэгчид нь эхний нэвтрэлтийнхээ дараа өөрсдийн хаяг/шошгыг өөрчилж болох юм. Гэхдээ энэ өөрчлөлт нь бодлогын шахалтуудын эрхшээлд байдаг. Дээрх жишээ нь процессийн хамгийн бага бүрэн бүтэн байдлыг 5, түүний хамгийн их утга нь 15, гэхдээ анхдагч эффектив хаяг/шошго нь 10 гэж Biba бодлогод хэлж байна. Процесс нь магадгүй хэрэглэгч setpmac тушаалыг ажиллуулснаас болоод хаяг/шошгоо өөрчлөхөөр сонгох хүртэл 10 дээр ажиллах болно. setpmac тушаал нь нэвтрэлтийн үед хүрээг тохируулах Biba-ийн шахалтад байх болно.

Бүх тохиолдлуудад login.conf-д өөрчлөлт хийсний дараа нэвтрэлтийн ангиллын боломжийн мэдээллийн баазыг cap_mkdb тушаал ашиглан дахин бүтээх ёстой бөгөөд энэ нь ойртож байгаа жишээ эсвэл хэлэлцүүлэг бүрт тусгагдах болно.

Олон сайтууд нь хэд хэдэн өөр өөр хэрэглэгчийн ангиллуудыг шаарддаг ялангуяа асар их тооны хэрэглэгчидтэй байж болохыг тэмдэглэх хэрэгтэй юм. Маш сайн төлөвлөх хэрэгтэй бөгөөд удирдахад туйлын хэцүү болж болох юм.

#### 17.4.1.3. Сүлжээний интерфэйсүүд болон хаяг/шошгоны тохиргоонууд

Хаяг/шошгонууд нь сүлжээний дагуух өгөгдлийн урсгалыг хянахад туслах зорилгоор сүлжээний интерфэйсүүд дээр бас тавигдаж болно. Бүх тохиолдолд тэдгээр нь бодлогууд обьектуудад үйлчилдэг шигээр үйлчилдэг. biba дээрх өндөр тохиргоонуудтай хэрэглэгчдийг жишээ нь доод хаяг/шошготой сүлжээний интерфэйсүүдэд хандахыг зөвшөөрдөггүй.

Сүлжээний интерфэйсүүд дээр MAC хаяг/шошгыг тохируулахдаа maclabel тохируулгыг ifconfig тушаал уруу өгч болох юм. Жишээ нь:

```
# ifconfig bge0 maclabel biba/equal
```

тушаал нь biba/equal-ийн MAC хаяг/шошгыг bge(4) интерфэйс дээр тохируулах болно. biba/high(low-high)-тай төстэй тохиргоог ашиглаж байх үед бүх хаяг/шошгыг тэр чигээр нь хаалтанд ("") хийх ёстой, тэгэхгүй бол алдаа буцаагдах болно.

Хаяглалтыг дэмждэг бодлогын модуль бүр тааруулах боломжтой хувьсагчтай байдаг бөгөөд тэдгээрийг сүлжээний интерфэйсүүд дээр MAC хаяг/шошгыг хаахдаа хэрэглэж болох юм.

Хаяг/шошгыг equal буюу тэнцүү гэж тохируулах нь ижил нөлөөлөлтэй байх болно. Тэдгээр тааруулах боломжтой хувьсагчуудын хувьд sysctl-ийн тушаалын гаралт, бодлогын гарын авлагын хуудаснууд эсвэл бүр энэ бүлгийн үлдсэн хэсэг дэх мэдээллийг дахин үзээрэй.

### 17.4.2. Ганц хаяг/шошго уу эсвэл олон хаяг/шошго уу?

Анхдагчаар систем нь singlelabel тохируулгыг ашиглах болно. Гэхдээ энэ нь администраторт юу гэж ойлгогдох вэ? Хэд хэдэн ялгаанууд байдаг бөгөөд тэдгээр нь системийн аюулгүй байдлын загварт уян хатан чанарын хувьд давуу болон сул талуудыг үзүүлдэг.

singlelabel нь зөвхөн нэг хаяг/шошгоны хувьд зөвшөөрөх бөгөөд жишээлбэл biba/high-ийг субьект эсвэл обьект бүрийн хувьд ашиглах юм. Энэ нь удирдлагын хувьд бага ажиллагааг өгдөг боловч хаяглалтыг дэмждэг бодлогуудын уян хатан чанарыг бууруулдаг. Олон администраторууд өөрсдийн аюулгүй байдлын бодлогодоо multilabel тохируулгыг ашиглахыг хүсэж болох юм.

multilabel тохируулга нь субьект эсвэл обьект бүрийг хуваалтад зөвхөн нэг хаяг/шошгыг зөвшөөрөх стандарт singlelabel тохируулгын оронд өөрийн гэсэн тусдаа MAC хаягтай байхыг зөвшөөрөх болно. multilabel болон single хаяг/шошгоны тохируулгууд нь Biba, Lomac, MLS болон SEBSD зэрэг хаяглалтын боломжийг хийж гүйцэтгэдэг бодлогуудад зөвхөн шаардлагатай байдаг.

Ихэнх тохиолдолд multilabel-ийг тохируулах ерөөсөө хэрэггүй байж болох юм. Дараах тохиолдол болон аюулгүй байдлын загварыг авч үзье:

- MAC тогтолцоо болон төрөл бүрийн бодлогуудын холимгийг ашигладаг FreeBSD вэб сервер.
- Энэ машин нь зөвхөн нэг хаяг/шошго biba/high-ийг системийн бүх юмандаа шаарддаг. Энд ганц хаяг/шошго нь үргэлж нөлөөлөх болохоор файлын систем нь multilabel тохируулгыг шаардахгүй.
- Гэхдээ энэ машин нь вэб сервер болох бөгөөд бичих боломжоос хамгаалахын тулд вэб серверийг biba/low-д ажиллуулах ёстой. Biba бодлого болон энэ нь хэрхэн ажилладаг талаар сүүлд хэлэлцэх болно. Тэгэхээр хэрэв өмнөх тайлбар ойлгоход хэцүү байгаа бол зүгээр л цааш үргэлжлүүлэн уншаад буцаж эргэж ирээрэй. Сервер нь ажиллаж байх үеийн төлвийнхээ ихэнх үед biba/low тавигдсан тусдаа хуваалтыг ашиглаж болох юм. Энэ жишээн дээр нэлээн их зүйл байхгүй байгаа, жишээ нь өгөгдөл, тохиргоо болон хэрэглэгчийн тохиргоонууд дээр хязгаарлалтууд байхгүй; гэхдээ энэ нь зөвхөн дээр дурдсаныг батлах хурдхан жишээ юм.

Хэрэв хаягладаггүй бодлогуудын аль нэг ашиглагдах бол multilabel тохируулга хэзээ ч шаардагдахгүй. Эдгээрт seeotheruids, portacl болон partition бодлогууд ордог.

Хуваалтад multilabel тохируулгыг ашиглаж multilabel-ийн ажиллагаан дээр тулгуурласан аюулгүй байдлын загварыг байгуулах нь удирдлагын хувьд илүү ажиллагаанд хүргэж болох юм. Учир нь файлын систем дэх бүх зүйлс хаяг/шошготой болох юм. Эдгээр зүйлсэд сангууд, файлууд, болон бүр төхөөрөмжийн цэгүүд хүртэл орно.

Дараах тушаал нь файлын системүүд дээр олон хаяг/шошготой байхаар multilabel-ийг

тохируулна. Үүнийг зөвхөн ганц хэрэглэгчийн горимд хийж болно:

```
# tunefs -l enable /
```

Энэ нь swap файлын системийн хувьд шаардлагатай биш юм.

Зарим хэрэглэгчид `multilabel` тугийг root хуваалт дээр тохируулахад асуудлуудтай тулгарсан байж болох юм. Хэрэв ийм тохиолдол бол энэ бүлгийн MAC Тогтолцооны алдааг олж засварлах хэсгийг дахин үзнэ үү.

# 17.5. Аюулгүй байдлын тохиргоог төлөвлөх нь

Шинэ технологи хийгдэх болгонд төлөвлөлтийн үе шат үргэлж зөв зүйтэй санаа байдаг. Төлөвлөх шатуудын үеэр администратор ерөнхийд нь "том дүр зургийг" харах ёстой бөгөөд ядаж дараах зүйлүүдийг хараандаа байлгаж байх хэрэгтэй:

- Шийдлийн шаардлагууд;
- Шийдлийн зорилгууд;

MAC суулгацуудын хувьд эдгээрт дараах зүйлс орно:

- Системүүд дээр байгаа мэдээлэл болон эх үүсвэрүүдийг хэрхэн ангилах.
- Мэдээлэл ба эх үүсвэрүүдийн ямар төрлүүдэд хандахыг хийгдэх ёстой хязгаарлалтуудын төрлийн хамтаар хязгаарлах.
- Энэ зорилгод хүрэхийн тулд аль MAC модуль эсвэл модулиуд шаардлагатай болох.

Системийн эх үүсвэрүүд болон аюулгүй байдлын тохиргоонуудыг дахин тохируулж өөрчлөх боломж үргэлж байдаг бөгөөд системээс хайж файлууд болон хэрэглэгчийн бүртгэлүүдийг засах нь ихэвчлэн маш тохиромжгүй байдаг. Төлөвлөх нь ямар нэг асуудалгүй, үр ашигтай итгэгдсэн системийг бүтээхэд туслах юм. Тохиргоо бүхий итгэгдсэн системийн туршилт нь ихэвчлэн амин чухал байдаг бөгөөд MAC шийдлийг жинхэнэ ажиллах системүүд дээр ашиглахаас *өмнө* лавтай ашигтай байдаг билээ. MAC бүхий систем дээр сул тохируулж орхих нь амжилтгүй байдлыг тохируулж байна гэсэн үг юм.

Өөр өөр орчнууд өөр тусгай хэрэгцээ болон шаардлагуудтай байж болох юм. Гүнзгий, бүрэн гүйцэд аюулгүй байдлын хувийн тохируулгыг үүсгэх нь систем ажиллагаанд орсны дараа өөрчлөлтүүдийн хэрэгцээг багасгах болно. Тиймээс дараа дараагийн хэсгүүд администраторуудад байдаг өөр өөр модулиудын талаар өгүүлэх бөгөөд тэдгээрийн хэрэглээ болон тохиргоог тайлбарлаж зарим тохиолдолд тэдгээр нь ямар нөхцөл байдлын үед хамгийн тохиромжтой байхыг харуулах болно. Жишээ нь вэб сервер нь mac_biba(4) болон mac_bsdextended(4) бодлогуудыг ашиглаж болох юм. Бусад тохиолдлуудад жишээ нь маш цөөн локал хэрэглэгчидтэй машины хувьд mac_partition(4) магадгүй зөв сонголт болж болох юм.

## 17.6. Модулийн тохиргоо

MAC тогтолцоонд орсон модуль бүр дээр дурдсан шиг цөмд эмхэтгэгдэж эсвэл цөмийн ажиллах үеийн модуль хэлбэрээр дуудагдаж болно. Бидний зөвлөдөг арга бол модулийг эхний ачаалалтын үйлдлийн үеэр дуудагдахаар болгож модулийн нэрийг /boot/loader.conf файлд нэмэх явдал юм.

Дараах хэсгүүд нь төрөл бүрийн MAC модулиудыг хэлэлцэж тэдгээрийн боломжуудыг тайлбарлах болно. Тэдгээрийг тусгай орчинд хийж гүйцэтгэхийг энэ бүлэг бас хамрах болно. Зарим модулиуд хаяглалтын хэрэглээг дэмждэг бөгөөд хаяглалт нь "энийг зөвшөөрсөн, харин энийг зөвшөөрөөгүй" гэх зэрэг хаяг/шошгыг хэрэгжүүлж хандалтыг хянадаг байна. Хаяг/шошгоны тохиргооны файл нь файлуудад хэрхэн хандаж болох, сүлжээний холболтыг хэрхэн солилцож болох гэх зэрэг олон асуудлуудыг хянадаг. Өмнөх хэсэг нь файл бүрийн эсвэл хуваалт бүрийн хандалтын хяналтыг идэвхжүүлэхийн тулд `multilabel` тугийг файлын системүүдэд хэрхэн тохируулах талаар үзүүлсэн.

Ганц хаяг/шошго бүхий тохиргоо нь системийн дагуу зөвхөн нэг хаяг/шошгыг хэрэглэх бөгөөд ийм учраас `tunefs`-ийн тохируулга `multilabel` гэж нэрлэгдсэн юм.

## 17.7. MAC seeotheruids модуль

Модулийн нэр: mac_seeotheruids.ko

Цамийн тохиргооны мөр: `options MAC_SEEOTHERUIDS`

Ачаалалтын тохируулга: `mac_seeotheruids_load="YES"`

mac_seeotheruids(4) модуль нь `sysctl`-ийн тааруулах боломжтой `security.bsd.see_other_uids` болон `security.bsd.see_other_gids` хувьсагчуудыг дуурайж өргөтгөдөг. Энэ тохируулга нь тохиргооноос өмнө ямар ч хаяг/шошгонуудыг тохируулахыг шаарддаггүй бөгөөд бусад модулиудтай хамааралгүйгээр ажиллаж чаддаг.

Модулийг дуудаж ачаалсны дараа боломжуудыг хянахын тулд дараах `sysctl`-ийн тааруулах боломжтой хувьсагчуудыг ашиглаж болно:

- `security.mac.seeotheruids.enabled` нь модулийн боломжуудыг идэвхжүүлж анхдагч тохируулгуудыг ашиглана. Эдгээр анхдагч тохируулгууд нь бусад хэрэглэгчдийн эзэмшиж байгаа процессууд болон сокетуудыг харах боломжийг хэрэглэгчдийн хувьд хаах болно.
- `security.mac.seeotheruids.specificgid_enabled` нь зарим нэг бүлгүүдийг энэ бодлогоос чөлөөлж тэдгээрийг зөвшөөрөх болно. Энэ бодлогоос зарим нэг бүлгүүдийг чөлөөлөхийн тулд `sysctl` тушаалын `security.mac.seeotheruids.specificgid=XXX` хувьсагчийг ашиглана. Дээрх жишээн дээрх *XXX*-ийг чөлөөлөх бүлгийн тоон ID-аар солих хэрэгтэй.
- `security.mac.seeotheruids.primarygroup_enabled` нь тусгай анхдагч бүлгүүдийг энэ бодлогоос чөлөөлөхийн тулд ашигладаг. Энэ хувьсагчийг хэрэглэхэд `security.mac.seeotheruids.specificgid_enabled` хувьсагч тохируулагдаагүй байж болно.

# 17.8. MAC bsdextended модуль

Модулийн нэр: mac_bsdextended.ko

Цөмийн тохиргооны мөр: `options MAC_BSDEXTENDED`

Ачаалалтын тохируулга: `mac_bsdextended_load="YES"`

mac_bsdextended(4) модуль файлын системийн галт ханыг идэвхжүүлдэг. Энэ модулийн бодлого нь стандарт файлын системийн зөвшөөрлүүдийн загварын өргөтгөл болж файлын систем дэх файлууд, хэрэгслүүд болон сангуудыг хамгаалахын тулд администраторт галт ханатай адил дүрмийн олонлогийг үүсгэх боломжийг олгодог. Файлын системийн обьектод хандахыг оролдоход дүрмүүдийн жагсаалтаас тохирох дүрэм таарах хүртэл эсвэл төгсгөл хүртэл шалгадаг. Энэ ажиллагааг sysctl(8)-ийн хувьсагч security.mac.bsdextended.firstmatch_enabled параметрийг хэрэглэж өөрчилж болно. FreeBSD дэх бусад галт ханын модулиудтай адилаар хандалтын хяналтын дүрмүүдийг агуулах файлыг үүсгэж rc.conf(5)-ийн хувьсагчийн тусламжтайгаар ачаалах үед системээр уншуулж болно.

Дүрмийн жагсаалтыг ipfw(8)-ийн синтакстай төстэйгөөр бичигддэг ugidfw(8) хэрэгслийг ашиглан оруулж болно. Илүү хэрэгслүүдийг libugidfw(3) сан дахь функцуудыг ашиглан бичиж болно.

Энэ модультай ажиллаж байхдаа маш болгоомжтой байх хэрэгтэй; учир нь буруу хэрэглээ файлын системийн зарим хэсэгт хандах боломжгүй болгож болох юм.

## 17.8.1. Жишээнүүд

mac_bsdextended(4) модуль ачаалагдсаны дараа тухайн үед байгаа дүрмийн тохиргоог жагсаахад дараах тушаал ашиглагдаж болно:

```
# ugidfw list
0 slots, 0 rules
```

Яг бодож байсны дагуу ямар ч дүрмүүд тодорхойлогдоогүй байна. Энэ нь бүгд хандах боломжтой байна гэсэн үг юм. `root`-ийг орхиж бусад хэрэглэгчдийн бүх хандалтыг хаах дүрмийг үүсгэхийн тулд ердөө л дараах тушаалыг ажиллуулна:

```
# ugidfw add subject not uid root new object not uid root mode n
```

Энэ нь бүх хэрэглэгчдийг `ls` зэрэг хамгийн энгийн тушаалуудыг ажиллуулахыг хаах учраас маш буруу санаа юм. Илүү эх оронч дүрмүүдийн жагсаалт иймэрхүү байж болно:

```
# ugidfw set 2 subject uid user1 object uid user2 mode n
# ugidfw set 3 subject uid user1 object gid user2 mode n
```

Энэ нь `user1` хэрэглэгчээс `user2`-ийн гэрийн сан уруу хандах сангийн жагсаалт үзүүлэх зэрэг дурын болон бүх хандалтыг хаах болно.

`user1`-ийн оронд `not uid user2` тохируулгыг дамжуулж болно. Энэ нь дээрхийн адил хандалтын хязгаарлалтуудыг зөвхөн нэг хэрэглэгчийн хувьд биш бүх хэрэглэгчийн хувьд тавих болно.

`root` хэрэглэгчид эдгээр өөрчлөлтүүд нөлөөлөхгүй.

Энэ нь файлын системийг бэхэлж батжуулахад туслахын тулд mac_bsdextended(4) модулийг хэрхэн ашиглаж болох ерөнхий санааг харуулах ёстой. Илүү дэлгэрэнгүй мэдээллийг mac_bsdextended(4) болон ugidfw(8) гарын авлагын хуудаснуудаас үзнэ үү.

# 17.9. MAC ifoff модуль

Модулийн нэр: mac_ifoff.ko

Цөмийн тохиргооны мөр: `options MAC_IFOFF`

Ачаалалтын тохируулга: `mac_ifoff_load="YES"`

mac_ifoff(4) модуль нь сүлжээний интерфэйсүүдийг шууд идэвхгүй болгож системийн эхний ачаалалтын үеэр идэвхжүүлэхгүй байлгах зорилгоор байдаг. Энэ нь систем дээр ямар ч хаяг/шошгуудыг тохируулахыг шаарддаггүйгээс гадна бас бусад MAC модулиудаас хамааралгүй юм.

Хяналтын ихэнх нь доор дурдсан `sysctl`-ийн тааруулж болох хувьсагчуудаар хийгддэг.

- `security.mac.ifoff.lo_enabled` нь loopback (lo(4)) буюу буцах интерфэйс дээрх бүх урсгалыг нээнэ/хаана.
- `security.mac.ifoff.bpfrecv_enabled` нь Berkeley Packet Filter буюу Беркли Пакет шүүгч интерфэйс (bpf(4)) дээрх бүх урсгалыг нээнэ/хаана.
- `security.mac.ifoff.other_enabled` нь бусад бүх интерфэйсүүд дээр бүх урсгалыг нээнэ/хаана.

mac_ifoff(4)-ийн хамгийн нийтлэг хэрэглээний нэг бол ачаалах дарааллын үеэр сүлжээний урсгалыг зөвшөөрөх ёсгүй орчинд сүлжээг монитор хийх явдал юм. Өөр нэг санал болгох хэрэглээ бол хамгаалагдсан сангуудад шинэ эсвэл өөрчлөгдсөн файлуудыг олсон тохиолдолд сүлжээний урсгалыг автоматаар хаахын тулд security/aide-г ашигладаг скриптийг бичих байж болох юм.

# 17.10. MAC portacl модуль

Модулийн нэр: mac_portacl.ko

Цөмийн тохиргооны мөр: `MAC_PORTACL`

Ачаалалтын тохируулга: `mac_portacl_load="YES"`

mac_portacl(4) модулийг төрөл бүрийн sysctl хувьсагчуудыг ашиглан локал TCP болон UDP портуудыг холбохыг хязгаарлахад хэрэглэдэг. Мөн чанартаа mac_portacl(4) нь заагдсан эрх бүхий портуудыг өөрөөр хэлбэл 1024-оос бага портуудыг холбох боломжийг root биш хэрэглэгчдэд зөвшөөрдөг.

Ачаалагдсаны дараа энэ модуль нь бүх сокетууд дээр MAC бодлогыг идэвхжүүлдэг. Дараах тааруулж болох хувьсагчууд байдаг:

- security.mac.portacl.enabled нь бодлогыг бүр мөсөн нээнэ/хаана.
- security.mac.portacl.port_high нь mac_portacl(4)-ийн хамгаалалтыг нь идэвхжүүлдэг хамгийн дээд портын дугаарыг тохируулдаг.
- security.mac.portacl.suser_exempt нь тэгээс ялгаатай утгаар тохируулагдсан үедээ root хэрэглэгчийг энэ бодлогоос чөлөөлнө.
- security.mac.portacl.rules нь яг mac_portacl бодлогыг заадаг; доорхоос харна уу.

mac_portacl бодлого нь security.mac.portacl.rules sysctl-д заагдсаны дагуу хэрэгцээнээсээ хамааран хэдэн ч дүрмүүдтэй байж болох rule[,rule,…] текст хэлбэрийн байдаг. Дүрэм бүр idtype:id:protocol:port гэсэн хэлбэрийн байдаг. idtype параметр нь uid эсвэл gid байж болох бөгөөд id параметрийг хэрэглэгчийн id эсвэл бүлгийн id гэж тайлбарладаг. protocol параметр нь tcp эсвэл udp гэж заагдан дүрмийг TCP эсвэл UDP-ийн алинд хамаарахыг тодорхойлоход хэрэглэгддэг. Сүүлийн port параметр нь заагдсан хэрэглэгч эсвэл бүлэгт холбохыг зөвшөөрөх портын дугаар юм.

Дүрмийн олонлог нь цөмөөр шууд тайлбарлагддаг болохоор хэрэглэгчийн ID бүлгийн ID болон портын параметруудын хувьд зөвхөн тоон утгуудыг ашиглаж болно. Өөрөөр хэлбэл хэрэглэгч, бүлэг болон портын үйлчилгээний нэрсийг ашиглаж болохгүй.

Анхдагчаар UNIX® төст системүүд дээр 1024-өөс бага портуудыг зөвхөн эрх бүхий процессууд буюу өөрөөр хэлбэл root-ээр ажилладаг процессуудад ашиглахад/холбоход хэрэглэдэг. mac_portacl(4)-ийн хувьд эрхгүй процессуудыг 1024-өөс бага портуудад холбохыг зөвшөөрөхдөө энэ стандарт UNIX® хязгаарлалтыг хаасан байх ёстой. Үүнийг sysctl(8)-ийн net.inet.ip.portrange.reservedlow болон net.inet.ip.portrange.reservedhigh хувьсагчуудыг тэг болгон хийж болно.

Доор жишээнүүдийг үзнэ үү, эсвэл дэлгэрэнгүй мэдээллийг mac_portacl(4) гарын авлагын хуудаснаас лавлана уу.

### 17.10.1. Жишээнүүд

Дараах жишээнүүд нь дээрх хэлэлцүүлгийг арай илүү тайлбарлах болно:

```
# sysctl security.mac.portacl.port_high=1023
# sysctl net.inet.ip.portrange.reservedlow=0 net.inet.ip.portrange.reservedhigh=0
```

Эхлээд бид mac_portacl(4)-ийг стандарт эрх бүхий портуудыг хамарч ердийн UNIX® холболтын хязгаарлалтуудыг хаахаар тохируулна.

```
# sysctl security.mac.portacl.suser_exempt=1
```

`root` хэрэглэгчийг энэ бодлогоор хязгаарлахгүйн тулд `security.mac.portacl.suser_exempt`-г тэгээс ялгаатай утгаар тохируулна. mac_portacl(4) модуль нь одоо UNIX® төст системүүд анхдагч тохиргоотойгоор ажилладаг шигээр тохируулагдсан байна.

```
# sysctl security.mac.portacl.rules=uid:80:tcp:80
```

UID 80 бүхий (ердийн тохиолдолд `www` хэрэглэгч) хэрэглэгчид 80 портыг холбохыг зөвшөөрнө. `root` эрхгүйгээр вэб сервер ажиллуулахыг `www` хэрэглэгчид зөвшөөрөхөд үүнийг ашиглаж болно.

```
# sysctl security.mac.portacl.rules=uid:1001:tcp:110,uid:1001:tcp:995
```

UID 1001 бүхий хэрэглэгчид TCP 110 ("pop3") болон 995 ("pop3s") портуудыг холбохыг зөвшөөрнө. Энэ нь 110 болон 995 портуудаар холболтуудыг хүлээн авдаг сервер эхлүүлэхийг хэрэглэгчид зөвшөөрдөг.

## 17.11. MAC хуваалтын модуль

Модулийн нэр: mac_partition.ko

Цөмийн тохиргооны мөр: `options MAC_PARTITION`

Ачаалалтын тохируулга: `mac_partition_load="YES"`

mac_partition(4) бодлого нь процессуудыг тэдгээрийн MAC хаяг/шошго дээр үндэслэн тусгай "хуваалтуудад" оруулдаг. Үүнийг jail(8)-ийн тусгай нэг төрөл гэж бодох хэрэгтэй, гэхдээ энэ нь тийм ч зохистой харьцуулалт биш юм.

Ачаалах процессийн үеэр энэ бодлогыг дуудаж идэвхжүүлэхийн тулд loader.conf(5) файлд нэмэгдэх ёстой нэг модуль нь энэ юм.

Энэ бодлогын ихэнх тохиргоо нь доор тайлбарлагдах setpmac(8) хэрэгслээр хийгддэг. Энэ бодлогод зориулагдсан дараах `sysctl`-ийн хувьсагч байдаг:

- `security.mac.partition.enabled` нь MAC процессийн хуваалтуудыг хэрэглэхийг идэвхжүүлдэг.

Энэ бодлого идэвхтэй болоход хэрэглэгчдэд зөвхөн өөрийн процессуудыг болон нэг хуваалтад байгаа бусад хэрэглэгчдийн процессуудыг харахыг зөвшөөрөх бөгөөд гэхдээ энэ хуваалтын хүрээнээс гадна байгаа хэрэгслүүдтэй ажиллахыг зөвшөөрөхгүй байх болно. Жишээ нь дээрх `insecure` ангилалд байгаа хэрэглэгчийг `top` тушаал болон процесс үүсгэх ёстой бусад олон тушаалуудад хандахыг зөвшөөрөхгүй юм.

Хэрэгслүүдийг хуваалтын хаяг/шошго уруу оруулах буюу тохируулахын тулд `setpmac`

хэрэгслийг хэрэглэнэ:

```
# setpmac partition/13 top
```

Энэ нь `top` тушаалыг `insecure` ангилал дахь хэрэглэгчдийн хаяг/шошгоны олонлогт нэмэх болно. `insecure` ангиллын хэрэглэгчдийн үүсгэсэн бүх процессууд `partition/13` хаяг/шошгод байхыг тэмдэглэх нь зүйтэй юм.

### 17.11.1. Жишээнүүд

Дараах тушаал нь хуваалтын хаяг/шошго болон процессийн жагсаалтыг танд харуулах болно:

```
# ps Zax
```

Дараагийн тушаал нь өөр хэрэглэгчийн процессийн хуваалтын хаяг/шошго болон тэр хэрэглэгчийн тухайн үед ажиллаж байгаа процессуудыг харахыг зөвшөөрөх болно:

```
# ps -ZU trhodes
```

mac_seeotheruids(4) бодлого дуудагдаж ачаалагдаагүй бол `root` хаяг/шошго дахь процессуудыг хэрэглэгч харж чадна.

Жинхэнэ ур дүй шаардсан шийдэл нь /etc/rc.conf файл дахь бүх үйлчилгээнүүдийг хааж тэдгээрт зөв хаяглалтыг тохируулж тэдгээрийг скриптээр эхлүүлдэг байж болох юм.

Дараах бодлогууд нь санал болгосон гурван анхдагч хаяг/шошгоны оронд бүхэл тоон тохируулгуудыг дэмждэг. Эдгээр тохируулгууд болон тэдгээрийн хязгаарлалтууд нь модулийн гарын авлагын хуудаснуудад дэлгэрэнгүй тайлбарлагдсан байгаа.

## 17.12. MAC олон түвшинт аюулгүй байдлын модуль

Модулийн нэр: mac_mls.ko

Цөмийн тохиргооны мөр: `options MAC_MLS`

Ачаалалтын тохируулга: `mac_mls_load="YES"`

mac_mls(4) бодлого нь систем дэх субьектууд болон обьектуудын хоорондын хандалтыг мэдээллийн урсгалын чанд бодлогын тусламжтайгаар хянаж хэрэгжүүлдэг.

MLS орчнуудад "clearance" буюу цэвэрлэгээ түвшин нь субьект болон обьектуудын хаяг/шошгонд тасалгаануудын цуг тохируулагддаг. Эдгээр цэвэрлэгээ буюу мэдрэхүйн түвшингүүд нь зургаан мянгаас их тоонд хүрч болох учир ямар ч администраторын хувьд

субьект эсвэл обьект бүрийг нарийн тохируулах нь сүрдмээр ажил байдаг. Харин үүнийг хөнгөвчлөх гурван ширхэг "хормын" хаяг/шошго энэ бодлогод орсон байдаг.

Эдгээр хаяг/шошгонууд нь `mls/low`, `mls/equal` болон `mls/high` юм. Эдгээр хаяг/шошгонууд нь гарын авлагын хуудсанд дэлгэрэнгүй тайлбарлагдсан болохоор энд зөвхөн товчхон тайлбарлая:

- `mls/low` хаяг/шошго нь доод тохиргоог агуулдаг бөгөөд энэ нь түүнийг бусад бүх обьектуудаар захируулахыг зөвшөөрдөг. `mls/low`-ээр хаяглагдсан болгон доод цэвэрлэгээний түвшинтэй байх бөгөөд өндөр түвшний мэдээлэлд хандах нь зөвшөөрөгдөөгүй байх болно. Мөн энэ хаяг/шошго нь цэвэрлэгээний өндөр түвшингийн обьектуудад бичих эсвэл тэдгээрт мэдээлэл дамжуулахаас сэргийлдэг.
- `mls/equal` хаяг/шошго энэ бодлогоос чөлөөлөгдөхөөр болсон обьектуудад тавигдах ёстой.
- `mls/high` хаяг/шошго нь цэвэрлэгээний боломжит хамгийн өндөр түвшин юм. Энэ хаяг/шошгыг заасан обьектууд систем дэх бусад бүх обьектуудаас давуу эрхтэй байх бөгөөд гэхдээ тэдгээр нь доод ангиллын обьектуудад мэдээлэл алдагдахыг зөвшөөрөхгүй байх болно.

MLS дараах боломжуудыг олгодог:

- Шатлаагүй зэрэглэлүүдийн олонлогтой аюулгүй байдлын шаталсан түвшин;
- Тогтмол дүрмүүд: дээш уншихгүй, доош бичихгүй (субьект нь өөрөөсөө дээд түвшинд биш зөвхөн өөрийн түвшний болон доод түвшний обьектуудад унших хандалттай байж болно. Үүнтэй адилаар субьект нь өөрөөсөө доод түвшинд биш зөвхөн өөрийн түвшний болон дээд түвшний обьектуудад бичих хандалттай байж болно.);
- Нууцлаг байдал (өгөгдлийн зохисгүй ил болголтоос сэргийлэх);
- Мэдрэмжийн олон түвшингүүдэд өгөгдөлтэй зэрэгцээгээр ажиллах системүүдийн дизайны үндэс (нууц болон итгэмжлэгдсэн мэдээллийн хооронд мэдээлэл алдахгүйгээр).

Тусгай төхөөрөмжүүд болон интерфэйсүүдийн хувьд дараах `sysctl`-ийн тааруулах боломжтой хувьсагчууд байдаг:

- `security.mac.mls.enabled` нь MLS бодлогыг нээх/хаахад хэрэглэгддэг.
- `security.mac.mls.ptys_equal` нь бүх pty(4) төхөөрөмжүүдийг үүсгэлтийнх нь үеэр `mls/equal` гэж хаяглана.
- `security.mac.mls.revocation_enabled` нь обьектуудын хаяг/шошго доод зэргийнх уруу болж өөрчлөгдсөний дараа тэдгээрт хандах хандалтыг цуцлахад хэрэглэгддэг.
- `security.mac.mls.max_compartments` нь обьектуудад хамгийн их тооны тасалгааны түвшингүүдийг тохируулахад хэрэглэгддэг; үндсэндээ системд зөвшөөрөгдсөн тасалгааны хамгийн их дугаар байна.

MLS хаяг/шошгонуудтай ажиллахын тулд setfmac(8) байдаг. Обьектод хаяг/шошгыг олгохын тулд дараах тушаалыг ажиллуулна:

```
# setfmac mls/5 test
```

test файлын хувьд MLS хаяг/шошгыг авахын тулд дараах тушаалыг ажиллуулна:

```
# getfmac test
```

Энэ нь MLS бодлогын боломжуудын товч дүгнэлт юм. Өөр нэг хандлага нь MLS бодлогын мэдээллийг тохируулах мастер бодлогын файлыг /etc санд үүсгэж тэр файлыг `setfmac` тушаалд өгөх явдал юм. Энэ аргыг бүх бодлогуудыг авч үзсэнийхээ дараа тайлбарлах болно.

### 17.12.1. Албадмал Мэдрэмжийг төлөвлөх нь

Олон түвшинт аюулгүй байдлын бодлогын модулиар администратор эмзэг мэдээллийн урсгалыг хянахын тулд төлөвлөдөг. Анхдагчаар өөрийн блок дээш унших, блок доош бичих мөн чанараараа систем бүгдийг доод төлөвт болгодог. Бүгд хандах боломжтой байх бөгөөд администратор тохиргооны явцад аажмаар үүнийг мэдээллийн итгэмжлэгдсэн байдлыг нэмэгдүүлэн өөрчилдөг.

Дээрх гурван үндсэн хаяг/шошгоноос гадна администратор хэрэглэгчид болон бүлгүүдийг шаардлагын дагуу тэдгээрийн хоорондмэдээллийн урсгалыг хаахаар бүлэглэж болно. Цэвэрлэгээний түвшингүүдэд мэдээллийг танигдсан үгсээр хайх нь амар байж болох бөгөөд жишээ нь `Confidential`, `Secret`, болон `Top Secret` гэх зэрэг ангиллууд байж болох юм. Зарим администраторууд төслийн түвшингүүд дээр үндэслэн өөр бүлгүүдийг үүсгэж болох юм. Ангиллын аргаас үл хамааран ийм хязгаарласан бодлогыг хийхээс өмнө сайн бодож гаргасан төлөвлөгөө байж байх ёстой.

Энэ аюулгүй байдлын бодлогын модулийн хувьд зарим жишээ тохиолдлууд гэх юм бол e-commerce вэб сервер, компанийн чухал мэдээлэл болон санхүүгийн байгууллагын орчнуудыг агуулсан файл сервер байж болох юм. Хамгийн үнэмшилгүй газар бол зөвхөн хоёр, гуравхан хэрэглэгчтэй ажлын станц байх юм.

## 17.13. MAC Biba модуль

Модулийн нэр: mac_biba.ko

Цөмийн тохиргооны мөр: `options MAC_BIBA`

Ачаалалтын тохируулга: `mac_biba_load="YES"`

mac_biba(4) модуль MAC Biba бодлогыг дууддаг. Энэ бодлого нь MLS бодлоготой адил ажилладаг бөгөөд ялгаатай нь мэдээллийн урсгалын дүрмүүд нь нэлээн эсрэгээр байдаг. Энэ нь эмзэг мэдээллийн буурсан урсгалаас сэргийлдэг гэдэг бол MLS бодлого нь эмзэг мэдээллийн өгссөн урсгалаас сэргийлдэг; тиймээс энэ хэсгийн ихэнх нь хоёр бодлогод хоёуланд нь хамаатай юм.

Biba орчнуудад "integrity" буюу бүрэн бүтэн байдлын хаяг/шошго субьект эсвэл обьект бүр дээр тавигддаг. Эдгээр хаяг/шошгууд нь шаталсан зэргүүд болон шатлаагүй бүрэлдэхүүнүүдээс тогтдог. Обьект болон субьектийн зэрэг өсөх тусам бүрэн бүтэн байдал ч бас дээшилдэг.

Дэмжигдсэн хаяг/шошгууд нь `biba/low`, `biba/equal`, болон `biba/high` бөгөөд доор тайлбарлав:

- `biba/low` хаяг/шошго нь обьект эсвэл субьектийн авч болох хамгийн доод бүрэн бүтэн байдал гэж үздэг. Үүнийг обьектууд эсвэл субьектууд дээр тавих нь илүү өндрөөр тэмдэглэгдсэн обьектууд эсвэл субьектууд уруу хийх тэдгээрийн бичих хандалтыг хаана. Гэхдээ тэдгээрт унших хандалт байх болно.
- `biba/equal` хаяг/шошго нь бодлогоос чөлөөлөгдөх обьектууд дээр зөвхөн тавигдах ёстой.
- `biba/high` хаяг/шошго нь доод хаяг/шошго дээр тавигдсан обьектуудад бичихийг зөвшөөрөх боловч тэр обьектийг уншихыг зөвшөөрдөггүй. Бүхэл системийн бүрэн бүтэн байдалд нөлөөлдөг обьектуудад энэ хаяг/шошгыг тавихыг зөвлөдөг.

Biba дараах боломжуудыг олгодог:

- Шатлаагүй бүрэн бүтэн байдлын зэрэглэлүүдийн олонлог бүхий шаталсан бүрэн бүтэн байдлын түвшин;
- Тогтмол дүрмүүд: дээш бичихгүй, доош уншихгүй (MLS-ийн эсрэг). Субьект нь өөрөөсөө дээд түвшинд биш зөвхөн өөрийн түвшний болон доод түвшний обьектуудад бичих хандалттай байж болно. Үүнтэй адилаар субьект нь өөрөөсөө доод түвшинд биш зөвхөн өөрийн түвшний болон дээд түвшний обьектуудад унших хандалттай байж болно;
- Бүрэн бүтэн байдал (өгөгдлийн зохисгүй өөрчлөлтөөс сэргийлэх);
- Бүрэн бүтэн байдлын түвшингүүд (MLS-ийн мэдрэмжийн түвшингүүдийн оронд).

Дараах `sysctl`-ийн тааруулах боломжтой хувьсагчуудыг Biba бодлоготой ажиллахын тулд хэрэглэж болно.

- `security.mac.biba.enabled` нь машин дээр Biba бодлогыг нээхэд/хаахад хэрэглэгдэж болно.
- `security.mac.biba.ptys_equal` нь Biba бодлогыг pty(4) төхөөрөмжүүд дээр хаахад хэрэглэглэгдэж болно.
- `security.mac.biba.revocation_enabled` нь хаяг/шошго субьектийг захирахаар өөрчлөгдсөн бол обьектод хийх хандалтыг цуцлах болно.

Системийн обьектууд дахь Biba бодлогын тохиргоонд хандахын тулд `setfmac` болон `getfmac` тушаалуудыг ашиглана:

```
# setfmac biba/low test
# getfmac test
test: biba/low
```

### 17.13.1. Албадмал бүрэн бүтэн байдлыг төлөвлөх нь

Бүрэн бүтэн байдал нь мэдрэмтгий байдлаас өөр бөгөөд мэдээллийг итгэгдээгүй талуудаар хэзээ ч удирдуулахгүй байлгаж баталгаажуулдаг. Үүнд субьектууд болон обьектууд, тэдгээрийн хооронд дамжих мэдээлэл ордог. Энэ нь хэрэглэгчдэд зөвхөн өөрчилж чадах боломж болон бүр зарим тохиолдолд тэдэнд хэрэгтэй мэдээлэлд хандах боломжийг олгодог.

mac_biba(4) аюулгүй байдлын бодлогын модуль нь аль файлууд болон програмуудыг

хэрэглэгч эсвэл хэрэглэгчид харах ёстойг заахыг администраторт зөвшөөрч програмууд болон файлууд нь аюул заналаас ангид бөгөөд тэр хэрэглэгч, эсвэл хэрэглэгчдийн бүлгийн хувьд системээр итгэгдсэн гэдгийг баталгаажуулж дууддаг.

Эхний төлөвлөлтийн үеэр администратор зэргүүд, түвшингүүд болон бүсүүдэд хэрэглэгчдийг хуваахад бэлдэх ёстой. Хэрэглэгчдийн хувьд зөвхөн өгөгдлөөс гадна бас програмууд болон хэрэгслүүдэд тэдгээрийг эхлэхээс өмнө болон тэдгээрийг эхлүүлсний дараа тэдгээрт хандах хандалт хаагдсан байх болно. Энэ бодлогын модуль идэвхжүүлэгдсэний дараа систем өндөр хаяг/шошго уруу анхдагчаар шилжих бөгөөд хэрэглэгчдийн хувьд өөр зэргүүд болон түвшингүүдийг тохируулах нь администраторын хэрэг юм. Цэвэрлэгээний түвшингүүдийг дээр тайлбарласны дагуу ашиглахын оронд сайн төлөвлөх арга нь сэдвүүдийг оруулж болох юм. Жишээ нь эх кодын архив, эх код эмхэтгэгч болон бусад хөгжүүлэлтийн хэрэгслүүдэд өөрчлөх хандалтыг зөвхөн хөгжүүлэгчдэд зөвшөөрөх байж болно. Тэгээд бусад хэрэглэгчдийг тест хийгчид, дизайн хийгчид эсвэл зүгээр л энгийн хэрэглэгчид зэрэг өөр зэрэглэлд бүлэглэж зөвхөн унших хандалтыг зөвшөөрөх юм.

Цаанаасаа хийгдсэн аюулгүй байдлын хяналтаас болоод доод түвшний бүрэн бүтэн байдлын субьект нь дээд түвшний бүрэн бүтэн байдлын субьект уруу бичиж чаддаггүй; дээд түвшний бүрэн бүтэн байдлын субьект нь доод түвшний бүрэн бүтэн байдлын обьектийг ажиглаж эсвэл уншиж чаддаггүй. Хамгийн доод боломжит зэрэгт хаяг/шошгыг тохируулах нь субьектуудыг түүнд хандах боломжгүй болгож болох юм. Энэ аюулгүй байдлын бодлогын модулийн зарим хэтийн орчнуудад хүчилсэн вэб сервэр, хөгжүүлэлтийн болон тестийн машин, болон эх кодын архив зэрэг орж болох юм. Тийм ч ашигтай бус шийдэлд персонал ажлын станц, чиглүүлэгч маягаар ашиглагдаж байгаа машин эсвэл сүлжээний галт хана зэрэг байж болох юм.

# 17.14. MAC LOMAC модуль

Модулийн нэр: mac_lomac.ko

Цөмийн тохиргооны файл: `options MAC_LOMAC`

Ачаалалтын тохируулга: `mac_lomac_load="YES"`

MAC Biba бодлогоос ондоо нь mac_lomac(4) бодлого нь бүрэн бүтэн байдлын дүрмүүдийг эвдэхгүйн тулд бүрэн бүтэн байдлын түвшинг заавал багасгасны дараа бүрэн бүтэн байдлын хувьд доор орших обьект уруу хандахыг зөвшөөрдөг.

Low-watermark integrity policy буюу доод түвшний бүрэн бүтэн байдлын MAC хувилбарыг хуучин lomac(4)-ийн шийдэлтэй эндүүрч болохгүй бөгөөд энэ хувилбар нь Biba-тай бараг л төстэй ажилладаг боловч ялгаатай тал нь субьектийн бууруулалтыг туслах зэргийн тасалгааны тусламжтай дэмжихийн тулд хөвөгч хаяг/шошгуудыг ашигладаг явдал юм. Энэ хоёр дахь тасалгаа нь `[auxgrade]` хэлбэрийг авдаг. lomac бодлогыг туслах зэргээр зааж өгөх үед энэ нь иймэрхүү харагдах ёстой: `lomac/10[2]`. Энд байгаа хоёр (2) гэсэн тоо нь туслах зэрэг юм.

MAC LOMAC бодлого нь бүрэн бүтэн байдлын хаяг/шошгоор бүх системийн обьектуудыг хаа сайгүй хаяглах явдалд тулгуурладаг бөгөөд субьектуудад бүрэн бүтэн байдлын хувьд доор

орших обьектуудаас уншихыг зөвшөөрч дараа нь өндөр бүрэн бүтэн байдал бүхий обьектуудад ирээдүйд хийгдэж болзошгүй бичилтүүдээс урьдчилан сэргийлэхийн тулд субьект дээрх хаяг/шошгыг доошлуулж бууруулдаг. Энэ нь дээр хэлэлцэгдсэн `[auxgrade]` тохируулга болохоор уг бодлого нь илүү сайн нийцтэй байдлыг хангаж Biba-аас бага эхний тохиргоог шаардаж болох юм.

### 17.14.1. Жишээнүүд

Biba болон MLS бодлогуудын нэгэн адил `setfmac` болон `setpmac` хэрэгслүүд системийн обьектууд дээр хаяг/шошгонууд байрлуулахад хэрэглэгдэж болно:

```
# setfmac /usr/home/trhodes lomac/high[low]
# getfmac /usr/home/trhodes lomac/high[low]
```

Энд байгаа туслах зэрэг нь `low` буюу доор гэж байгааг анзаараарай, энэ нь зөвхөн MAC LOMAC бодлогын хангадаг боломж юм.

## 17.15. MAC Шорон дахь Nagios

Дараах нь зөв тохируулсан бодлогуудын хамтаар төрөл бүрийн MAC модулиудыг ашиглан аюулгүй орчинг үүсгэхийг харуулах болно. Энэ нь зөвхөн тест бөгөөд хүн бүгдийн аюулгүй байдлын асуудалд бүрэн хариулт болно гэж тооцох ёсгүй юм. Бодлогыг зөвхөн шийдэж түүнийг орхигдуулах нь хэзээ ч ажиллахгүй бөгөөд жинхэнэ ажиллаж байгаа үйлдвэрлэлийн орчинд сүйрлийн болж болох юм.

Энэ процессийг эхлүүлэхээсээ өмнө `multilabel` тохируулга файлын систем бүр дээр энэ бүлгийн эхэнд дурдсаны дагуу тавигдах ёстой. Ингэж хийхгүй бол алдаа гарах болно. Энд байхдаа net-mngt/nagios-plugins, net-mngt/nagios, болон www/apache22 портууд бүгд суулгагдаж тохируулагдаж зөв ажиллаж байгаа эсэхийг шалгаарай.

### 17.15.1. Хэрэглэгчийн insecure ангилал үүсгэнэ

Дараах хэрэглэгчийн ангиллыг /etc/login.conf файлд нэмж:

```
insecure:\
:copyright=/etc/COPYRIGHT:\
:welcome=/etc/motd:\
:setenv=MAIL=/var/mail/$,BLOCKSIZE=K:\
:path=~/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/usr/local/bin
:manpath=/usr/shared/man /usr/local/man:\
:nologin=/usr/sbin/nologin:\
:cputime=1h30m:\
:datasize=8M:\
:vmemoryuse=100M:\
:stacksize=2M:\
:memorylocked=4M:\
:memoryuse=8M:\
```

```
:filesize=8M:\
:coredumpsize=8M:\
:openfiles=24:\
:maxproc=32:\
:priority=0:\
:requirehome:\
:passwordtime=91d:\
:umask=022:\
:ignoretime@:\
:label=biba/10(10-10):
```

мөн дараах мөрийг анхдагч хэрэглэгчийн ангилалд нэмж процедурыг эхэлнэ:

```
:label=biba/high:
```

Энэ хийгдсэний дараа мэдээллийн баазыг дахин бүтээхийн тулд дараах тушаалыг ажиллуулах ёстой:

```
# cap_mkdb /etc/login.conf
```

### 17.15.2. Ачаалалтын тохиргоо

Дахин ачаалах гэсний хэрэггүй, шаардлагатай модулиудыг систем эхлүүлэхэд дуудахын тулд дараах мөрүүдийг /boot/loader.conf файлд нэмнэ:

```
mac_biba_load="YES"
mac_seeotheruids_load="YES"
```

### 17.15.3. Хэрэглэгчдийг тохируулна

`root` хэрэглэгчийг анхдагч ангилалд доор дурдсаныг ашиглан тохируулна:

```
# pw usermod root -L default
```

`root` эсвэл системийн хэрэглэгчид биш бүх хэрэглэгчийн бүртгэлүүд одоо нэвтрэлийн ангилал шаардах болно. Нэвтрэлтийн ангилал шаардлагатай, түүнгүй бол хэрэглэгчид vi(1) зэрэг нийтлэг тушаалд хандах боломжгүй болно. Дараах `sh` скрипт үүнийг хийх болно:

```
# for x in `awk -F: '($3 >= 1001) && ($3 != 65534) { print $1 }' \
    /etc/passwd`; do pw usermod $x -L default; done;
```

`nagios` болон `www` хэрэглэгчдийг insecure ангилалд оруулна:

```
# pw usermod nagios -L insecure
```

```
# pw usermod www -L insecure
```

### 17.15.4. Contexts буюу Сэдвийн файл үүсгэнэ

Сэдвийн файл нь одоо үүсгэгдсэн байх ёстой; дараах жишээ файлыг /etc/policy.contexts-д байрлуулах ёстой.

```
# This is the default BIBA policy for this system.

# System:
/var/run                        biba/equal
/var/run/*                      biba/equal

/dev                            biba/equal
/dev/*                          biba/equal

/var                biba/equal
/var/spool                      biba/equal
/var/spool/*                    biba/equal

/var/log                        biba/equal
/var/log/*                      biba/equal

/tmp                biba/equal
/tmp/*              biba/equal
/var/tmp            biba/equal
/var/tmp/*          biba/equal

/var/spool/mqueue       biba/equal
/var/spool/clientmqueue     biba/equal

# For Nagios:
/usr/local/etc/nagios
/usr/local/etc/nagios/*         biba/10

/var/spool/nagios               biba/10
/var/spool/nagios/*             biba/10

# For apache
/usr/local/etc/apache           biba/10
/usr/local/etc/apache/*         biba/10
```

Энэ бодлого нь мэдээллийн урсгалд хязгаарлалтуудыг тавьж аюулгүй байдлыг хангадаг. Энэ тусгайлсан тохиргооны хувьд хэрэглэгчид, `root` болон бусад хэрэглэгчид Nagios программд хандахаар хэзээ ч зөвшөөрөгдсөн байх ёсгүй. Nagios-ийн тохиргооны файлууд болон

процессууд нь бүр мөсөн өөртөө багтсан буюу шоронд хийгдсэн байх болно.

Одоо энэ файлыг өөрийн систем уруу уншуулахдаа дараах тушаалыг ажиллуулна:

```
# setfsmac -ef /etc/policy.contexts /
# setfsmac -ef /etc/policy.contexts /
```

Дээрх файлын системийн байршил орчноосоо хамааран өөр байж болно; гэхдээ үүнийг файлын систем бүр дээр ажиллуулах ёстой.

/etc/mac.conf файл гол хэсэгт дараах өөрчлөлтүүдийг шаарддаг:

```
default_labels file ?biba
default_labels ifnet ?biba
default_labels process ?biba
default_labels socket ?biba
```

### 17.15.5. Сүлжээг идэвхжүүлнэ

Дараах мөрийг /boot/loader.conf-д нэмнэ:

```
security.mac.biba.trust_all_interfaces=1
```

Тэгээд дараа нь доор дурдсаныг rc.conf файлд хадгалагдсан сүлжээний картны тохиргоонд нэмнэ. Хэрэв анхдагч Интернэтийн тохиргоо DHCP-ээр хийгдсэн бол системийг ачаалах болгоны дараа үүнийг гараараа тохируулах хэрэгтэй болох юм:

```
maclabel biba/equal
```

### 17.15.6. Тохиргоог тест хийх нь

Вэб сервер болон Nagios нь системийг эхлүүлэхэд ажиллахааргүй байгаа эсэхийг шалгаад дахин ачаална. `root` хэрэглэгч Nagios-ийн тохиргооны сан дахь ямар ч файлд хандаж чадах ёсгүйг баталгаажуулна. Хэрэв `root` нь /var/spool/nagios-д ls(1)-ийг ажиллуулж чадаж байвал ямар нэг юм буруу байна гэсэн үг. Зөв бол "permission denied" алдаа буцаагдах ёстой.

Хэрэв бүгд зүгээр юм шиг санагдвал Nagios, Apache, болон Sendmail-ийг одоо аюулгүй байдлын бодлогод тааруулж ажиллуулж болно. Үүнийг дараах тушаал хийх болно:

```
# cd /etc/mail && make stop && \
setpmac biba/equal make start && setpmac biba/10\(10-10\) apachectl start && \
setpmac biba/10\(10-10\) /usr/local/etc/rc.d/nagios.sh forcestart
```

Бүгд зөв ажиллаж байгаа эсэхийг баталгаажуулж дахин шалгаарай. Хэрэв үгүй бол

бүртгэлийн файлуудаас алдааны мэдэгдлүүд байгаа эсэхийг шалгана. sysctl(8) хэрэгсэл ашиглаж mac_biba(4) аюулгүй байдлын бодлогын модулийн үйлчлэлийг хааж бүгдийг эхнээс нь эхлэхийг оролдоорой.

`root` хэрэглэгч аюулгүй байдлын үйлчлэлийг өөрчилж тохиргооны файлыг айлгүйгээр засварлаж чадна. Дараах тушаал нь шинээр үүсгэсэн бүрхүүлийн хувьд аюулгүй байдлын бодлогыг доод зэрэг уруу орж буурахыг зөвшөөрөх болно:

```
# setpmac biba/10 csh
```

Үүнийг болгохгүй байлгахын тулд login.conf(5)-оор хэрэглэгчийг хүрээнд оруулна. Хэрэв setpmac(8) тушаалыг тасалгааных нь хүрээнээс гадна ажиллуулах гэж оролдвол алдаа буцаагдах бөгөөд тушаал ажиллахгүй байх болно. Энэ тохиолдолд root-ийг `biba/high(high-high)` болгож тохируулна.

## 17.16. Хэрэглэгчийг түгжих

Энэ жишээ нь харьцангуй жижиг, тавиас бага хэрэглэгчтэй хадгалалтын системийг авч үздэг. Хэрэглэгчид нь нэвтрэлтийн боломжуудтай байх бөгөөд тэдэнд зөвхөн өгөгдөл биш бас хандалтын эх үүсвэрүүдийг хадгалахыг зөвшөөрнө.

Энэ тохиолдолд mac_bsdextended(4) нь mac_seeotheruids(4)-тэй холилдон оршиж болох бөгөөд системийн обьектуудад хандахыг хаагаад зогсохгүй бас хэрэглэгчийн процессийг нуух хандалтыг бас хаадаг.

Дараах мөрийг /boot/loader.conf файлд нэмж эхэлнэ:

```
mac_seeotheruids_load="YES"
```

mac_bsdextended(4) аюулгүй байдлын бодлогын модулийг дараах rc.conf хувьсагчийг хэрэглэн идэвхтэй болгож болно:

```
ugidfw_enable="YES"
```

/etc/rc.bsdextended файлд хадгалагдах анхдагч дүрмүүд нь системийг эхлүүлэхэд дуудагдана. Гэхдээ анхдагч оруулгууд нь өөрчлөлтүүд шаардаж болох юм. Энэ машин нь зөвхөн хэрэглэгчдэд үйлчлэхээр зориулагдсан болохоор сүүлийн хоёроос бусдыг хааж тайлбар болгон үлдээж болох юм. Сүүлийн хоёр нь анхдагчаар хэрэглэгчийн эзэмших системийн обьектуудыг дуудуулах болно.

Шаардлагатай хэрэглэгчдийг энэ машин уруу нэмээд дахин ачаална. Тест хийх зорилгоор хоёр консол дээр өөр хэрэглэгчээр нэвтрэхийг оролдоорой. Бусад хэрэглэгчдийн процессууд харж болохоор байгаа эсэхийг харахын тулд `ps aux` тушаалыг ажиллуулна. ls(1)-ийг нөгөө хэрэглэгчийн гэрийн сан дээр ажиллуулахыг оролдоорой, энэ нь амжилтгүй болох болно.

Супер хэрэглэгчийн хандалтыг хаахын тулд ашигладаг тусгай `sysctl`-уудыг өөрчлөхөөс бусад тохиолдолд `root` хэрэглэгчээр тест битгий хийгээрэй.

Шинэ хэрэглэгч нэмэгдэхэд тэдгээрийн mac_bsdextended(4) дүрмүүд дүрмийн олонлогийн жагсаалтад байхгүй байна. Дүрмийн олонлогийг хурдан шинэчлэхийн тулд kldunload(8) болон kldload(8) хэрэгслүүдийг ашиглан аюулгүй байдлын бодлогын модулийг буулгаж дараа нь түүнийг дахин ачаалж хийнэ.

# 17.17. MAC Тогтолцооны алдааг олж засварлах

Хөгжүүлэлтийн явцад цөөн хэрэглэгчид энгийн тохируулга дээр асуудлууд гарснаа мэдээлсэн. Эдгээр асуудлуудын заримыг доор жагсаав:

## 17.17.1. `multilabel` тохируулгыг / дээр идэвхжүүлж болохгүй байна

`multilabel` туг миний root (/) хуваалтан дээр идэвхтэй болохгүй байна!

50 хэрэглэгч тутмын нэг нь ийм асуудалтай байдаг бололтой, харин бид энэ асуудалтай эхний тохиргооны үеэр тулгарсан. "bug" буюу "цох" гэж нэрлэгдэх үүний цаадах ажиглалт нь үүнийг буруу баримтжуулалт эсвэл баримтын буруу тайлбарлалтын үр дүн гэж намайг итгэхэд хүргэсэн. Энэ яагаад болсноос үл хамааран үүнийг шийдэхийн тулд дараах алхмуудыг хийж болох юм:

1. /etc/fstab-ийг засварлаж root хуваалтыг зөвхөн унших зорилгоор `ro` гэж тохируулна.
2. Ганц хэрэглэгчийн горимд дахин ачаална.
3. `tunefs -l enable` тушаалыг / дээр ажиллуулна.
4. Системийг энгийн горимд дахин ачаална.
5. `mount -urw`/ тушаалыг ажиллуулж `ro` тохируулгыг `rw` болгож /etc/fstab файлд өөрчлөн системийг дахин ачаална.
6. root файлын систем дээр `multilabel` тохируулга зөв тохируулагдсаныг баталгаажуулж `mount` тушаалын гаралтыг дахин шалгаарай.

## 17.17.2. MAC-ийн дараа X11 серверийг эхлүүлж чадахгүй байна

MAC-ийн тусламжтай аюулгүй орчинг үүсгэсний дараа би X-ийг дахиж эхлүүлж чадахаа больчихлоо!

Энэ нь MAC `хуваалт`ын бодлого эсвэл MAC хаяглалтын бодлогуудын аль нэгний буруу хаяглалтаас болсон байж болох юм. Дибаг хийхийн тулд доор дурдсаныг оролдоод үзээрэй:

1. Алдааны мэдэгдлийг шалгана; хэрэв хэрэглэгч `insecure` ангилалд байгаа бол `хуваалт`ын бодлого гэмтэн байж болох юм. Хэрэглэгчийн ангиллыг `default` буюу анхдагч ангилал уруу тохируулж мэдээллийн баазыг `cap_mkdb` тушаалын тусламжтай

дахин бүтээх хэрэгтэй. Хэрэв энэ нь асуудлыг арилгаж чадахгүй байгаа бол хоёрдугаар алхам уруу ор.

2. Хаяг/шошгоны бодлогуудыг давхар шалгаарай. Асуудалтай байгаа хэрэглэгч, X11 програм болон /dev оруулгуудын хувьд бодлогууд зөв заагдсан эсэхийг баталгаажуулаарай.
3. Хэрэв эдгээрийн аль нь ч асуудлыг тань шийдэхгүй бол TrustedBSD вэб сайтад байрлах TrustedBSD-ийн хэлэлцүүлгийн жагсаалтууд эсвэл FreeBSD ерөнхий асуултууд захидлын жагсаалт захидлын жагсаалт уруу алдааны мэдэгдэл болон өөрийн орчны тухай мэдээллийг илгээгээрэй.

### 17.17.3. Error: _secure_path(3) cannot stat .login_conf

Намайг `root` хэрэглэгчээс систем дээрх өөр хэрэглэгч уруу шилжихийг оролдох үед `_secure_path: unable to state .login_conf` гэсэн алдаа гараад байна.

Энэ мэдэгдэл нь тэр болох гэж байгаа хэрэглэгчийн хаяг/шошгоны тохиргооноос хэрэглэгчийн өөрийнх нь тохиргоо өндөр байгааг ихэвчлэн үзүүлдэг. Жишээ нь систем дээрх хэрэглэгч `joe` анхдагч `biba/low` гэсэн хаяг/шошготой байна гэж бодъё. `biba/high` хаяг/шошготой `root` хэрэглэгч `joe`-ийн гэр санг харж чадахгүй. Энэ нь `root` хэрэглэгч `su` тушаал ашиглан `joe` болсон ч гэсэн болохгүй байна. Энэ тохиолдолд Biba бүрэн бүтэн байдлын загвар нь `root` хэрэглэгчийг бүрэн бүтэн байдлын доод түвшин тохируулагдсан обьектуудыг харахыг зөвшөөрөхгүй байх болно.

### 17.17.4. `root` хэрэглэгчийн нэр эвдэрсэн байна!

Энгийн эсвэл бүр ганц хэрэглэгчийн горимд `root` хэрэглэгч танигддаггүй. `whoami` тушаал 0 (тэг) буцаах бөгөөд `su` тушаал `who are you?` гэсэн алдааны мэдэгдлийг буцаадаг. Юу болоод байгаа юм бол оо?

Энэ нь хаяглах бодлого sysctl(8)-оор хаагдсан эсвэл бодлогын модулийг буулгаснаас болдог. Хэрэв бодлого хаагдсан эсвэл түр зуур хаагдсан бол `label` тохируулгыг арилган нэвтрэлтийн боломжуудын мэдээллийн баазыг дахин тохируулах хэрэгтэй. Бүх `label` тохируулгууд арилсан эсэхийг баталгаажуулж login.conf файлаа дахин шалгаж мэдээллийн баазаа `cap_mkdb` тушаалаар дахин бүтээх хэрэгтэй.

Энэ нь master.passwd файлд эсвэл мэдээллийн баазад хандах хандалтыг бодлого хязгаарласнаас болоод бас гарч болох юм. Системд ашиглагдаж байгаа ерөнхий бодлоготой зөрчилдөх хаяг/шошгоны доор администратор файлыг өөрчлөхөд ихэвчлэн ингэдэг. Ийм тохиолдлуудад хэрэглэгчийн мэдээллийг систем унших бөгөөд файл нь шинэ хаяг/шошго удамшин авсан болохоор хандалт хаалттай байх болно. Бодлогыг sysctl(8)-ий тусламжтай хаах хэрэгтэй. Ингэхэд бүх зүйлс хэвийндээ эргэн орох болно.

# Chapter 18. Аюулгүй байдлын үйл явцад аудит хийх нь

## 18.1. Ерөнхий агуулга

FreeBSD үйлдлийн системд нягт-боловсруулсан, аюулгүй байдлын үйл явцад аудит хийх дэмжлэг орсон байгаа. Үйл явцад аудит хийх нь нэвтрэн оролтууд, тохиргооны өөрчлөлтүүд, болон файл болон сүлжээний хандалт зэрэг төрөл бүрийн аюулгүй байдлын холбогдолтой системийн үйл явцуудын хянаж бүртгэсэн бичлэгийг найдвартай, нягт-боловсруулсан, ба тохируулах боломжтойгоор хийх боломжийг бүрдүүлдэг. Эдгээр хянаж бүртгэсэн бичлэгүүд нь системийг шууд хянах, халдлага илрүүлэх, болон халдлагын дараах анализ хийхэд үнэлж баршгүй байж болох юм. FreeBSD Sun™-ий гаргасан BSM API болон файлын хэлбэрийг шийдлээ болгосон бөгөөд Sun™-ий Solaris болон Apple®-ийн Mac OS® X-ийн аудит шийдлүүдтэй харилцан ажиллах боломжтой.

Энэ бүлэг нь үйл явцыг аудит хийхийг тохируулах болон суулгах тал дээр анхаарна. Энд аудитийн бодлогуудын талаар тайлбарлах бөгөөд аудитийн тохиргооны жишээг бас харж болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Үйл явцыг аудит хийх гэж юу вэ болон яаж ажилладаг талаар.
- Хэрэглэгчид болон процессуудын үйл явцыг аудит хийхийг FreeBSD дээр яаж тохируулах талаар.
- Аудитийн мөрийг аудит багасгах болон хянах хэрэгслүүдээр хэрхэн шалгах талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- UNIX® болон FreeBSD-ийн үндсийг ойлгох (Юниксийн үндэс).
- Цөмийг тохируулах/хөрвүүлэх үндэстэй танилцах (FreeBSD цөмийг тохируулах нь).
- Аюулгүй байдлын талаар болон түүний FreeBSD -тэй хэрхэн холбогддог талаар ойлголттой байх (Аюулгүй байдал).

Аудит хэрэгсэл нь зарим нэг хязгаарлагдмал боломжуудтай бөгөөд тэдгээрээс дурдвал бүх аюулгүй байдлын холбогдолтой системийн үйл явцуудыг аудит хийх боломжгүй, X11 дээр үндэслэгдсэн дэлгэцийн менежерүүд болон гуравдагч дэмонууд зэрэг зарим нэвтрэх механизмууд хэрэглэгчийн нэвтрэх сессийг аудит хийх тохиргоог буруу хийдэг зэрэг болно.

Аюулгүй байдлын үйл явцыг аудит хийх хэрэгсэл нь системийн ажиллагааны маш дэлгэрэнгүй бичлэгүүдийг үүсгэх чадвартай: нарийвчилсан тохиргоо хийгдсэн завгүй системд мөр бичлэгийн файлын өгөгдөл тохиргооноос хамааран зарим тохиолдолд гигабайтаас ч илүү асар их болох боломжтой. Администраторууд их хэмжээний аудит хийх тохиргоотой холбоотой дискний хэмжээний шаардлагыг тооцох ёстой.

> Жишээ нь, аудит бичиж байгаа файлын систем дүүрэх нөхцөлд өөр бусад файлын системүүдэд хамааралгүйгээр файлын системийг /var/audit модонд зориулах нь магадгүй зохимжтой байж болох юм.

## 18.2. Энэ бүлгийн түлхүүр ухагдахуунууд

Энэ бүлгийг уншихаас өмнө аудиттай холбоотой цөөн түлхүүр ухагдахуунуудыг тайлбарлах шаардлагатай:

- *үйл явц*: Аудит хийх боломжтой үйл явц гэдэг нь аудит дэд системийн тусламжтайгаар хянаж бүртгэх боломжтой дурын үйл явцыг хэлнэ. Аюулгүй байдалтай холбоотой үйл явцуудын жишээнүүдэд файлын үүсгэлт, сүлжээний холболтыг босгох, эсвэл хэрэглэгчийн нэвтрэлтийг дурдаж болно. Үйл явцууд нь нэг бол жинхэнэ хэрэглэгч хүртэл мөрдөн гаргаж болох "шинж чанарлаг", эсвэл тэгж болохооргүй "шинж чанарлаг бус " байна. "Шинж чанарлаг бус" үйл явцуудын жишээнүүд гэвэл нууц үгийн буруу оролдлогууд гэх мэт нэвтрэн орох процессийн жинхэнэ эсэхийг шалгахаас өмнөх үеийн дурын үйл явцуудыг хэлж болно.
- *ангилал*: Үйл явцын ангиллууд гэдэг нь холбоотой үйл явцууд бүхий хэсэг бүлгүүдийн нэр бөгөөд эдгээр нь сонголтын илэрхийллүүдэд хэрэглэгддэг. Энгийн ашиглагддаг үйл явцуудын ангиллуудад "файл үүсгэлт" (fc), "ажиллуулах" (ex) мөн "нэвтрэх_гарах" (lo) зэрэг орно.
- *бичлэг*: Бичлэг гэдэг нь аюулгүй байдлын үйл явцыг тайлбарласан аудитийн хянан бүртгэсэн оруулга юм. Бичлэгүүд нь үйл явцын төрлийн бичлэг, субьектийн (хэрэглэгч) үйлдэл хийж байгаа тухай мэдээлэл, огноо болон цагийн мэдээлэл, дурын нэмэлт өгөгдлүүд болон обьектуудын мэдээлэл, амжилт эсвэл уналтын нөхцөлүүдийг агуулдаг.
- *мөр*: Аудитийн мөр буюу хянан бүртгэлийн файл нь аюулгүй байдлын үйл явцуудыг тайлбарласан аудит бичлэгүүдийн сериэс тогтоно. Ерөнхийдөө мөрүүд нь үйл явцуудын биелж дууссан цагийн дагуу он цагийн дарааллаар байрладаг. Зөвхөн жинхэнэ процессууд аудитийн мөрд бичлэг нэмэх эрхтэй байдаг.
- *сонголтын илэрхийлэл*: Сонголтын илэрхийлэл нь угтваруудын жагсаалт болон үйл явцтай тохирох аудитийн үйл явцын ангиллын нэрсээс тогтох мөр юм.
- *урьдчилсан сонголт*: Энэ нь администраторын сонирхох боломж бүхий үйл явцууд, сонирхлыг татахгүй байгаа үйл явц зэргийг тайлбарласан аудитийн бичлэгүүдийг үүсгэхээс зайлсхийх зорилготойгоор ялгасан системийн процесс юм. Урьдчилсан тохиргоо нь аль хэрэглэгчийн хувьд үйл явцуудын аль ангиллуудыг таних болон жинхэнэ болон жинхэнэ биш процессуудад хамаарах глобал тохиргоонуудын сонголтын илэрхийллүүдийн серийг ашигладаг.
- *хураангуйлалт*: Энэ нь байгаа аудитийн мөрнүүдээс аль бичлэгүүдийг хадгалалт, хэвлэлт болон анализ хийхээр сонгосон процесс юм. Үүний нэгэн адил аудит мөрөөс хэрэггүй аудит бичлэгүүдийг устгах процесс бас хамаарна. Хураангуйлалтыг ашиглаад администраторууд аудит өгөгдлийн хадгалалтын бодлогуудыг боловсруулах боломжтой юм. Жишээ нь, дэлгэрэнгүй аудитийн мөрнүүд нэг сарын хугацаанд хадгалагдаад дараагаар нь тэдгээр мөрнүүдийг архивын зорилгоор зөвхөн нэвтрэлтийн мэдээллийг үлдээн багасгаж болох юм.

## 18.3. Аудит хийх дэмжлэг суулгах нь

Үйл явцыг Аудит хийх хэрэглэгчийн талбарын дэмжлэг үндсэн FreeBSD үйлдлийн системд орсон байгаа. Үйл явцыг Аудит хийх дэмжлэг анхдагчаар эмхэтгэгдэн орсон боловч яг энэ боломжийг дэмжихийн тулд дараах мөрийг цөмд оруулан хөрвүүлсэн байх шаардлагатай:

```
options AUDIT
```

FreeBSD цөмийг тохируулах нь -д тайлбарласан ердийн процессийн дагуу цөмийг дахин хөрвүүлж суулгана.

Аудит идэвхжсэн цөм бүтээгдэж суулгагдаад систем дахин ачаалсны дараа дараах мөрийг rc.conf(5) -д нэмж аудит дэмонг идэвхжүүлнэ:

```
auditd_enable="YES"
```

Тэгээд системийг дахин ачаалах замаар эсвэл гараар аудит дэмонг ажиллуулах замаар аудит дэмжлэгийг эхлүүлэх ёстой:

```
service auditd start
```

## 18.4. Аудитийн тохиргоо

Аюулгүй байдлын аудитийн тохиргооны бүх файлуудыг /etc/security дотроос олж болно. Дараах файлууд аудит дэмон эхлэхээс өмнө байх ёстой:

- audit_class - Аудитийн ангиллуудын тодорхойлолтуудыг агуулна.
- audit_control - Анхдагч аудитийн ангиллууд, аудитийн хянан бүртгэлтийн эзлэхүүнд үлдээх хамгийн бага дискний зай, хамгийн их аудитийн мөрийн хэмжээ гэх зэрэг аудит дэд системийн шинж чанарыг хянана.
- audit_event - Системийн аудит үйл явцуудын тайлбарууд, нэрс болон үйл явц болгон аль ангилалд хамаарах жагсаалт.
- audit_user - Нэвтрэн орох үеийн глобал анхдагчуудаас бүрдсэн зөвхөн хэрэглэгчид хамааралтай аудитийн шаардлагууд
- audit_warn - Аудитийн бичлэгүүдийн зай хангалттүй болох эсвэл аудитийн мөрийн файл дахин эргэсэн зэрэг зайлшгүй шаардлагатай тохиолдолд анхааруулах мэдээллүүдийг үүсгэдэг өөрчлөх боломж бүхий auditd-ийн ашигладаг бүрхүүлийн скрипт.

Тохиргоон дахь алдаанууд үйл явцуудын буруу хянан бүртгэлд хүргэж болзошгүй тул аудитийн тохиргооны файлуудыг засварлах болон ажиллагааг хангахдаа заавал болгоомжтой байх шаардлагатай.

### 18.4.1. Үйл явц сонголтын илэрхийллүүд

Сонголтын илэрхийллүүд нь аль үйл явцуудыг аудит хийх ёстойг тодорхойлох аудитийн тохиргооны хэд хэдэн газар ашиглагддаг. Илэрхийллүүд нь тохирох үйл явцын ангиллуудын жагсаалтаас бүрдэх ба эдгээр тус бүр нь тохирох бичлэгүүдийг хүлээн авах ёстой юу эсвэл орхих ёстой юу гэдгийг харуулсан угтвартайгаас гадна оруулга нь амжилттай эсвэл амжилтгүй үйлдлүүдтэй тохирох ёстойг харуулах боломж бас байна. Сонголтын илэрхийллүүд нь зүүнээс баруун тийш биелэгддэг бөгөөд хоёр илэрхийллийг нэг дээр нь нөгөөг нь нэмж нийлүүлдэг.

Дараах жагсаалт нь audit_class-д байгаа анхдагч үйл явцын ангиллуудаас тогтоно:

- `all` - *all* - Бүх үйл явцын ангиллуудыг тааруулах(match).
- `ad` - *administrative* - Удирдлагын үйлдлүүд систем дээр бүхэлдээ гүйцэтгэгдэнэ.
- `ap` - *application* - Програмын тодорхойлсон үйлдэл.
- `cl` - *file close* - `close` системийн дуудлагыг аудит хийх.
- `ex` - *exec* - Програмын ажиллагааг аудит хийх. Тушаалын мөрийн нэмэлт өгөгдлүүд болон орчны хувьсагчуудыг `argv` ба `envv` параметрүүдийг ашиглан `policy` тохиргоонд тохиргоо хийн audit_control(5) -ийн тусламжтайгаар хянадаг.
- `fa` - *file attribute access* - stat(1), pathconf(2) болон бусад адил үйл явцуудын обьектийн шинж чанаруудад хандсан хандалтыг аудит хийх.
- `fc` - *file create* - Үр дүнд нь файл үүсдэг үйл явцуудыг аудит хийх.
- `fd` - *file delete* - Файлыг устгадаг үйл явцуудыг аудит хийх.
- `fm` - *file attribute modify* - chown(8), chflags(1), flock(2) зэрэг файлын шинж чанарын өөрчлөлт гарч байгаа үйл явцуудыг аудит хийх.
- `fr` - *file read* - Өгөгдөл уншигдаж байгаа, мөн файлуудыг уншихаар нээсэн зэрэг үйл явцуудыг аудит хийх.
- `fw` - *file write* - Өгөгдөл бичигдэж байгаа, мөн файлд бичсэн эсвэл файл өөрчлөгдсөн зэрэг үйл явцуудыг аудит хийх.
- `io` - *ioctl* - ioctl(2) системийн дуудлагын хэрэглээг аудит хийх.
- `ip` - *ipc* - POSIX хоолойнууд болон System V IPC үйлдлүүд зэрэг Процесс-Хоорондох Холбооны төрөл бүрийн хэлбэрүүдийг аудит хийх.
- `lo` - *login_logout* - Систем дээр болж байгаа login(1) ба logout(1) үйл явцуудыг аудит хийх.
- `na` - *non attributable* - Шинж чанаргүй үйл явцуудыг аудит хийх.
- `no` - *invalid class* - Аудит бус үйл явцуудыг тааруулах(match).
- `nt` - *network* - connect(2) ба accept(2) зэрэг сүлжээний үйлдлүүдтэй холбоотой үйл явцуудыг аудит хийх.
- `ot` - *other* - Бусад үйл явцуудыг аудит хийх.
- `pc` - *process* - exec(3) ба exit(3) зэрэг процессийн үйлдлүүдийг аудит хийх.

Эдгээр аудит үйл явцын ангиллуудыг audit_class болон audit_event тохиргооны файлуудыг өөрчилснөөр өөрчилж болно.

Жагсаалтад байгаа аудитийн ангилал бүр амжилттай/амжилтгүй үйлдлүүдийг таарсан эсэхийг болон ангилал ба төрлийн хувьд таарч байгааг нэмж байгаа эсвэл устгаж байгааг харуулсан угтвартай байна.

- (none) Үйл явцын амжилттай болон амжилтгүйг аудит хийх.
- `+` Энэ ангилал дахь амжилттай үйл явцуудыг аудит хийх.
- `-` Энэ ангилал дахь амжилтгүй үйл явцуудыг аудит хийх.
- `^` Энэ ангилал дахь амжилттай, амжилтгүй аль нь ч биш үйл явцуудыг аудит хийх.
- `^+` Энэ ангилал дахь амжилттай үйл явцуудыг аудит хийхгүй.
- `^-` Энэ ангилал дахь амжилтгүй үйл явцуудыг аудит хийхгүй.

Дараах сонголтын мөр амжилттай ба амжилтгүй нэвтрэлт/гаралтын үйл явцуудаас гадна зөвхөн амжилттай ажиллуулсныг сонгож байна:

```
lo,+ex
```

## 18.4.2. Тохиргооны файлууд

Аудит системийг тохируулахдаа ихэнх тохиолдолд администраторууд зөвхөн хоёр файлыг өөрчлөх хэрэгтэй: audit_control болон audit_user. Эхнийх нь системийн дагуух аудит өмчүүд болон бодлогуудыг хянадаг; хоёр дахь нь хэрэглэгчийн аудитийг нарийн тохируулахад ашиглагддаг.

### 18.4.2.1. audit_control файл

audit_control файл нь аудит дэд системийн хувьд анхдагч утгуудын тоог тодорхойлно. Энэ файлын дотор бид дараах зүйлсийг харна:

```
dir:/var/audit
flags:lo
minfree:20
naflags:lo
policy:cnt
filesz:0
```

`dir` тохиргоо нь аудит бүртгэлүүдийг хадгалах нэг болон хэд хэдэн санг заахад хэрэглэгдэнэ. Хэрэв нэгээс их сан байгаа бол бичигдсэн дарааллаараа ашиглагдана. Файлын систем дүүрсэн тохиолдолд аудит дэд систем болон бусад дэд системүүд бие биедээ нөлөөлж болзошгүй учир аудит бүртгэлүүдийг тусгайлан зориулсан файлын систем дээр хадгалахаар аудит системийг ихэвчлэн тохируулдаг.

`flags` талбар нь системийн дагуух шинж чанар бүхий үйл явцуудад зориулсан анхдагч урьдчилан сонголтын багийг тодорхойлдог. Дээрх жишээн дээр бүх хэрэглэгчийн хувьд амжилттай болон амжилтгүй нэвтрэлт болон гаралтын үйл явцууд аудит хийгдэж байна.

`minfree` тохиргоо нь аудит мөр хадгалагдах файлын системийн хувьд хамгийн бага чөлөөт зайны хувийг тодорхойлдог. Энэ тогтоосон хэмжээнээс илүү гарахад анхааруулга үүсгэгддэг. Дээрх жишээ хамгийн бага чөлөөт зайг 20 хувиар тогтоожээ.

`naflags` тохиргоо нь нэвтрэн оролтын процесс болон системийн дэмонууд зэрэг шинж чанаргүй үйл явцуудыг аудит хийх аудитийн ангиллуудыг тодорхойлдог.

`policy` тохиргоо нь таслалаар тусгаарлагдсан, аудитийн зан авирын төрөл бүрийн шинж чанарыг хянах бодлогын тугуудын жагсаалтыг тодорхойлдог. Анхдагч `cnt` туг нь аудит амжилттүй болсон ч гэсэн систем ажиллагаагаа үргэлжлүүлэхийг заадаг (энэ туг зайлшгүй шаардлагатай). Өөр нэг байнга ашиглагддаг туг бол `argv` бөгөөд энэ нь execve(2) системийн дуудлагад орж байгаа тушаалын мөрийн нэмэлт өгөгдлүүдийг тушаалын ажиллагааг аудит хийхийн хэсэг болох боломж олгодог.

`filesz` тохиргоо нь мөрийн файл автоматаар төгсөх болон эргэхээс өмнөх аудит мөрийн файлын хамгийн их хэмжээг байтаар тодорхойлдог. Анхдагч утга нь 0 байх ба автоматаар эргүүлэхийг хориглосон байна. Хэрэв хүссэн файлын хэмжээ тэгээс ялгаатай ба 512k -аас бага бол түүнийг орхиж бүртгэлийн мэдээлэл үүсгэнэ.

#### 18.4.2.2. audit_user файл

audit_user файл нь зарим нэг хэрэглэгчдэд зориулсан аудитийн шаардлагуудыг администраторууд тодорхойлохыг зөвшөөрдөг. Мөр болгон хэрэглэгчид зориулсан аудитийг хийхийг хоёр талбараар тохируулдаг: нэг дэх нь хэрэглэгчийн хувьд үргэлж аудит хийх шаардлагатай нэг хэсэг үйл явцуудыг тодорхойлдог `alwaysaudit` талбар ба хоёр дахь нь `neveraudit` талбар бөгөөд хэрэглэгчийн хувьд хэзээ ч аудит хийх шаардлагагүй нэг хэсэг үйл явцуудыг тодорхойлдог.

Дараах жишээн дээр audit_user файл нь нэвтрэлт/гаралтын үйл явцууд, `root` хэрэглэгчийн амжилттай тушаалын ажиллагаа, файл үүсгэлт ба `www` хэрэглэгчийн амжилттай тушаалын ажиллагааг аудит хийж байна. Хэрэв дээрх жишээ audit_control файлтай цуг ашиглагдвал `root`-ийн `lo` оруулга нь давхардах бөгөөд `www` хэрэглэгчийн нэвтрэлт/гаралтын үйл явцууд бас аудит хийгдэнэ.

```
root:lo,+ex:no
www:fc,+ex:no
```

# 18.5. Аудит дэд системийг удирдах нь

## 18.5.1. Аудит мөрүүдийг харах нь

Аудит мөрүүд нь BSM хоёртын хэлбэрээр хадгалагддаг бөгөөд өөрчлөх болон текст уруу хөрвүүлэхэд тусгай хэрэгслүүд ашиглах шаардлагатай. praudit(1) тушаал нь мөрийн файлуудыг хялбар текст хэлбэрт хөрвүүлдэг; auditreduce(1) тушаал нь аудит мөрийн файлыг шинжлэх, архивлах эсвэл хэвлэх зорилгоор багасгахад ашиглагддаг. `auditreduce` нь үйл явцын төрөл, үйл явцын ангилал, үйл явцын хэрэглэгч, огноо эсвэл цаг, файлын зам эсвэл обьектийн үйлдэл үзүүлсэн зэрэг төрөл бүрийн сонголтын параметрүүдийг дэмждэг.

Жишээ нь `praudit` хэрэгсэл нь заасан аудит бүртгэлийн бүх агуулгыг жирийн текстээр харуулна:

```
# praudit /var/audit/AUDITFILE
```

AUDITFILE нь харуулах аудит бүртгэл юм.

Аудит мөрүүд нь токенуудаас бүтэх аудит бичлэгүүдийн цувралаас тогтох бөгөөд `praudit` нь мөр болгонд нэгийг дараалуулан хэвлэнэ. Токен бүр аудит бичлэгийн толгойг агуулсан `header` эсвэл нэрийн хайлтаас гарсан файлын замыг агуулсан `path` зэрэг тусгай төрлийн байна. Дараах жишээ нь `execve` үйл явцыг харуулж байна:

```
header,133,10,execve(2),0,Mon Sep 25 15:58:03 2006, + 384 msec
exec arg,finger,doug
path,/usr/bin/finger
attribute,555,root,wheel,90,24918,104944
subject,robert,root,wheel,root,wheel,38439,38032,42086,128.232.9.100
return,success,0
trailer,133
```

Энэхүү аудит нь амжилттай `execve` дуудлагыг илэрхийлж байгаа бөгөөд түүнд `finger doug` тушаал ажиллажээ. Нэмэлт өгөгдлийн токен нь цөм уруу бүрхүүлээс гарч боловсруулагдсан тушаалын мөрийг агуулна. `path` буюу замын токен нь цөмийн хайсан ажиллах файлын замыг агуулна. `attribute` буюу шинж чанарын токен нь хоёртын файлыг тайлбарлах ба тухайлбал програм setuid эсэхийг тодорхойлоход ашиглагдах файлын горимыг агуулна. `subject` буюу субьект токен нь субьект процессийг тайлбарлах бөгөөд аудит хэрэглэгчийн ID, идэвхитэй хэрэглэгчийн ID ба бүлгийн ID, жинхэнэ хэрэглэгчийн ID ба бүлгийн ID, процессийн ID, сессийн ID, портын ID болон нэвтрэлтийн хаяг гэсэн дарааллаар хадгална. Аудит хэрэглэгчийн ID ба жинхэнэ хэрэглэгчийн ID нь ялгаатайг анхаарах хэрэгтэй: `robert` гэдэг хэрэглэгч энэ тушаалыг ажиллуулахаасаа өмнө `root` бүртгэл уруу шилжсэн бөгөөд энэ нь эхний шалгуулсан хэрэглэгчийг ашиглан аудит хийгдсэн байна. Төгсгөлд нь `return` буюу буцах токен нь амжилттай ажиллагааг харуулж `trailer` нь бичлэгийг төгсгөнө.

`praudit` нь бас XML гаралтын хэлбэрийг дэмждэг бөгөөд үүнийг `-x` нэмэлт өгөгдлийг ашиглан сонгож болдог.

### 18.5.2. Аудитийн мөрүүдийг багасгах нь

Аудит бүртгэлүүд нь маш их байж болно, администратор зарим хэрэглэгчтэй холбоотой бичлэгүүд зэрэг хэсэг бичлэгүүдийг ашиглахын тулд шилж сонгохыг магадгүй хүснэ:

```
# auditreduce -u trhodes /var/audit/AUDITFILE | praudit
```

Энэ нь AUDITFILE файлд хадгалагдсан `trhodes` хэрэглэгчийн бүх аудитийн бичлэгүүдийг сонгож байна.

### 18.5.3. Аудит хянах эрхүүдийг томилох нь

`audit` бүлгийн гишүүд /var/audit дахь аудит мөрүүдийг унших эрхтэй; анхандаа энэ бүлэг нь хоосон байх бөгөөд тэгэхээр зөвхөн `root` хэрэглэгч аудит мөрүүдийг уншиж чадна. Аудит хянах эрхүүдийг хэрэглэгчдэд томилохын тулд хэрэглэгчдийг `audit` бүлэгт нэмж болно. Аудитийн бүртгэлийн агуулгыг хянах чадвар нь хэрэглэгчид болон процессуудын үйл хөдлөлийн дотоод уруу нэлээн гүнзгий ханддаг учир аудит хянах эрхүүдийг томилохдоо болгоомжтой хийхийг зөвлөж байна.

### 18.5.4. Аудит хоолойнуудыг шууд монитор хийх нь

Аудит хоолойнууд нь төхөөрөмжийн файлын систем дахь клон хийгдсэн псевдо төхөөрөмжүүд бөгөөд програмыг шууд явж байгаа аудит бичлэгийн урсгалд холбох боломж олгоно. Энэ нь голчлон халдлага илрүүлэх болон систем монитор хийх програмуудын зохиогчдын сонирхлыг татдаг. Гэхдээ администраторуудын хувьд аудитийн хоолойны төхөөрөмж нь аудитийн мөрийн файлын эзэмшил эсвэл үйл явцын урсгалыг зогсоох, бүртгэл эргүүлэх зэрэг асуудлуудтай холбогдолгүйгээр шууд монитор хийх эвтэйхэн боломжийг бүрдүүлдэг. Шууд явж байгаа аудитийн үйл явцын урсгалыг хянахдаа дараах тушаалын мөрийг ашиглана:

```
# praudit /dev/auditpipe
```

Анхандаа аудитийн хоолойны төхөөрөмжийн цэгүүдэд зөвхөн `root` хэрэглэгч хандах эрхтэй байдаг. `audit` бүлгийн хэрэглэгчид хандах боломжтой болгохын тулд `devfs` дүрмийг devfs.rules -д нэмнэ:

```
add path 'auditpipe*' mode 0440 group audit
```

devfs файлын системийг тохируулах талаар дэлгэрэнгүй мэдээллийг devfs.rules(5) -ээс харна уу.

Аудитийн үйл явцын буцааж өгөх давталтуудыг үүсгэх нь хялбар бөгөөд аудит үйл явц бүрийн үзэлт нь олон аудитийн үйл явцуудыг үүсгэхэд хүргэнэ. Жишээ нь, хэрэв сүлжээний бүх I/O аудит хийгдсэн бөгөөд praudit(1) нь SSH сессээс ажилласан бол үйл явц бүр хэвлэгдэхэд өөр үйл явцыг бас үүсгэх учраас үргэлжилсэн аудитийн үйл явцууд их хэмжээгээр үүсэх болно. Энэ асуудлыг бий болгохгүйн тулд нарийн тохируулаагүй I/O аудит хийх сессээс `praudit`-ийг аудитийн хоолойны төхөөрөмж дээр ажиллуулахыг зөвлөж байна.

### 18.5.5. Аудит мөрийн файлуудыг эргүүлэх нь

Аудит мөрүүд нь зөвхөн цөмөөр бичигдэх бөгөөд auditd аудит дэмоноор удирдагддаг. Администраторууд аудит бүртгэлүүдийг шууд эргүүлэхдээ newsyslog.conf(5) эсвэл бусад хэрэгслүүдийг ашиглан хийх ёсгүй юм. Харин `audit` удирдах хэрэгслийг ашиглан аудитийг унтраах, аудит системийг дахин тохируулах, болон бүртгэлийг эргүүлэх үйлдлүүдийг хийх

боломжтой байдаг. Дараах тушаал аудит дэмонг шинэ аудит бүртгэл үүсгэж цөмийг шинэ бүртгэл уруу шилжихийг дохино. Хуучин бүртгэл нь төгсөж нэр нь өөрчлөгдөх бөгөөд дараагаар түүнтэй администратор ажиллах боломж бүрдэнэ.

```
# audit -n
```

Хэрэв auditd дэмон ажиллахгүй байгаа бол энэ тушаал нь амжилтгүй болох бөгөөд алдааны мэдээлэл үүсгэнэ.

Дараах мөрийг /etc/crontab -д нэмснээр cron(8) -оос арван хоёр цаг тутам эргүүлэх болно:

```
0     */12      *       *       *       root    /usr/sbin/audit -n
```

Шинэ /etc/crontab -ийг хадгалсны дараа өөрчлөлт үйлчилж эхлэх болно.

Файлын хэмжээн дээр тулгуурласан аудитийн мөрийн файлыг автоматаар эргүүлэх нь audit_control(5) дахь `filesz` тохиргоогоор хийгдэх боломжтой бөгөөд гарын авлагын энэ бүлгийн тохиргооны файлуудын хэсэгт тайлбарласан болно.

### 18.5.6. Аудит мөрүүдийг шахах нь

Аудит мөрийн файлууд асар их болох тусам мөрүүдийг аудит дэмоноор хаалгасны дараа шахах эсвэл архивлах нь зүйтэй юм. audit_warn скрипт нь аудитийн мөрүүдийг эргүүлэх үеийн цэвэр төгсгөл зэрэг төрөл бүрийн аудиттай холбоотой үйл явцуудад зориулан өөрчилсөн үйлдлүүдийг гүйцэтгэхэд ашиглагдана. Жишээ нь хаагдах үед аудит мөрүүдийг шахах дараах кодыг audit_warn скриптэд нэмж болно:

```
#
# Compress audit trail files on close.
#
if [ "$1" = closefile ]; then
        gzip -9 $2
fi
```

Бусад архивлах идэвхүүдэд мөрийн файлуудыг төв сервер уруу хуулах, хуучин мөрийн файлуудыг устгах эсвэл хэрэггүй бичлэгүүдийг хасч аудит мөрийг багасгах зэрэг орж болно. Аудит мөрийн файлууд цэвэрхэн дууссан тохиолдолд скрипт ажиллана, тэгэхээр буруу унтраасны дараа дуусаагүй мөрүүд дээр ажиллахгүй.

# Chapter 19. Хадгалалт

## 19.1. Ерөнхий агуулга

Энэ бүлэг нь FreeBSD дээр дискнүүдийг ашиглах талаар тайлбарлах болно. Эдгээрт санах ой дээр тулгуурласан дискнүүд, сүлжээнд залгагдсан дискнүүд, стандарт SCSI/IDE хадгалалтын төхөөрөмжүүд болон USB интерфэйс ашигладаг төхөөрөмжүүд багтах юм.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Физик диск (хуваалтууд болон зүсмэлүүд) дээрх өгөгдлийн зохион байгуулалтыг тайлбарладаг FreeBSD-ийн ашигладаг ухагдахуун.
- Өөрийн систем дээр нэмэлт хатуу дискнүүдийг хэрхэн нэмэх талаар.
- USB хадгалалтын төхөөрөмжүүдийг ашиглахын тулд FreeBSD-г хэрхэн тохируулах талаар.
- Санах ойн диск зэрэг виртуал файлын системүүдийг хэрхэн тохируулах талаар.
- Дискний зайн хэрэглээг хязгаарлахын тулд ноогдлыг хэрхэн ашиглах талаар.
- Дискийг халдагчдаас хамгаалж нууцлахын тулд хэрхэн шифрлэх талаар.
- FreeBSD дээр CD болон DVD-г хэрхэн үүсгэж шарах талаар.
- Нөөцлөлтөд зориулсан хадгалалтын төрөл бүрийн тохируулгууд.
- FreeBSD дээр байдаг нөөцлөлтийн програмуудыг хэрхэн ашиглах талаар.
- Уян диск уруу хэрхэн нөөцлөх талаар.
- Файлын системийн хормын хувилбар гэж юу болох түүнийг хэрхэн үр дүнтэйгээр ашиглах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- Шинэ FreeBSD цөмийг хэрхэн тохируулж суулгах талаар мэдэх (FreeBSD цөмийг тохируулах нь).

## 19.2. Төхөөрөмжийн нэрс

Дараах нь FreeBSD-д дэмжигдсэн физик хадгалалтын төхөөрөмжүүд болон тэдгээртэй холбоотой төхөөрөмжийн нэрсийн жагсаалт юм.

*Хүснэгт 7. Физик диск нэрлэх заншил*

| Хөтчийн төрөл | Хөтчийн төхөөрөмжийн нэр |
|---|---|
| IDE хатуу хөтчүүд | `ad` |
| IDE CDROM хөтчүүд | `acd` |
| SCSI хатуу хөтчүүд болон USB Mass хадгалалтын төхөөрөмжүүд | `da` |

| Хөтчийн төрөл | Хөтчийн төхөөрөмжийн нэр |
| --- | --- |
| SCSI CDROM хөтчүүд | `cd` |
| Төрөлжүүлсэн стандарт бус CDROM хөтчүүд | Mitsumi CD-ROM-ийн хувьд `mcd` ба Sony CD-ROM хөтчүүдийн хувьд `scd` |
| Уян хөтчүүд | `fd` |
| SCSI соронзон хальсны хөтчүүд | `sa` |
| IDE соронзон хальсны хөтчүүд | `ast` |
| Flash хөтчүүд | DiskOnChip® Flash хөтчийн хувьд `fla` |
| RAID хөтчүүд | Adaptec® AdvancedRAID-н хувьд `aacd`, Mylex®-ийн хувьд `mlxd` ба `mlyd`, AMI MegaRAID®-ийн хувьд `amrd`, Compaq Smart RAID-ийн хувьд `idad`, 3ware® RAID-ийн хувьд `twed`. |

# 19.3. Диск нэмэх

Дараах хэсэг зөвхөн нэг хөтөчтэй машинд шинэ SCSI диск хэрхэн нэмэх талаар тайлбарлах болно. Эхлээд компьютераа унтраагаад хөтчийг компьютер, хянагч болон хөтчийн үйлдвэрлэгчийн зааврууд дагуу суулгана. Үүнийг хийх маш олон төрлийн процедуруудаас болоод энэ тухай дэлгэрэнгүй мэдээлэл нь энэ баримтын хамрах хүрээнээс гадна юм.

`root` хэрэглэгчээр нэвтрэх хэрэгтэй. Та хөтчийг суулгасны дараа шинэ диск олдсон эсэхийг /var/run/dmesg.boot-с шалгаарай. Шинээр нэмсэн хөтөч нь da1 байх бөгөөд бид үүнийг /1 дээр холбохыг хүснэ (хэрэв та IDE хөтөч нэмж байгаа бол төхөөрөмжийн нэр нь ad1 болно).

FreeBSD нь IBM-PC-тэй нийцтэй компьютеруудтай ажилладаг учраас PC BIOS хуваалтуудыг бодолцох ёстой. Эдгээр нь уламжлалт BSD хуваалтуудаас өөр юм. PC диск нь дөрөв хүртэлх тооны BSD хуваалттай байдаг. Хэрэв диск нь жинхэнээрээ FreeBSD-д зориулагдах бол та *dedicated буюу зориулагдсан* горимыг ашиглаж болно. Үгүй бол FreeBSD нь PC BIOS хуваалтуудын аль нэгэн дээр байрлах болно. FreeBSD нь PC BIOS хуваалтуудыг уламжлалт BSD хуваалтуудтай эндүүрэхгүйн тулд *зүсмэлүүд* гэж нэрлэдэг. Та бас FreeBSD-д зориулагдсан боловч өөр үйлдлийн систем суулгагдсан компьютер дээр ашигласан диск дээрх зүсмэлүүдийг хэрэглэж болох юм. Энэ нь FreeBSD биш өөр үйлдлийн системийн `fdisk` хэрэгсэлтэй андуурахаас хамгаалах нэг сайн арга юм.

Зүсмэлийн хувьд бол хөтөч нь /dev/da1s1e гэж нэмэгдэх болно. Үүнийг SCSI диск, нэгжийн дугаар 1 (хоёр дахь SCSI диск), зүсмэл 1 (PC BIOS хуваалт 1) болон e BSD хуваалт гэж уншина. Зориулагдсан тохиолдолд хөтөч нь ердөө л /dev/da1e гэж нэмэгдэнэ.

Секторуудын тоог хадгалахын тулд 32 битийн бүхэл тоог ашигладгаас болоод bsdlabel(8) нь нэг дискний хувьд 2^32-1 сектор буюу ихэнх тохиолдолд 2TB болж хязгаарлагддаг. fdisk(8) хэлбэршүүлэлт нь 2^32-1-с ихгүй эхлэх сектор болон 2^32-1-с ихгүй уртыг зөвшөөрч хуваалтуудыг 2TB, дискнүүдийг ихэнх тохиолдолд 4TB болгож хязгаарладаг. sunlabel(8) хэлбэршүүлэлт нь нэг хуваалтын хувьд 2^32-1 сектороор, нийтдээ 16TB-ийн 8 хуваалтаар хязгаарлагддаг. Илүү том дискнүүдийн хувьд gpart(8) ашиглан GPT хуваалтуудыг үүсгэж

болно. GPT нь 4 зүсмэлээр хязгаарлагддагүйгээрээ ашигтай байдаг.

### 19.3.1. sysinstall(8)-г ашиглах нь

1. Sysinstall-г жолоодох нь

   Та `sysinstall`-н хялбар ашиглаж болох цэснүүдийн тусламжтайгаар шинэ дискийг хуваан хаяглаж болох юм. `root` хэрэгчээр нэвтрэх буюу эсвэл `su` тушаалыг ашиглаарай. `sysinstall`-г ажиллуулж `Configure` цэс уруу орно. `FreeBSD Configuration Menu` дотор доош шилжиж `Fdisk` тохируулгыг сонгоно.

2. fdisk хуваалт засварлагч

   fdisk-ийн дотор байхдаа `A`-г дарвал дискийг бүхэлд нь FreeBSD-д ашиглах болно. Асуух үед нь хэрэв та "ирээдүйд суулгаж болзошгүй үйлдлийн системүүдтэй хамтран ажиллахаар үлдэхийг хүсвэл" `YES` гэж хариулаарай. `W`-г ашиглан өөрчлөлтүүдийг диск уруу бичнэ. Одоо FDISK засварлагчаас `Q`-г дарж гараарай. Дараа нь танаас "Master Boot Record буюу Мастер Ачаалагч Бичлэгийн" талаар асуух болно. Та ажиллаж байгаа систем дээр диск нэмж байгаа болохоор `None`-г сонгох хэрэгтэй.

3. Дискний Шошго засварлагч

   Дараа нь sysinstall-с гарч дахин түүнийг эхлүүлэх хэрэгтэй. Дээрх зааврыг дагаарай, гэхдээ энэ удаад `Label` тохируулгыг сонгоорой. Энэ нь `Disk Label Editor буюу дискний шошго засварлагч` уруу орно. Энд та уламжлалт BSD хуваалтуудыг үүсгэдэг. Диск нь `a-h` гэж хаяглагдсан найм хүртэлх хуваалтуудтай байж болно. Хуваалтын шошгонуудын цөөн хэд нь тусгай хэрэглээтэй байдаг. `a` хуваалт нь root хуваалтанд (/) ашиглагддаг. Тиймээс зөвхөн таны системийн диск (өөрөөр хэлбэл таны ачаалалт хийсэн диск) `a` хуваалттай байх ёстой. `b` хуваалт нь swap хуваалтуудад хэрэглэгддэг бөгөөд та swap хуваалттай олон дисктэй байж болох юм. `c` хуваалт нь зориулагдсан горимд бүх дискийг, зүсмэлийн горимд бүхэл FreeBSD зүсмэлийг эсвэл заадаг. Бусад хуваалтууд нь ерөнхий хэрэглээнд зориулагдсан.

   sysinstall-ийн шошго засварлагч нь root биш, swap биш хуваалтуудад зориулж `e` хуваалтыг илүүтэй үздэг. Шошго засварлагч дотор байхдаа `C`-г даран ганц файлын систем үүсгэх хэрэгтэй. Асуух үед, хэрэв энэ нь FS (файлын систем) эсвэл swap байх юм бол `FS`-г сонгож холбох цэгийг (өөрөөр хэлбэл /mnt) бичээрэй. Хэрэв дискийг суулгацын дараах горимд нэмж байгаа бол sysinstall нь танд зориулж оруулгуудыг /etc/fstab файлд үүсгэхгүй, тиймээс таны зааж өгсөн холбох цэг нь чухал биш юм.

   Та одоо шинэ шошгыг диск уруу бичиж түүн дээр файлын систем үүсгэхэд бэлэн боллоо. Үүнийг `W`-г дарж хийнэ. sysinstall-ын шинэ хуваалтыг холбож чадахгүй байна гэсэн алдааг өнгөрүүлэх хэрэгтэй. Шошго засварлагч болон sysinstall-с бүр мөсөн гараарай.

4. Төгсгөл

   Хамгийн сүүлийн алхам нь /etc/fstab файлыг засварлаж өөрийн шинэ дискний оруулгыг нэмэх явдал юм.

## 19.3.2. Тушаалын мөрийн хэрэгслүүдийг ашиглах нь

### 19.3.2.1. Зүсмэлүүдийг ашиглах нь

Энэ тохиргоо нь таны дискийг өөрийн чинь компьютер дээр суулгагдсан байж болох бусад үйлдлийн системтэй зөв ажиллаж өөр бусад үйлдлийн системийн `fdisk` хэрэгслүүдтэй эндүүрэхгүй байх боломжийг бүрдүүлдэг. Шинэ дискийг суулгахад энэ аргыг ашиглахыг зөвлөдөг. Хэрэв танд үнэхээр тохирох шалтгаан байгаа тохиолдолд `зориулагдсан` горимыг ашиглаарай!

```
# dd if=/dev/zero of=/dev/da1 bs=1k count=1
# fdisk -BI da1 #Initialize your new disk
# bsdlabel -B -w da1s1 auto #Label it.
# bsdlabel -e da1s1 # Edit the bsdlabel just created and add any partitions.
# mkdir -p /1
# newfs /dev/da1s1e # Repeat this for every partition you created.
# mount /dev/da1s1e /1 # Mount the partition(s)
# vi /etc/fstab # Add the appropriate entry/entries to your /etc/fstab.
```

Хэрэв танд IDE диск байвал da-г ad гэж солиорой.

### 19.3.2.2. Зориулагдсан

Хэрэв та шинэ хөтчийг өөр үйлдлийн системтэй цуг хуваалцахгүй бол `зориулагдсан` горимыг ашиглаж болох юм. Энэ горим нь Microsoft үйлдлийн системийн толгойг эргүүлж болохыг санаарай; гэхдээ тэдгээр нь ямар ч эвдрэл гэмтэл үүсгэхгүй. IBM-ийн OS/2® нь харин олсон бүх ойлгохгүй байгаа ямар ч хуваалтыг хувьдаа "завших" болно.

```
# dd if=/dev/zero of=/dev/da1 bs=1k count=1
# bsdlabel -Bw da1 auto
# bsdlabel -e da1               # create the `e' partition
# newfs /dev/da1e
# mkdir -p /1
# vi /etc/fstab             # add an entry for /dev/da1e
# mount /1
```

Өөр нэг арга нь:

```
# dd if=/dev/zero of=/dev/da1 count=2
# bsdlabel /dev/da1 | bsdlabel -BR da1 /dev/stdin
# newfs /dev/da1e
# mkdir -p /1
# vi /etc/fstab                 # add an entry for /dev/da1e
# mount /1
```

# 19.4. RAID

## 19.4.1. Програм хангамжийн RAID

### 19.4.1.1. Нийлүүлэгдсэн Дискний Драйвер (Concatenated Disk Driver буюу CCD) тохиргоо

Бөөн хадгалалтын шийдлийг сонгохдоо бодолцох хамгийн чухал хүчин зүйлүүд нь хурд, найдвартай байдал болон өртөг юм. Энэ гурвыг гурвууланг нь тэнцүү байлгах нь ховор байдаг; ерөнхийдөө хурдан, найдвартай бөөн хадгалалтын төхөөрөмж нь үнэтэй бөгөөд үнийн хувьд хямдыг сонгох нь хурд эсвэл найдвартай байдлын аль нэгийг золиослох хэрэгтэй болдог.

Доор тайлбарласан системийг дизайн хийхдээ өртгийг хамгийн чухал хүчин зүйл гэж сонгож авсан бөгөөд үүний дараа хурд, хурдын дараа найдвартай байдлыг сонгосон. Энэ системийн өгөгдөл дамжуулах хурд нь эцсийн эцэст сүлжээгээр шахагддаг. Найдвартай байдал нь маш чухал боловч доор тайлбарласан CCD хөтөч нь CD-R-ууд дээр аль хэдийн бүтнээрээ нөөцлөгдсөн, амархнаар солигдож болох өгөгдөлд шууд үйлчилдэг.

Өөрийн шаардлагыг тодорхойлох нь бөөн хадгалалтын шийдлийг сонгох анхны алхам юм. Хэрэв таны шаардлага нь хурд эсвэл найдвартай байдлыг өртгөөс илүүтэй үзэж байгаа бол таны шийдэл энэ хэсэгт тайлбарласан шийдлээс өөр болох болно.

#### 19.4.1.1.1. Тоног төхөөрөмжийг суулгах нь

IDE системийн дискнээс гадна гурван Western Digital 30GB, 5400 RPM IDE диск нь доор тайлбарласан CCD дискний гол цөм болж нийтдээ ойролцоогоор 90GB шууд хадгалалт болно. Туйлын хүслээр бол IDE диск бүр өөрийн IDE хянагч болон кабельтай байна, гэхдээ өртгийг багасгахын тулд нэмэлт IDE хянагчууд ашиглагдаагүй болно. Харин дискнүүд нь jumper буюу холбогчоор тохируулагдсан бөгөөд ингэснээр IDE хянагч бүр нэг мастер болон нэг боолтой байна.

Дахин ачаалах үед системийн BIOS залгагдсан дискнүүдийг автоматаар олохоор тохируулагдсан байдаг. Илүү чухал зүйл нь FreeBSD тэдгээрийг дахин ачаалахад олсон явдал юм:

```
ad0: 19574MB <WDC WD205BA> [39770/16/63] at ata0-master UDMA33
ad1: 29333MB <WDC WD307AA> [59598/16/63] at ata0-slave UDMA33
ad2: 29333MB <WDC WD307AA> [59598/16/63] at ata1-master UDMA33
ad3: 29333MB <WDC WD307AA> [59598/16/63] at ata1-slave UDMA33
```

Хэрэв FreeBSD дискнүүдийг бүгдийг нь олохгүй байгаа бол та тэдгээрийг зөвөөр холбосон эсэхээ шалгаарай. IDE хөтчүүдийн ихэнх нь бас "Cable Select" холбогчтой байдаг. Энэ нь мастер/боол харилцаанд зориулагдсан холбогч *биш* юм. Хөтчийн баримтаас зөв холбогчийг таних талаар лавлаарай.

Дараа нь тэдгээрийг файлын системийн хэсэг болгон залгах талаар бодох хэрэгтэй. Та vinum(4) (Vinum Эзлэхүүн Менежер) болон ccd(4) хоёуланг судлах хэрэгтэй. Энэ тохиргооны

хувьд ccd(4)-г сонгосон.

#### 19.4.1.1.2. CCD-г тохируулах нь

ccd(4) хөтөч нь хэд хэдэн адил дискнүүдийг авч тэдгээрийг нэг логик файл систем болгон нийлүүлэх боломжийг олгодог. ccd(4)-г ашиглахын тулд танд ccd(4) дэмжлэг цуг бүтээгдсэн цөм хэрэгтэй. Энэ мөрийг цөмийн тохиргооны файлдаа нэмээд цөмөө дахин бүтээж суулгаарай:

```
device   ccd
```

ccd(4) дэмжлэг цөмийн дуудагдах модуль хэлбэрээр бас дуудагдаж болно.

ccd(4)-г тохируулахын тулд та эхлээд дискнүүдийг хаяглах bsdlabel(8)-г ашиглах ёстой:

```
bsdlabel -w ad1 auto
bsdlabel -w ad2 auto
bsdlabel -w ad3 auto
```

Энэ нь бүх дискний дагуух ad1c, ad2c болон ad3c-д зориулж bsdlabel үүсгэдэг.

Дараагийн алхам нь дискний шошгоны төрлийг өөрчлөх явдал юм. Та дискнүүдийг засварлахдаа bsdlabel(8)-г ашиглаж болно:

```
bsdlabel -e ad1
bsdlabel -e ad2
bsdlabel -e ad3
```

Энэ нь диск бүр дэх тухайн дискний шошгыг `EDITOR` орчны хувьсагчид заасан засварлагчаар, ихэнхдээ vi(1)-ээр онгойлгодог.

Өөрчлөлт хийгдээгүй дискний шошго иймэрхүү харагдах болно:

```
8 partitions:
#        size   offset    fstype   [fsize bsize bps/cpg]
  c: 60074784        0    unused        0     0     0   # (Cyl.    0 - 59597)
```

ccd(4)-д зориулж ашиглахаар шинэ `e` хуваалтыг нэмнэ. Үүнийг ихэвчлэн `c` хуваалтаас хуулж болох боловч `fstype` нь `4.2BSD` байх *ёстой*. Дискний шошго одоо иймэрхүү харагдах ёстой:

```
8 partitions:
#        size   offset    fstype   [fsize bsize bps/cpg]
  c: 60074784        0    unused        0     0     0   # (Cyl.    0 - 59597)
  e: 60074784        0    4.2BSD        0     0     0   # (Cyl.    0 - 59597)
```

#### 19.4.1.1.3. Файлын системийг бүтээх нь

Та бүх дискнүүдээ хаягласны дараа ccd(4)-г бүтээх ёстой. Үүнийг хийхийн тулд дараах тохируулгуудтай адилаар ccdconfig(8)-г ашиглана:

```
ccdconfig ccd0 32 0 /dev/ad1e /dev/ad2e /dev/ad3e
```

Тохируулга бүрийн хэрэглээ болон утгыг доор харуулав: * Эхний нэмэлт өгөгдөл нь тохируулах төхөөрөмж байх бөгөөд энэ тохиолдолд /dev/ccd0c байна. /dev/ хэсэг байхгүй ч байж болно. * Файлын системд зориулсан interleave. interleave нь дискний блокууд дээрх судлын хэмжээг тодорхойлдог бөгөөд нэг бүр нь ихэвчлэн 512 байт байдаг. Тэгэхээр 32 interleave нь 16,384 байт байна. * ccdconfig(8)-д зориулсан тугнууд. Хэрэв та хөтчийг толин тусгал үүсгэж идэвхжүүлэхийг хүсвэл тугийг энд зааж өгч болно. Энэ тохиргоо нь ccd(4)-н хувьд толин тусгал үүсгэлтийг хангадаггүй учир энэ нь 0 (тэг) гэж тохируулагдсан. * ccdconfig(8) уруу өгөгдөх сүүлийн нэмэлт өгөгдлүүд нь массивт оруулах төхөөрөмжүүд юм. Төхөөрөмж бүрийн хувьд бүрэн гүйцэд замын нэрийг ашиглах хэрэгтэй.

ccdconfig(8)-г ажиллуулсны дараа ccd(4) тохируулагдана. Файлын систем суулгагдаж болно. Тохируулгуудын талаар newfs(8)-с лавлана уу, эсвэл ердөө л ингэж ажиллуулна:

```
newfs /dev/ccd0c
```

#### 19.4.1.1.4. Бүгдийг автомат болгох нь

Ерөнхийдөө та ccd(4)-г дахин ачаалах бүртээ холбохыг хүснэ. Үүнийг хийхийн тулд та эхлээд тохируулах хэрэгтэй. Өөрийн одоогийн тохиргоогоо дараах тушаал ашиглаж /etc/ccd.conf уруу бичих хэрэгтэй:

```
ccdconfig -g > /etc/ccd.conf
```

Дахин ачаалах үед скрипт `/etc/rc` нь хэрэв /etc/ccd.conf байвал `ccdconfig -C` тушаалыг ажиллуулна. Энэ нь ccd(4)-г холбож болохоор болгож автоматаар тохируулна.

Хэрэв та ганц хэрэглэгчийн горим уруу ачаалж байгаа бол ccd(4)-г mount(8) хийхээсээ өмнө массивыг тохируулахын тулд дараах тушаалыг ажиллуулах шаардлагатай:

```
ccdconfig -C
```

ccd(4)-г автоматаар холбохын тулд ccd(4)-н оруулгыг /etc/fstab файлд байрлуулах хэрэгтэй. Ингэсэн тохиолдолд энэ нь ачаалах үед холбогдох болно:

```
/dev/ccd0c              /media      ufs     rw      2       2
```

#### 19.4.1.2. Vinum Эзлэхүүн Менежер

Vinum Эзлэхүүн Менежер нь виртуал диск хөтчийг хийдэг блок төхөөрөмжийн драйвер юм. Энэ нь дискний тоног төхөөрөмжийг блок төхөөрөмжийн интерфэйсээс тусгаарлаж уян хатан байдал, ажиллагаа болон найдвартай байдлыг дискний хадгалалтын уламжлалт зүсмэлийн харагдалтаас илүүтэйгээр хангах тийм аргаар өгөгдлийг дүрсэлдэг. vinum(4) нь RAID-0, RAID-1 болон RAID-5 загваруудыг тус бүрт нь болон холбоотой байдлаар нь шийддэг.

vinum(4)-ийн талаар дэлгэрэнгүй мэдээллийг Vinum Эзлэхүүн Менежер-с үзнэ үү.

### 19.4.2. Тоног төхөөрөмжийн RAID

FreeBSD нь бас төрөл бүрийн тоног төхөөрөмжийн RAID хянагчуудыг дэмждэг. Эдгээр төхөөрөмжүүд нь FreeBSD-д зориулсан тусгай програм хангамжаар массивыг удирдах шаардлагагүйгээр RAID дэд системийг хянадаг.

Карт дээрх BIOS-г ашиглан карт нь дискний үйлдлүүдийн ихэнхийг өөрөө хянадаг. Дараах нь Promise IDERAID хянагчийг ашиглах тохиргооны товч тайлбар юм. Энэ карт суулгагдаж систем эхлэх үед мэдээллийг хүсэх мөрийг харуулна. Картны тохиргооны дэлгэц уруу орохын тулд зааврыг дагана. Эндээс залгагдсан бүх хөтчүүдийг нэгтгэх боломж танд байх болно. Ингэж хийснийхээ дараа диск(нүүд) нь FreeBSD-д нэг хөтөч шиг харагдах болно. Бусад RAID түвшингүүдийг бас тохируулж болно.

### 19.4.3. ATA RAID1 массивуудыг дахин бүтээх нь

FreeBSD нь массив дахь гэмтсэн, ажиллагаагүй болсон дискийг шууд солих боломжийг олгодог. Энэ нь дахин ачаалахаасаа өмнө таныг ийм асуудлыг мэдэхийг шаарддаг.

Та магадгүй доор дурдсантай адилыг /var/log/messages эсвэл dmesg(8) гаралт дээр харж болох юм:

```
ad6 on monster1 suffered a hard error.
ad6: READ command timeout tag=0 serv=0 - resetting
ad6: trying fallback to PIO mode
ata3: resetting devices .. done
ad6: hard error reading fsbn 1116119 of 0-7 (ad6 bn 1116119; cn 1107 tn 4 sn 11)\\
status=59 error=40
ar0: WARNING - mirror lost
```

atacontrol(8) ашиглан дэлгэрэнгүй мэдээллийг шалгана:

```
# atacontrol list
ATA channel 0:
    Master:      no device present
    Slave:   acd0 <HL-DT-ST CD-ROM GCR-8520B/1.00> ATA/ATAPI rev 0

ATA channel 1:
    Master:      no device present
    Slave:       no device present
```

```
ATA channel 2:
    Master:  ad4 <MAXTOR 6L080J4/A93.0500> ATA/ATAPI rev 5
    Slave:       no device present

ATA channel 3:
    Master:  ad6 <MAXTOR 6L080J4/A93.0500> ATA/ATAPI rev 5
    Slave:       no device present

# atacontrol status ar0
ar0: ATA RAID1 subdisks: ad4 ad6 status: DEGRADED
```

1. Та дискийг аюулгүйгээр салган авахын тулд эхлээд ata сувгийг ажиллахгүй байгаа дисктэй цуг салгана:

   ```
   # atacontrol detach ata3
   ```

2. Дискийг сольно.
3. ata сувгийг дахин залгана:

   ```
   # atacontrol attach ata3
   Master:  ad6 <MAXTOR 6L080J4/A93.0500> ATA/ATAPI rev 5
   Slave:   no device present
   ```

4. Шинэ дискийг массивт нөөц маягаар нэмнэ:

   ```
   # atacontrol addspare ar0 ad6
   ```

5. Массивыг дахин бүтээнэ:

   ```
   # atacontrol rebuild ar0
   ```

6. Дараах тушаалыг ашиглаж үйл явцыг шалгаж болно:

   ```
   # dmesg | tail -10
   [output removed]
   ad6: removed from configuration
   ad6: deleted from ar0 disk1
   ad6: inserted into ar0 disk1 as spare

   # atacontrol status ar0
   ar0: ATA RAID1 subdisks: ad4 ad6 status: REBUILDING 0% completed
   ```

7. Энэ үйлдэл хийгдэж дуустал хүлээх хэрэгтэй.

# 19.5. USB хадгалалтын төхөөрөмжүүд

Одоо үед маш олон гадаад хадгалалтын шийдлүүд байгаа бөгөөд Universal Serial Bus (USB): хатуу хөтчүүд, USB хуруун хөтчүүд, CD-R шарагчид зэргийг ашигладаг. FreeBSD нь эдгээр төхөөрөмжүүдийн дэмжлэгийг хангадаг.

## 19.5.1. Тохиргоо

USB бөөн хадгалалтын төхөөрөмжүүдийн драйвер umass(4) нь USB хадгалалтын төхөөрөмжүүдийн дэмжлэгийг хангадаг. Хэрэв та GENERIC цөм ашиглавал өөрийн тохиргоондоо юу ч өөрчлөх шаардлагагүй. Хэрэв та өөрчлөн тохируулсан цөм ашиглах бол таны цөмийн тохиргооны файлд дараах мөрүүд байгаа эсэхийг шалгаарай:

```
device scbus
device da
device pass
device uhci
device ohci
device ehci
device usb
device umass
```

umass(4) драйвер нь USB хадгалалтын төхөөрөмжүүдэд хандахын тулд SCSI дэд системийг хэрэглэдэг бөгөөд таны USB төхөөрөмж системд SCSI төхөөрөмж маягаар харагдах болно. Таны эх хавтан дээрх USB бичил схемээс хамаарч USB 1.X-ийн дэмжлэгийн хувьд танд зөвхөн `device uhci` эсвэл `device ohci` хоёрын аль нэг хэрэгтэй болно, гэхдээ хоёуланг нь цөмийн тохиргоондоо байлгах нь гэмгүй юм. USB 2.0 хянагчуудад зориулсан дэмжлэгийг ehci(4) драйвер (`device ehci` мөр) хангадаг. Хэрэв та ямар нэг мөр нэмсэн бол шинэ цөмөө эмхэтгэж суулгахаа битгий мартаарай.

Хэрэв таны USB төхөөрөмж чинь CD-R эсвэл DVD шарагч бол SCSI CD-ROM драйвер cd(4)-г цөмд дараах мөрийн тусламжтай нэмэх ёстой:

```
device cd
```

Шарагч нь SCSI хөтөч гэж харагддаг учир atapicam(4) драйверийг цөмийн тохиргоонд ашиглах ёсгүй.

## 19.5.2. Тохиргоог тест хийх нь

Тохиргоог тест хийхэд бэлэн боллоо: өөрийн USB төхөөрөмжийг залгахад системийн мэдэгдлийн буферт (dmesg(8)) хөтөч нь иймэрхүү харагдах ёстой:

```
umass0: USB Solid state disk, rev 1.10/1.00, addr 2
GEOM: create disk da0 dp=0xc2d74850
da0 at umass-sim0 bus 0 target 0 lun 0
da0: <Generic Traveling Disk 1.11> Removable Direct Access SCSI-2 device
da0: 1.000MB/s transfers
da0: 126MB (258048 512 byte sectors: 64H 32S/T 126C)
```

Мэдээж хэрэг үйлдвэрлэгч, төхөөрөмжийн цэг (da0) болон бусад зүйлс таны тохиргооноос хамаараад өөр байж болно.

USB төхөөрөмж нь SCSI төхөөрөмж гэж харагддаг болохоор `camcontrol` тушаалыг ашиглаж системд холбогдсон USB хадгалалтын төхөөрөмжүүдийн жагсаалтыг харуулж болно:

```
# camcontrol devlist
<Generic Traveling Disk 1.11>      at scbus0 target 0 lun 0 (da0,pass0)
```

Хэрэв хөтөч нь файлын системтэй ирвэл та түүнийг холбож чадна. Диск нэмэх хэсэг нь хэрэв шаардлагатай бол USB хөтчийг хэлбэршүүлж түүн дээр хуваалт үүсгэхэд танд туслах болно.

Дурын төхөөрөмжийг итгэлгүй хэрэглэгчдэд холбох боломжийг олгох нь өөрөөр хэлбэл `vfs.usermount`-ийг доор тайлбарласнаар идэвхжүүлэх нь аюулгүй байдлын үүднээс авч үзвэл аюултай юм. FreeBSD дээрх ихэнх файлын систем нь хортой төхөөрөмжийн эсрэг хамгаалалттүй байдаг.

Энэ төхөөрөмжийг энгийн хэрэглэгч холбож чаддагаар болгохын тулд тодорхой алхмуудыг хийх хэрэгтэй. Эхлээд USB хадгалалтын төхөөрөмж холбогдсон үед үүссэн төхөөрөмжүүдэд хэрэглэгч хандаж болохоор байх хэрэгтэй. Үүний шийдэл нь эдгээр төхөөрөмжүүдийн бүх хэрэглэгчдийг `operator` бүлгийн гишүүн болгох явдал юм. Үүнийг pw(8)-ээр хийнэ. Хоёрдугаарт төхөөрөмжүүд нь үүсэх үед `operator` бүлэг тэдгээрийг уншиж бичиж чадаж байх ёстой. Тохирох мөрүүдийг /etc/devfs.rules файлд нэмснээр үүнийг хийж болно:

```
[localrules=5]
add path 'da*' mode 0660 group operator
```

Хэрэв системд SCSI дискнүүд байгаа бол үүнийг арай өөрөөр хийх ёстой. Өөрөөр хэлбэл хэрэв систем нь аль хэдийн da0-ээс da2 хүртэлх холбогдсон дискнүүдийг агуулж байвал хоёр дахь мөрийг дараах маягаар солих хэрэгтэй:

```
add path 'da[3-9]*' mode 0660 group operator
```

Энэ нь байгаа дискнүүдийг `operator` бүлэгт хамааруулахгүй болгоно.

Та бас өөрийн devfs.rules(5) дүрмийн олонлогийг /etc/rc.conf файлд идэвхжүүлэх хэрэгтэй:

```
devfs_system_ruleset="localrules"
```

Дараа нь цөм нь ердийн хэрэглэгчдэд файлын системийг холбох боломжтойгоор тохируулагдах ёстой. Хамгийн хялбар арга бол /etc/sysctl.conf-д мөр нэмэх явдал юм:

```
vfs.usermount=1
```

Дараагийн дахин ачаалалтын дараа энэ нь идэвхжихийг санаарай. Өөрөөр энэ хувьсагчийг тохируулахын тулд sysctl(8)-г ашиглаж болох юм.

Төгсгөлийн алхам нь файлын систем холбогдох санг үүсгэх явдал юм. Энэ санг файлын системийг холбох хэрэглэгч эзэмшсэн байх хэрэгтэй. Үүнийг хийх нэг арга нь `root`-ийн хувьд тэр хэрэглэгчийн эзэмшсэн дэд санг /mnt/username (*username*-г тухайн хэрэглэгчийнхээ нэвтрэх нэрээр болон *usergroup*-г хэрэглэгчийнхээ үндсэн бүлгийн нэрээр солиорой) гэж үүсгэх явдал юм:

```
# mkdir /mnt/username
# chown username:usergroup /mnt/username
```

USB хуруун хөтөч залгагдаж /dev/da0s1 төхөөрөмж гарч иржээ гэж бодъё. Эдгээр төхөөрөмжүүд нь ихэвчлэн FAT файлын системээр хэлбэршүүлэгдсэн ирдэг бөгөөд эдгээрийг иймэрхүү маягаар холбож болно:

```
% mount -t msdosfs -m=644 -M=755 /dev/da0s1 /mnt/username
```

Хэрэв та төхөөрөмжийг залгаснаа салгавал (дискийг урьдаар салгах ёстой) та системийн мэдэгдлийн буфераас доор дурдсантай төстэй мэдэгдлийг харах ёстой:

```
umass0: at uhub0 port 1 (addr 2) disconnected
(da0:umass-sim0:0:0:0): lost device
(da0:umass-sim0:0:0:0): removing device entry
GEOM: destroy disk da0 dp=0xc2d74850
umass0: detached
```

### 19.5.3. Нэмэлт унших материалууд

Диск нэмэх болон Файлын системүүдийг холбох болон салгах хэсгүүдээс гадна төрөл бүрийн гарын авлагын хуудаснуудыг унших нь хэрэгтэй байж болох юм: FreeBSD 8.X-ийн хувьд umass(4), camcontrol(8), болон usbconfig(8) эсвэл FreeBSD-ийн өмнөх хувилбаруудын хувьд usbdevs(8) байна.

# 19.6. Оптик зөөвөрлөгчийг (CD-үүд) үүсгэж ашиглах нь

## 19.6.1. Танилцуулга

CD-үүд нь тэдгээрийг ердийн дискнүүдээс ялгах хэд хэдэн боломжуудтай байдаг. Эхлээд хэрэглэгч CD дээр бичих боломжгүй байсан. Тэдгээр нь замуудын хооронд толгойг шилжүүлэхдээ сааталгүйгээр үргэлжлэн уншдагаар хийгджээ. Тэр үед байсан адил хэмжээтэй зөөвөрлөгчийг зөөхөөс тэдгээрийг систем хооронд зөөх нь хамаагүй хялбар байдаг.

CD-үүд нь замтай байдаг боловч энэ нь дискний физик хэсэг биш харин үргэлжлэн уншигдах өгөгдлийн хэсгийг хэлдэг. FreeBSD дээр CD үүсгэхдээ CD дээр замууд үүсгэх өгөгдлийн файлуудыг бэлдэж дараа нь замуудыг CD уруу бичнэ.

ISO 9660 файлын систем нь эдгээр ялгаануудтай ажиллахаар хийгдсэн. Энэ нь тэр үед нийтлэг байсан файлын системийн хязгаарууд харамсалтай нь кодчилдог. Азаар энэ нь зөв бичигдсэн CD-үүдэд тэдгээр хязгаарууд давж гарахыг зөвшөөрөх өргөтгөлөөр хангадаг бөгөөд тэдгээр өргөтгөлүүдийг дэмждэггүй системүүдтэй ажиллаж чадсан хэвээр байдаг.

sysutils/cdrtools портод ISO 9660 файлын системийг агуулах өгөгдлийн файлыг үүсгэдэг програм mkisofs(8) байдаг. Энэ нь төрөл бүрийн өргөтгөлүүдийг дэмждэг тохируулгуудтай бөгөөд доор тайлбарлагдсан болно.

CD шарахдаа ямар хэрэгслийг ашиглах нь таны CD шарагч ATAPI юу аль эсвэл өөр үү гэдгээс шалтгаална. ATAPI CD шарагчид нь үндсэн системийн `burncd` програмыг ашигладаг. SCSI болон USB CD шарагчид нь sysutils/cdrtools портын `cdrecord`-г ашиглах ёстой. Мөн ATAPI тоног төхөөрөмж дээр SCSI хөтчүүдийн хувьд ATAPI/CAM модул ашиглан `cdrecord`-г хэрэглэх боломжтой байдаг.

Хэрэв та график хэрэглэгчийн интерфэйстэй CD шарагч програм хангамжийг хүсэж байгаа бол X-CD-Roast эсвэл K3b-г үзээрэй. Эдгээр хэрэгслүүд нь багц хэлбэрээр эсвэл sysutils/xcdroast болон sysutils/k3b портуудад байдаг. X-CD-Roast болон K3b нь ATAPI тоног төхөөрөмж дээр ATAPI/CAM модулийг шаарддаг.

## 19.6.2. mkisofs

sysutils/cdrtools портын хэсэг mkisofs(8) програм нь UNIX®-ийн файлын системийн нэрийн талбар дахь сангийн модны дүрс болох ISO 9660 файлын системийг үүсгэдэг. Хамгийн хялбар хэрэглээ нь:

```
# mkisofs -o imagefile.iso /path/to/tree
```

Энэ тушаал нь */path/to/tree* дахь модны хуулбар ISO 9660 файлын системийг агуулах *imagefile.iso* файлыг үүсгэх болно. Энэ процессод файлын нэрсийг ISO 9660 файлын системийн стандартын хязгаарлалтуудад багтах нэрсэд тааруулах бөгөөд ISO файлын системүүдэд байдаггүй нэрс бүхий файлуудыг оруулахгүй байх болно.

Тэдгээр хязгаарлалтуудыг давж гарах хэд хэдэн тохируулгууд байдаг. Ялангуяа `-R` тохируулга UNIX® системүүдэд нийтлэг байдаг Rock Ridge өргөтгөлүүдийг идэвхжүүлдэг, `-J` нь Microsoft системүүдэд хэрэглэгддэг Joilet өргөтгөлүүдийг идэвхжүүлдэг бөгөөд `-hfs` нь Mac OS®-д хэрэглэгддэг HFS файлын системүүдийг үүсгэхэд ашиглагддаг.

Зөвхөн FreeBSD системүүдэд ашиглагдах CD-үүдийн хувьд `-U` тохируулга бүх файлын нэрийн хязгаарлалтуудыг хаахад ашиглагдаж болно. `-R` тохируулгатай хэрэглэгдэх үед энэ нь таны эхэлсэн FreeBSD-ийн модтой ижил файлын системийн дүрсийг үүсгэдэг, гэхдээ энэ нь ISO 9660 стандартыг хэд хэдэн замаар зөрчиж болох юм.

Ердийн хэрэглээний сүүлийн тохируулга нь `-b` юм. Энэ нь ачаалагдах "El Torito" CD-г үүсгэхэд хэрэглэгдэх ачаалагдах дүрсний байрлалыг заахад ашиглагддаг. Энэ тохируулга нь CD уруу бичигдэх модны дээд хэсгийн ачаалагдах дүрс хүрэх замыг заах нэмэлт өгөгдлийг авдаг. Анхдагчаар mkisofs(8) нь "floppy disk emulation буюу уян дискний эмуляц" гэж нэрлэгддэг горимд ISO дүрсийг үүсгэдэг бөгөөд ачаалагдах дүрсийг яг 1200, 1440, эсвэл 2880 KB хэмжээтэй байна гэж тооцдог. FreeBSD түгээлтийн дискнүүдэд хэрэглэгддэг ачаалагч дуудагч зэрэг зарим ачаалагч дуудагчид нь эмуляц горимыг ашигладаггүй; энэ тохиолдолд `-no-emul-boot` тохируулгыг ашиглах шаардлагатай. Тэгэхээр хэрэв /tmp/myboot нь ачаалагдах FreeBSD системийг /tmp/myboot/boot/cdboot дэх ачаалагдах дүрстэй цуг агуулж байвал та ISO 9660 файлын системийн дүрсийг /tmp/bootable.iso-д иймэрхүү маягаар үүсгэж болох юм:

```
# mkisofs -R -no-emul-boot -b boot/cdboot -o /tmp/bootable.iso /tmp/myboot
```

Үүнийг хийснийхээ дараа хэрэв та цөмдөө md-г тохируулсан бол файлын системийг ингэж холбож болно:

```
# mdconfig -a -t vnode -f /tmp/bootable.iso -u 0
# mount -t cd9660 /dev/md0 /mnt
```

Энэ үед та /mnt болон /tmp/myboot нь ижил болохыг шалгаж болно.

mkisofs(8)-ийн ажиллагааг нарийн тааруулахын тулд та түүний бусад олон тохируулгуудыг ашиглаж болно. Ялангуяа ISO 9660-ийн байрлал болон Joilet ба HFS дискнүүдийн үүсгэлтэд өөрчлөлтүүд хийж болно. Дэлгэрэнгүйг mkisofs(8)-ийн гарын авлагын хуудаснаас үзнэ үү.

### 19.6.3. burncd

Хэрэв танд ATAPI CD шарагч байгаа бол та ISO дүрсийг CD уруу шарахдаа `burncd` тушаалыг ашиглаж болно. `burncd` нь үндсэн системийн хэсэг бөгөөд /usr/sbin/burncd гэж суулгагдсан байдаг. Энэ нь цөөн тохируулгуудтай болохоор хэрэглэхэд их хялбар байдаг:

```
# burncd -f cddevice data imagefile.iso fixate
```

Дээрх тушаал нь *imagefile.iso*-н хуулбарыг *cddevice* уруу шарах болно. Анхдагч төхөөрөмж нь /dev/acd0 юм. Бичих хурд, шарсны дараа CD-г гаргах болон аудио өгөгдөл бичихийг заах тохируулгуудын талаар burncd(8)-с үзнэ үү.

### 19.6.4. cdrecord

Хэрэв танд ATAPI CD шарагч байхгүй бол та өөрийн CD-үүдийг шарахын тулд cdrecord-г ашиглах шаардлагатай. cdrecord нь үндсэн системд байдаггүй; та үүнийг sysutils/cdrtools дахь портоос эсвэл тохирох багцаас суулгах ёстой. Үндсэн системд хийгдсэн өөрчлөлт нь энэ програмын хоёртын хувилбарыг ажиллахгүй болгож, магадгүй "асуудалд (coaster)" хүргэж болох юм. Тийм болохоор та өөрийн системээ шинэчлэхдээ портоо бас шинэчлэх эсвэл хэрэв та -STABLE салбарыг дагаж байгаа бол портыг шинэ хувилбар гарахад нь шинэчлэх хэрэгтэй.

cdrecord нь олон тохируулгатай байдаг боловч үндсэн хэрэглээ нь burncd-с бүр илүү хялбар байдаг. ISO 9660 дүрсийг шарахдаа:

```
# cdrecord dev=device imagefile.iso
```

cdrecord-г хэрэглэхэд гардаг нэг заль нь dev тохируулгыг олох явдал юм. Зөв тохиргоог олохын тулд cdrecord-ийн -scanbus тугийг хэрэглэх хэрэгтэй бөгөөд энэ нь иймэрхүү үр дүнд хүргэж болох юм:

```
# cdrecord -scanbus
Cdrecord-Clone 2.01 (i386-unknown-freebsd7.0) Copyright (C) 1995-2004 Jörg Schilling
Using libscg version 'schily-0.1'
scsibus0:
        0,0,0     0) 'SEAGATE ' 'ST39236LW       ' '0004' Disk
        0,1,0     1) 'SEAGATE ' 'ST39173W        ' '5958' Disk
        0,2,0     2) *
        0,3,0     3) 'iomega  ' 'jaz 1GB         ' 'J.86' Removable Disk
        0,4,0     4) 'NEC     ' 'CD-ROM DRIVE:466' '1.26' Removable CD-ROM
        0,5,0     5) *
        0,6,0     6) *
        0,7,0     7) *
scsibus1:
        1,0,0   100) *
        1,1,0   101) *
        1,2,0   102) *
        1,3,0   103) *
        1,4,0   104) *
        1,5,0   105) 'YAMAHA  ' 'CRW4260        ' '1.0q' Removable CD-ROM
        1,6,0   106) 'ARTEC   ' 'AM12S          ' '1.06' Scanner
        1,7,0   107) *
```

Энэ нь жагсаалтан дахь төхөөрөмжүүдийн хувьд тохирох dev утгыг жагсаадаг. Өөрийн CD шарагчийг олохын тулд dev тохируулгын утгад гурван дугаарыг таслалаар тусгаарлан хэрэглэнэ. Энэ тохиолдолд CRW төхөөрөмж нь 1,5,0, байх бөгөөд тохирох ролт нь dev=1,5,0 болно. Энэ утгыг заах амархан аргууд байдаг; дэлгэрэнгүйг cdrecord(1)-с үзнэ үү. Мөн тэндээс аудио замуудыг бичих, хурдыг хянах болон бусад зүйлүүдийн тухай мэдээллийг үзэж болно.

### 19.6.5. Аудио CD-үүдийг хувилах

Та аудио өгөгдлийг CD-ээс файлуудын цуваа болгон задалж дараа нь эдгээр файлуудыг хоосон CD дээр бичин аудио CD-г хувилж болно. Энэ процесс нь ATAPI болон SCSI хөтчүүдийн хувьд нэлээн өөр байдаг.

**Procedure: SCSI хөтчүүд**

1. Аудиог `cdda2wav` ашиглан задлана.

```
% cdda2wav -vall -D2,0 -B -Owav
```

2. `cdrecord` ашиглан .wav файлуудыг бичнэ.

```
% cdrecord -v dev=2,0 -dao -useinfo  *.wav
```

   `cdrecord` хэсэгт тайлбарласны дагуу *2,0* гэж зөв заагдсан эсэхийг шалгаарай.

**Procedure: ATAPI хөтчүүд**

ATAPI/CAM модулийн тусламжтай `cdda2wav` тушаал ATAPI хөтчүүд дээр ашиглагдаж болно. Энэ хэрэгсэл нь доор санал болгож байгаа аргыг бодвол ихэнх хэрэглэгчдийн хувьд ихэвчлэн илүүтэй сонголт байдаг (доргио засварлалт, төгсгөгчийн асуудал гэх мэт).

1. ATAPI CD драйвер нь зам бүрийг /dev/acddtnn маягаар болгодог бөгөөд *d* нь хөтчийн дугаар ба *nn* нь шаардлагатай бол урдаа 0 тавьж хоёр оронтой тоогоор бичигдсэн замын дугаар юм. Тэгэхээр эхний диск дээрх эхний зам нь /dev/acd0t01, хоёр дахь нь /dev/acd0t02, гурав дахь нь /dev/acd0t03 гэх мэтчилэн байна.

   Тохирох файлууд /dev санд байгаа эсэхийг шалгаарай. Хэрэв оруулгууд байхгүй байгаа бол зөөвөрлөгчийг дахин үзэхээр системийг хүчлэх хэрэгтэй:

```
# dd if=/dev/acd0 of=/dev/null count=1
```

2. Зам бүрийг dd(1) ашиглан задална. Файлуудыг задлахдаа та тусгай блокийн хэмжээг бас ашиглах ёстой.

```
# dd if=/dev/acd0t01 of=track1.cdr bs=2352
# dd if=/dev/acd0t02 of=track2.cdr bs=2352
...
```

3. Задалсан файлуудаа диск уруу `burncd` ашиглан шарна. Та эдгээрийг аудио файл гэж

зааж өгөх хэрэгтэй бөгөөд burncd нь дуусахдаа дискийг бэхжүүлэх ёстой.

```
# burncd -f /dev/acd0 audio track1.cdr track2.cdr ... fixate
```

### 19.6.6. Өгөгдлийн CD-үүдийг хувилах

Та өгөгдлийн CD-г mkisofs(8)-р үүсгэсэн дүрс файлтай ажиллагааны хувьд адилхан дүрс файл уруу хуулж болох бөгөөд та үүнийг ямар ч өгөгдлийн CD хувилахад ашиглаж болно. Энд өгөгдсөн жишээ нь таны CDROM төхөөрөмжийг acd0 гэж үзэх болно. Өөрийн зөв CDROM төхөөрөмжөөр солиорой.

```
# dd if=/dev/acd0 of=file.iso bs=2048
```

Одоо та нэгэнт дүрстэй болсон болохоор үүнийг CD уруу дээр тайлбарласны дагуу шарж болно.

### 19.6.7. Өгөгдлийн CD-үүдийг ашиглах

Одоо та стандарт өгөгдлийн CDROM үүсгэсэн болохоор түүнийг холбож түүн дээрх өгөгдлийг уншихыг хүсэх байх. Анхдагчаар mount(8) нь файлын системийг ufs төрлийнх гэж үздэг. Хэрэв та доорх шиг оролдвол:

```
# mount /dev/cd0 /mnt
```

Incorrect super block гэж гомдоллохыг та харах бөгөөд холболт хийгдэхгүй байх болно. CDROM нь UFS файлын систем биш, тэгэхээр ингэж холбохыг оролдох нь амжилтгүй болох болно. Та mount(8)-д файлын системийн төрөл нь ISO9660 гэж зааж өгөхөд л бүгд ажиллах болно. Та -t cd9660 тохируулгыг mount(8)-д өгч үүнийг хийнэ. Жишээ нь хэрэв та CDROM төхөөрөмж /dev/cd0-г /mnt-д холбохыг хүсвэл дараах тушаалыг ажиллуулах болно:

```
# mount -t cd9660 /dev/cd0 /mnt
```

Таны төхөөрөмжийн нэр (энэ жишээн дээр /dev/cd0) таны CDROM ямар интерфэйс ашиглаж байгаагаас хамааран өөр байж болох юм. Мөн -t cd9660 тохируулга нь ердөө л mount_cd9660(8)-г ажиллуулдаг. Дээрх жишээг ингэж богиносгож болно:

```
# mount_cd9660 /dev/cd0 /mnt
```

Та ерөнхийдөө энэ аргаар ямар ч үйлдвэрлэгчийн өгөгдлийн CDROM-уудыг ашиглаж болно. Гэхдээ зарим нэг ISO 9660 өргөтгөлүүдтэй дискнүүд хачин ажиллаж болох юм. Жишээ нь Joilet дискнүүд нь бүх файлын нэрсийг хоёр байт Юникод тэмдэгтээр хадгалдаг. FreeBSD цөм нь Юникодоор ярьдаггүй, гэхдээ FreeBSD-ийн CD9660 драйвер Юникод тэмдэгтүүдийг шууд хувиргаж чаддаг. Хэрэв зарим нэг Англи бус тэмдэгтүүд асуултын тэмдэг хэлбэрээр

харагдвал та ашиглаж байгаа локал тэмдэгтийн олонлогоо `-C` тохируулгаар зааж өгөх хэрэгтэй. Дэлгэрэнгүй мэдээллийг mount_cd9660(8) гарын авлагын хуудаснаас лавлана уу.

Энэ тэмдэгтийн хувиргалтыг `-C` тохируулгын тусламжтай хийхийн тулд цөм cd9660_iconv.ko модулийг дуудсан байхыг шаардах болно. Энэ мөрийг loader.conf файлд нэмж үүнийг:

```
cd9660_iconv_load="YES"
```

гэж хийн машиныг дахин ачаалах буюу эсвэл модулийг kldload(8)-н тусламжтай дуудан хийж болох юм.

Хааяа таныг CDROM-г холбохыг оролдох үед `Device not configured` гэсэн алдаа гарч болох юм. Энэ нь ихэнхдээ CDROM хөтөч нь төхөөрөмжид диск байхгүй эсвэл хөтөч нь шугаманд (bus) харагдахгүй байна гэж үзэж байна гэсэн үг юм. CDROM хөтөч нь хоёр секундын дотор үүнийг мэддэг болохоор тэвчээртэй байгаарай.

Шугамын дахин тогтоолтод хариу өгөх хангалттай хугацаа байхгүйн улмаас заримдаа SCSI CDROM-ийг олохгүй байж болох юм. Хэрэв та SCSI CDROM-той бол та дараах тохируулгыг цөмийн тохиргоондоо нэмж өөрийн цөмийг дахин бүтээнэ үү.

```
options SCSI_DELAY=15000
```

Энэ нь таны SCSI шугамд ачаалах үедээ 15 секунд түр саатахыг хэлж өгөх бөгөөд ингэснээр шугамын дахин тогтоолтод таны CDROM хөтчөөр хариу өгүүлэхийн тулд бүх байж болох боломжийг түүнд өгч байна гэсэн үг юм.

### 19.6.8. Түүхий өгөгдлийн CD-үүдийг шарах

Та ISO 9660 файлын системийг үүсгэлгүйгээр файлыг CD уруу шууд шарахаар сонгож болно. Зарим хүмүүс үүнийг нөөцлөх зорилгоор хийдэг. Энэ нь стандарт CD-г шарахаас илүү хурдан ажилладаг:

```
# burncd -f /dev/acd1 -s 12 data archive.tar.gz fixate
```

Тийм CD-д шарагдсан өгөгдлийг авахын тулд та түүхий төхөөрөмжийн цэгээс өгөгдлийг унших ёстой:

```
# tar xzvf /dev/acd1
```

Та энэ дискийг ердийн CDROM-ийг холбодог шиг холбож чадахгүй. Ийм CDROM нь FreeBSD-ээс өөр ямар ч үйлдлийн систем дээр уншигдахгүй. Хэрэв та CD-гээ холбохыг эсвэл өөр үйлдлийн системтэй өгөгдлөө хуваалцах хүсэлтэй байгаа бол дээр тайлбарласны дагуу mkisofs(8)-г ашиглах ёстой.

### 19.6.9. ATAPI/CAM драйверийг ашиглах

Энэ драйвер нь ATAPI төхөөрөмжүүдэд (CD-ROM, CD-RW, DVD хөтчүүд гэх мэт...) SCSI дэд системээр хандах боломжийг олгох бөгөөд ингэснээр sysutils/cdrdao эсвэл cdrecord(1) зэрэг програмуудыг ашиглах боломжийг олгодог.

Энэ драйверыг ашиглахын тулд та дараах мөрийг /boot/loader.conf файл уруу нэмэх хэрэгтэй болно:

```
atapicam_load="YES"
```

тэгээд өөрийн машинаа дахин ачаална.

Хэрэв та өөрийн цөмдөө atapicam(4) дэмжлэгийг статикаар эмхэтгэхийг хүсвэл энэ мөрийг өөрийн цөмийн тохиргооны файлдаа нэмэх хэрэгтэй болно:

```
device atapicam
```

Мөн та өөрийн цөмийн тохиргооны файлдаа дараах мөрүүдийг бас нэмэх хэрэгтэй болно:

```
device ata
device scbus
device cd
device pass
```

Эдгээр нь аль хэдийн байж байх ёстой. Дараа нь дахин бүтээгээд өөрийн цөмийг суулгаж машинаа дахин ачаалах хэрэгтэй.

Ачаалах процессийн үед таны шарагч иймэрхүү маягаар гарч ирэх ёстой:

```
acd0: CD-RW <MATSHITA CD-RW/DVD-ROM UJDA740> at ata1-master PIO4
cd0 at ata1 bus 0 target 0 lun 0
cd0: <MATSHITA CDRW/DVD UJDA740 1.00> Removable CD-ROM SCSI-0 device
cd0: 16.000MB/s transfers
cd0: Attempt to query device size failed: NOT READY, Medium not present - tray closed
```

Одоо хөтчид /dev/cd0 төхөрөөмжийн нэрийг ашиглан хандаж болох бөгөөд жишээ нь CD-ROM-г /mnt-д холбохдоо дараах тушаалыг бичих хэрэгтэй:

```
# mount -t cd9660 /dev/cd0 /mnt
```

`root` хэрэглэгчээр дараах тушаалыг ажиллуулж та шарагчийн SCSI хаягийг авч болно:

```
# camcontrol devlist
<MATSHITA CDRW/DVD UJDA740 1.00>   at scbus1 target 0 lun 0 (pass0,cd0)
```

Тэгэхээр `1,0,0` нь cdrecord(1) болон бусад SCSI програмтай ашиглах SCSI хаяг болох юм.

ATAPI/CAM болон SCSI системийн талаар дэлгэрэнгүй мэдээллийг atapicam(4) болон cam(4) гарын авлагын хуудаснуудаас лавлана уу.

# 19.7. Оптик зөөвөрлөгчийг (DVD-үүд) үүсгэж ашиглах нь

## 19.7.1. Танилцуулга

CD-тэй харьцуулахад DVD нь оптик зөөвөрлөгч хадгалалтын технологийн дараачийн үе юм. DVD нь ямар ч CD-ээс илүү өгөгдлийг агуулдаг бөгөөд одоогийн видео хэвлэлтийн стандарт болжээ.

Бичигддэг DVD гэж бидний нэрлэдэг DVD-үүдийн физик 5 бичигддэг хэлбэршүүлэлтийг тодорхойлж болно:

- DVD-R: Энэ нь бичигддэг DVD-ий анхны хэлбэршүүлэлт юм. DVD-R стандарт нь DVD хэлэлцүүлгээр тодорхойлогдсон бөгөөд энэ нь зөвхөн нэг удаа бичих хэлбэршүүлэлт юм.
- DVD-RW: Энэ нь DVD-R стандартын дахин бичигдэх хувилбар юм. DVD-RW нь ойролцоогоор 1000 удаа бичигдэх боломжтой.
- DVD-RAM: Энэ нь DVD хэлэлцүүлгийн дэмждэг бас дахин бичигддэг хэлбэршүүлэлт юм. DVD-RAM нь зөөгдөж болох хатуу хөтөч маягаар харагддаг. Гэхдээ энэ зөөвөрлөгч нь ихэнх DVD-ROM хөтчүүд болон DVD-Видео тоглуулагчуудтай нийцтэй биш байдаг; цөөн DVD бичигчид DVD-RAM хэлбэршүүлэлтийг дэмждэг. DVD-RAM-ийн хэрэглээний талаар илүү дэлгэрэнгүйг DVD-RAM ашиглах нь-с уншина уу.
- DVD+RW: Энэ нь DVD+RW холбооноос тодорхойлсон дахин бичигдэх хэлбэршүүлэлт юм. DVD+RW нь ойролцоогоор 1000 удаа бичигдэх боломжтой.
- DVD+R: Энэ хэлбэршүүлэлт нь DVD+RW хэлбэршүүлэлтийн нэг удаа бичих хувилбар юм.

Бичигддэг DVD-ий нэг давхарга нь 4,700,000,000 байт буюу 4.38 GB эсвэл 4485 MB (1 килобайт нь 1024 байт) хүртэлх мэдээлэл агуулж чадна.

Физик зөөвөрлөгч болон програмыг ялгаж ойлгох ёстой. Жишээ нь DVD-Видео нь дурын бичигддэг DVD физик зөөвөрлөгч DVD-R, DVD+R, DVD-RW гэх зэрэг уруу бичигдэж болох тусгай байршлын зураглал юм. Зөөвөрлөгчийн төрлийг сонгохын өмнө шарагч болон DVD-Видео тоглуулагч (дан тоглуулагч эсвэл компьютер дээрх DVD-ROM хөтөч) нь хэрэглэхээр төлөвлөж байгаа зөөвөрлөгчтэй нийцтэй эсэхийг шалгах хэрэгтэй.

### 19.7.2. Тохиргоо

DVD бичлэг хийхэд growisofs(1) програм ашиглагдана. Энэ тушаал нь dvd+rw-tools хэрэгслүүдийн (sysutils/dvd+rw-tools) нэг хэсэг юм. dvd+rw-tools нь DVD зөөвөрлөгчийн бүх төрлийг дэмждэг.

Эдгээр хэрэгслүүд нь төхөөрөмжүүд уруу хандахын тулд SCSI дэд системийг ашигладаг, тиймээс таны цөмд ATAPI/CAM дэмжлэг нэмэгдсэн байх ёстой. Хэрэв таны шарагч USB интерфэйс ашигладаг бол энэ нэмэлт нь хэрэггүй бөгөөд та USB төхөөрөмжүүдийн тохиргооны талаар илүү дэлгэрэнгүйг USB хадгалалтын төхөөрөмжүүд-с унших шаардлагатай.

Та мөн ATAPI төхөөрөмжүүдийн хувьд DMA хандалтыг идэвхжүүлэх ёстой бөгөөд дараах мөрийг /boot/loader.conf файлд нэмж үүнийг хийнэ:

```
hw.ata.atapi_dma="1"
```

dvd+rw-tools-г ашиглахаасаа өмнө өөрийн DVD шарагчтай холбоотой мэдээллийг dvd+rw-tools' тоног төхөөрөмжийн нийцтэй байдал хаягаас лавлах хэрэгтэй.

Хэрэв та график хэрэглэгчийн интерфэйсийг хүсэж байвал growisofs(1) болон бусад олон шарагч хэрэгслүүдийг хэрэглэгчид ашиглахад амар интерфэйсээр хангадаг K3b (sysutils/k3b) програмыг үзэх хэрэгтэй.

### 19.7.3. Өгөгдлийн DVD-үүдийг шарах нь

growisofs(1) тушаал нь mkisofs-ийн нүүр хэсэг юм, энэ нь шинэ файлын системийн байршлыг үүсгэхийн тулд mkisofs(8)-г дуудах бөгөөд DVD дээр бичих үйлдлийг гүйцэтгэнэ. Энэ нь та шарах процессоос өмнө өгөгдлийн дүрсийг үүсгэх хэрэггүй гэсэн үг юм.

DVD+R эсвэл DVD-R уруу өгөгдлийг /path/to/data сангаас шарахдаа дараах тушаалыг ашиглана:

```
# growisofs -dvd-compat -Z /dev/cd0 -J -R /path/to/data
```

Файлын системийг үүсгэхдээ `-J -R` тохируулгуудыг mkisofs(8)-д дамжуулдаг (энэ тохиолдолд Joilet болон Rock Ridge өргөтгөлүүдтэй ISO 9660 файлын систем). Дэлгэрэнгүйг mkisofs(8) гарын авлагын хуудаснаас лавлана уу.

`-Z` тохируулгыг ямар ч тохиолдолд (олон сессүүд эсвэл ганц сесс) эхний сессийг бичихдээ хэрэглэдэг. DVD төхөөрөмж */dev/cd0*-г өөрийн тохиргооны дагуу өөрчлөх хэрэгтэй. `-dvd-compat` параметр дискийг хаах бөгөөд бичилтийг нэмэх нь боломжгүй болох юм. Энэ нь DVD-ROM хөтчүүдтэй зөөвөрлөгчийн нийцтэй байдлыг илүүтэй хангах юм.

Мөн урьдчилан урласан дүрсийг шарах бас боломжтой, жишээ нь *imagefile.iso* дүрсийг шарахын тулд бид дараах тушаалыг ажиллуулна:

```
# growisofs -dvd-compat -Z /dev/cd0=imagefile.iso
```

Бичих хурдыг олж зөөвөрлөгч ба ашиглагдаж байгаа хөтчөөс хамаарч автоматаар тохируулах болно. Хэрэв та бичих хурдыг өөрчлөх хүсэлтэй байгаа бол `-speed=` параметрийг ашиглах хэрэгтэй. Дэлгэрэнгүй мэдээллийг growisofs(1) гарын авлагын хуудаснаас уншина уу.

Та өөртөө 4.38GB-аас их хэмжээтэй, ажиллагаатай файлтай болохын тулд mkisofs(8) болон бусад бүх програмд (жишээ нь growisofs(1)) `-udf -iso-level 3` гэсэн сонголтыг өгч UDF/ISO-9660 гибрид файлын системийг үүсгэсэн байх шаардлагатай. Энэ нь зөвхөн файлыг шууд диск рүү бичих эсвэл ISO дүрс файлыг үүсгэхэд л шаардлагатай. Энэ замаар үүсгэсэн дискийг зөвхөн UDF-г дэмждэг үйлдлийн системд ашиглагдах боломжтой байхаар mount_udf(8) хэрэгслийн тусламжтайгаар UDF файлын систем маягаар холбох ёстой бөгөөд тэгэхгүй бол энэ нь эвдэрсэн файлуудтай юм шиг харагдах болно.

Ийм ISO дүрс үүсгэхийн тулд:

```
% mkisofs -R -J -udf -iso-level 3 -o imagefile.iso /path/to/data
```

Диск рүү файлуудыг шууд бичихийн тулд:

```
# growisofs -dvd-compat -udf -iso-level 3 -Z /dev/cd0 -J -R
/path/to/data
```

Том файлууд аль хэдийн агуулсан ISO дүрс танд байгаа бол түүнийг диск рүү шарахад нэмэлт сонголтууд growisofs(1)-д шаардлагагүй.

Мөн хуучин хувилбарууд нь том файлыг дэмждэггүй учир та sysutils/cdrtools (mkisofs(8)-г агуулдаг) хэрэгслийн хамгийн сүүлийн хувилбартай байгаа эсэхээ шалгаарай. Хэрэв та асуудалтай тулгарвал хөгжүүлэлтийн хувилбар руу шилжээрэй, өөрөөр хэлбэл sysutils/cdrtools-devel рүү шилжээд mkisofs(8) -ийн гаран авлагын хуудсыг уншаарай.

### 19.7.4. DVD-Видео шарах нь

DVD-Видео нь ISO 9660 болон микро-UDF (M-UDF тодорхойлолтууд дээр тулгуурласан тусгай файлын байршлын зураглал юм. DVD-Видео нь бас өгөгдлийн бүтцийн тусгай шатлалыг үзүүлдэг бөгөөд энэ нь DVD-г зохиохын тулд multimedia/dvdauthor зэрэг тусгай програмыг та яагаад ашиглах хэрэгтэй болдгийн шалтгаан юм.

Хэрэв танд DVD-Видео файлын системийн дүрс байгаа бол ямар ч дүрсний нэгэн адил аргаар шарах хэрэгтэй. Өмнөх хэсгийн жишээнээс үзнэ үү. Хэрэв та DVD зохиолт хийсэн бөгөөд үр дүн нь жишээ нь /path/to/video санд байгаа бол DVD-Видеог шарахын тулд дараах тушаалыг ашиглах хэрэгтэй:

```
# growisofs -Z /dev/cd0 -dvd-video /path/to/video
```

`-dvd-video` тохируулга mkisofs(8)-д дамжуулагдах бөгөөд энэ нь DVD-Видео файлын системийн байршлын зураглал үүсгэхийг тушаах болно. Үүнээс гадна `-dvd-video` тохируулга нь growisofs(1)-ийн `-dvd-compat` тохируулгыг агуулдаг.

### 19.7.5. DVD+RW ашиглах нь

CD-RW-с ялгаатай нь шинэ DVD+RW нь ашиглагдахаа өмнө хэлбэршүүлэгдсэн байх ёстой. growisofs(1) нь шаардлагатай үед автоматаар үүнийг хийх бөгөөд энэ аргыг *зөвлөдөг* юм. Гэхдээ та `dvd+rw-format` тушаалыг ашиглан DVD+RW-г хэлбэршүүлж болно:

```
# dvd+rw-format /dev/cd0
```

Та энэ үйлдлийг зөвхөн нэг удаа хийх хэрэгтэй бөгөөд зөвхөн шинэ DVD+RW зөөвөрлөгчдийн хувьд хэлбэршүүлэх ёстойг санаарай. Дараа нь та DVD+RW-г дээрх хэсгүүдэд дурдсаны адил шарж болно.

Хэрэв та шинэ өгөгдлийг (зарим өгөгдлийг нэмэх биш бүр мөсөн шинэ файлын систем шарах) DVD+RW уруу шарахыг хүсэж байгаа бол түүнийг хоосон болгох шаардлагагүй юм, иймэрхүүгээр өмнөх бичилтэн дээрээ (шинэ сесс үүсгээд) дараад л бичих хэрэгтэй юм:

```
# growisofs -Z /dev/cd0 -J -R /path/to/newdata
```

DVD+RW хэлбэршүүлэлт нь өмнөх бичилтэд өгөгдлийг хялбараар нэмэх боломжийг олгодог. Энэ үйлдэл нь шинэ сессийг хуучин байгаатай нь нийлүүлэх бөгөөд энэ нь олон сесс бүхий бичилт биш юм. growisofs(1) нь зөөвөрлөгч дээр байгаа ISO 9660 файлын системийг *өсгөх (сунгах)* болно.

Жишээ нь хэрэв бид өөрсдийн урьдны DVD+RW уруу өгөгдөл нэмэхийг хүсвэл доор дурдсаныг ашиглах хэрэгтэй болно:

```
# growisofs -M /dev/cd0 -J -R /path/to/nextdata
```

Эхний сессийг шарахдаа бидний хэрэглэдэг mkisofs(8)-ийн адил тохируулгууд дараагийн бичилтүүдийн үеэр хэрэглэгдэх ёстой.

Хэрэв та DVD-ROM хөтчүүдтэй зөөвөрлөгчийн хувьд илүүтэй нийцтэй байхыг хүсвэл `-dvd-compat` тохируулгыг хэрэглэхийг хүсэж болох юм. DVD+RW тохиолдлын хувьд энэ нь таныг өгөгдөл нэмэхийг болиулж чадахгүй юм.

Хэрэв та ямар нэг шалтгаанаар зөөвөрлөгчийг хоосон болгохыг хүсвэл доор дурдсаныг хийх хэрэгтэй:

```
# growisofs -Z /dev/cd0=/dev/zero
```

### 19.7.6. DVD-RW ашиглах нь

DVD-RW нь дискний хоёр хэлбэршүүлэлтийг авдаг: нэмэгдсэн дараалсан хэлбэршүүлэлт болон хязгаарлагдмал дарж бичих хэлбэршүүлэлт юм. Анхдагчаар DVD-RW дискүүд нь дараалсан хэлбэршүүлэлтэд байдаг.

Шинэ DVD-RW нь хэлбэршүүлэлт хийгдэлгүйгээр шууд бичигдэж болдог, гэхдээ шинэ биш дараалсан хэлбэршүүлэлтэд байх DVD-RW нь шинэ эхний сесс бичигдэхээс өмнө хоосон болгогдсон байх шаардлагатай байдаг.

Дараалсан горим дахь DVD-RW-г хоослохдоо дараах тушаалыг ажиллуулна:

```
# dvd+rw-format -blank=full /dev/cd0
```

Бүр мөсөн хоослолт (`-blank=full`) 1х зөөвөрлөгч дээр нэг цаг орчим болно. Хэрэв DVD-RW нь Disk-At-Once (DAO) горимоор бичигдэх бол хурдан хоослолтыг `-blank` тохируулга ашиглан хийж болно. DVD-RW-г DAO горимд шарахын тулд дараах тушаалыг ашиглана:

```
# growisofs -use-the-force-luke=dao -Z /dev/cd0=imagefile.iso
```

`-use-the-force-luke=dao` тохируулгыг шаардах ёсгүй, учир нь growisofs(1) нь (хурдан хоосолсон) зөөвөрлөгчийг илрүүлэхийг бага оролдож DAO бичилтийг захиалах болно.

Яг үнэндээ дурын DVD-RW-ийн хувьд хязгаарлагдмал дарж бичих горимыг ашиглах хэрэгтэй бөгөөд энэ хэлбэршүүлэлт нь анхдагч нэмэгдсэн дараалсан хэлбэршүүлэлтээс илүү уян хатан байдаг.

Дараалсан DVD-RW дээр өгөгдлийг бичихдээ бусад DVD хэлбэршүүлэлтийн нэгэн адил зааврууды ашиглана:

```
# growisofs -Z /dev/cd0 -J -R /path/to/data
```

Хэрэв та зарим өгөгдлийг өөрийн урьдны бичлэгт нэмэхийг хүсвэл growisofs(1)-ийн `-M` тохируулгыг ашиглах хэрэгтэй болно. Гэхдээ хэрэв та нэмэгдсэн дараалсан горимд байгаа DVD-RW уруу өгөгдлийг нэмэх үйлдлийг хийвэл диск дээр шинэ сесс үүсгэгдэх бөгөөд үүний үр дүн нь олон сесс бүхий диск болох юм.

DVD-RW нь хязгаарлагдмал дарж бичих хэлбэршүүлэлтэд шинэ эхний сессээс өмнө хоосон болгогдох шаардлагагүй, та ердөө л дискийг `-Z` тохируулгатай дарж бичих хэрэгтэй бөгөөд энэ нь DVD+RW тохиолдолтой төстэй юм. Мөн диск дээр бичигдсэн байгаа ISO 9660 файлын системийг DVD+RW-тэй адил аргаар `-M` тохируулгын тусламжтай өсгөж (сунгаж) бас болно.

Үр дүн нь нэг сесс бүхий DVD болох юм.

DVD-RW-г хязгаарлагдмал дарж бичих хэлбэршүүлэлтэд оруулахдаа дараах тушаалыг ашиглах ёстой:

```
# dvd+rw-format /dev/cd0
```

Дараалсан хэлбэршүүлэлт уруу буцааж өөрчлөхдөө дараах тушаалыг ашиглана:

```
# dvd+rw-format -blank=full /dev/cd0
```

### 19.7.7. Олон сесс

Маш цөөн DVD-ROM хөтчүүд олон сесс бүхий DVD-үүдийг дэмждэг бөгөөд тэдгээр нь ихэнхдээ зөвхөн эхний сессийг уншдаг. DVD+R, DVD-R болон DVD-RW нь дараалсан хэлбэршүүлэлтдээ олон сессийг хүлээн авч чаддаг бөгөөд DVD+RW болон DVD-RW хязгаарлагдмал дарж бичих хэлбэршүүлэлтүүдийн хувьд олон сесс гэсэн ойлголт байдаггүй.

Дараалсан хэлбэршүүлэлтэд DVD+R, DVD-R эсвэл DVD-RW дээрх эхний (хаагдаагүй) сессийн дараа дараах тушаалыг ашиглаж дискэнд шинэ сесс үүсгэнэ:

```
# growisofs -M /dev/cd0 -J -R /path/to/nextdata
```

Энэ тушаалын мөрийг DVD+RW эсвэл DVD-RW-тэй цуг ашиглан хязгаарлагдмал дарж бичих горим дээр шинэ сессийг хуучин байгаатай нийлүүлэн өгөгдлийг нэмэх болно. Үр дүн нь нэг сесс бүхий диск болох юм. Энэ нь эдгээр зөөвөрлөгчүүд дээр эхний бичилтийн дараа өгөгдөл нэмэх арга юм.

Зөөвөрлөгч дээрх зарим зай нь сесс бүрийн хооронд сессийн төгсгөл болон эхлэлд хэрэглэгддэг. Тиймээс зөөвөрлөгчийн зайг оновчтой ашиглахын тулд их өгөгдөлтэй сессүүдийг нэмэх ёстой юм. Сессийн тоо DVD+R-ийн хувьд 154, DVD-R-ийн хувьд 2000 орчим, DVD+R хос давхаргын хувьд 127-оор хязгаарлагдана.

### 19.7.8. Дэлгэрэнгүй мэдээллийг

DVD-ийн талаар илүү мэдээллийг авахын тулд `dvd+rw-mediainfo /dev/cd0` тушаалыг хөтөч дотор диск байхад ажиллуулж болно.

dvd+rw-tools-н тухай дэлгэрэнгүй мэдээлэл нь growisofs(1) гарын авлагын хуудас, dvd+rw-tools вэб сайт болон cdwrite захидлын жагсаалтын архивуудаас олдож болно.

Бичигдсэн үр дүн эсвэл асуудалтай зөөвөрлөгчийн `dvd+rw-mediainfo` гаралт нь ямар ч асуудлын тайлангийн чухал хэсэг юм. Энэ гаралтгүйгээр танд туслах бараг л боломжгүй юм.

### 19.7.9. DVD-RAM ашиглах нь

#### 19.7.9.1. Тохиргоо

DVD-RAM бичигчид нь SCSI аль эсвэл ATAPI интерфэйстэй цуг ирдэг. ATAPI төхөөрөмжүүдийн хувьд DMA хандалт идэвхтэй болсон байх ёстой, дараах мөрийг /boot/loader.conf файлд нэмж үүнийг хийнэ:

```
hw.ata.atapi_dma="1"
```

#### 19.7.9.2. Зөөвөрлөгчийг бэлдэх нь

Өмнө нь бүлгийн танилцуулгад дурдсанаар DVD-RAM нь зөөврийн хатуу хөтөч маягаар харагддаг. Бусад хатуу дискүүдийн адил DVD- RAM нь ашиглагдаж эхлэхээсээ өмнө "бэлдэгдсэн" байх ёстой. Жишээн дээр дискний бүх зай стандарт UFS2 файлын системтэй ашиглагдана:

```
# dd if=/dev/zero of=/dev/acd0 bs=2k count=1
# bsdlabel -Bw acd0
# newfs /dev/acd0
```

DVD төхөөрөмж acd0-ийг өөрийн тохиргооны дагуу өөрчлөн ашиглах ёстой.

#### 19.7.9.3. Зөөвөрлөгчийг ашиглах нь

Дээрх үйлдлүүд DVD-RAM дээр хийгдсэний дараа үүнийг энгийн хатуу хөтчийн нэгэн адил холбож болно:

```
# mount /dev/acd0 /mnt
```

Үүний дараа DVD-RAM нь уншигдах бичигдэх боломжтой болно.

## 19.8. Уян дискүүдийг үүсгэж ашиглах нь

Өгөгдлийг уян дискүүд уруу хадгалах нь заримдаа ашигтай байдаг. Жишээ нь хэн нэгэнд нь шилжүүлж болдог ямар ч хадгалалтын зөөвөрлөгч байхгүй тохиолдолд эсвэл бага хэмжээний өгөгдлийг өөр компьютер уруу зөөх хэрэгцээ гарсан үед уян диск нь хэрэг болдог.

Энэ хэсэг нь FreeBSD дээр уян дискийг хэрхэн ашиглах талаар тайлбарлах болно. Энд 3.5 инчийн DOS уян дискүүдийг хэлбэршүүлж ашиглах талаар үндсэндээ тайлбарлах бөгөөд гэхдээ энэ ойлголт нь бусад уян дискний хэлбэршүүлэлттэй төстэй юм.

### 19.8.1. Уян дискүүдийг хэлбэршүүлэх нь

#### 19.8.1.1. Төхөөрөмж

Уян дискнүүдэд бусад төхөөрөмжүүдийн адил /dev сан дахь оруулгуудаар ханддаг. Түүхий уян дискэнд хандахын тулд /dev/fdN-г ердөө л ашиглах хэрэгтэй.

#### 19.8.1.2. Хэлбэршүүлэх нь

Уян дискийг ашиглахаасаа өмнө доод түвшний хэлбэршүүлэлт хийсэн байх хэрэгтэй. Үүнийг ихэвчлэн үйлдвэрлэгч хийдэг боловч хэлбэршүүлэлт нь зөөвөрлөгчийн бүрэн бүтэн байдлыг шалгах нэг сайн арга юм. Илүү том (эсвэл жижиг) дискний хэмжээг хүчлэн ашиглах боломжтой байдаг боловч 1440kB хэмжээнд зориулагдан ихэнх уян диск хийгдсэн байдаг.

Уян дискэнд доод түвшний хэлбэршүүлэлт хийхийн тулд та fdformat(1)-г ашиглах хэрэгтэй. Энэ хэрэгсэл нь төхөөрөмжийн нэрийг нэмэлт өгөгдөл маягаар оруулахыг хүлээж байдаг.

Алдааны мэдэгдлийг тэмдэглэж аваарай, учир нь эдгээр нь дискийг сайн эсвэл муу эсэхийг тодорхойлоход туслах болно.

##### 19.8.1.2.1. Уян дискнүүдийг хэлбэршүүлэх нь

/dev/fdN төхөөрөмжүүдийг ашиглан уян дискийг хэлбэршүүлэх хэрэгтэй. Шинэ 3.5 инч диск өөрийн хөтөч уруугаа хийгээд дараах тушаалыг ажиллуул:

```
# /usr/sbin/fdformat -f 1440 /dev/fd0
```

### 19.8.2. Дискний шошго

Дискэнд доод түвшний хэлбэршүүлэлт хийсний дараа танд диск дээр шошго тавих хэрэгтэй болно. Энэ дискний шошго нь дараа нь устгагдах боловч дискний хэмжээ болон геометрийг дараа нь тодорхойлоход системд хэрэг болдог.

Шинэ дискний шошго нь бүхэл дискийг хамарч уян дискний геометрийн тухай бүх л зөв мэдээллийг агуулах болно. Дискний шошгоны геометрийн утгууд нь /etc/disktab файлд жагсаагдсан байдаг.

Та одоо ингэж bsdlabel(8)-г ажиллуулж болно:

```
# /sbin/bsdlabel -B -w /dev/fd0 fd1440
```

### 19.8.3. Файлын систем

Одоо уян дискэнд дээд түвшний хэлбэршүүлэлт хийхэд бэлэн боллоо. Энэ нь дискийг FreeBSD унших болон түүнд бичих боломжийг олгох шинэ файлын системийг диск дээр байрлуулах болно. Шинэ файлын системийг үүсгэсний дараа дискний шошго устгагдах бөгөөд хэрэв та дискийг дахин хэлбэршүүлэхийг хүсвэл дискний шошгыг дахин үүсгэх шаардлагатай болно.

Уян дискний файлын систем нь UFS эсвэл FAT хоёрын аль нэг нь байна. FAT нь ерөнхийдөө уян дискнүүдийн хувьд илүү дээр сонголт байдаг.

Уян диск дээр шинэ файлын системийг байрлуулахын талд дараах тушаалыг ажиллуулна:

```
# /sbin/newfs_msdos /dev/fd0
```

Диск одоо ашиглахад бэлэн боллоо.

### 19.8.4. Уян дискийг ашиглах нь

Уян дискийг ашиглахын тулд mount_msdosfs(8) тушаалаар холбох хэрэгтэй. Мөн портын цуглуулгаас emulators/mtools-г ашиглаж бас болох юм.

## 19.9. Өгөгдлийн соронзон хальснууд үүсгэж ашиглах нь

Гол соронзон хальс зөөвөрлөгчүүд нь 4мм, 8мм, QIC, мини-хайрцаг болон DLT юм.

### 19.9.1. 4мм (DDS: Digital Data Storage)

4мм соронзон хальснууд нь QIC-г халж ажлын станцын нөөц зөөвөрлөгч болон сонгогдож байна. Conner компани нь QIC хөтчүүдийн тэргүүлэх үйлдвэрлэгч Archive-г худалдаж авч дараа нь QIC хөтчүүдийг үйлдвэрлэхээ зогсоосноор энэ чиг хандлага нь илүү хурдассан юм. 4мм хөтчүүд нь жижиг, чимээгүй боловч 8мм хөтчүүд шиг найдвартай ажиллагаагаараа алдартай биш юм. Хайрцагнууд нь үнэтэй биш бөгөөд 8мм-ийн хайрцагнуудаас бага (3 x 2 x 0.5 инч, 76 x 51 x 12 мм) юм. 4мм соронзон хальс нь 8мм-ийн нэгэн адил шалтгаанаар толгой нь богино настай бөгөөд хоёулаа мушгиа сканыг ашигладаг.

Эдгээр хөтчүүд дээрх өгөгдлийн дамжуулах чадвар нь ~150 kB/s-с эхэлж ~500 kB/s хүрнэ. Өгөгдлийн багтаамж 1.3 GB-с эхэлж 2.0 GB хүрнэ. Тоног төхөөрөмжийн шахалт энэ хөтчүүдийн ихэнхэд байх бөгөөд энэ нь багтаамжийг ойролцоогоор хоёр дахин нэмэгдүүлдэг. Олон хөтөч бүхий соронзон хальсны сан (library) автомат соронзон хальс солигчтой нэг кабинетийн хувьд 6 хөтөчтэй байж болно. Сангийн багтаамж нь 240 GB хүрнэ.

DDS-3 стандарт нь одоогоор 12 GB (эсвэл 24 GB шахагдсан) багтаамжтай соронзон хальсыг дэмждэг.

4мм хөтчүүд нь 8мм-ийн хөтчүүдийн нэгэн адил мушгиа хайлтыг хэрэглэдэг. Мушгиа хайлт хийхийн бүх ашигтай тал болон сул талууд нь 4мм болон 8мм-ийн хөтчүүдийн аль алинд нь хамаардаг.

Соронзон хальснууд нь 2000 удаагийн ашиглалт эсвэл 100 бүрэн нөөцлөлтийн дараа хэрэглээнээс гарах ёстой.

### 19.9.2. 8мм (Exabyte)

8мм соронзон хальснууд нь хамгийн нийтлэг SCSI соронзон хальсны хөтчүүд юм; тэдгээр нь соронзон хальснууд солих хамгийн сайн сонголт болдог. Бараг сайт бүр Exabyte 2 GB 8мм-ийн соронзон хальсны хөтөчтэй байдаг. 8мм-ийн хөтчүүд нь найдвартай, хэрэглэхэд амар, чимээгүй байдаг. Хайрцагнууд нь хямд, жижиг (4.8 x 3.3 x 0.6 инч; 122 x 84 x 15 мм) байдаг. 8мм-ийн соронзон хальсны нэг сул тал нь толгойнуудын дагуух соронзон хальсны харьцангуй хөдөлгөөний өндөр хувиас болоод харьцангуй богино толгой ба соронзон хальсны амьдрах хугацаатай байдаг явдал юм.

Өгөгдөл дамжуулах чадвар нь ~250 kB/s-аас ~500 kB/s хүртэл байна. Өгөгдлийн хэмжээ нь 300 MB-аас эхэлж 7 GB хүрнэ. Тоног төхөөрөмжийн шахалт энэ хөтчүүдийн ихэнхэд байх бөгөөд энэ нь багтаамжийг ойролцоогоор хоёр дахин нэмэгдүүлдэг. Эдгээр хөтчүүд нь нэг буюу эсвэл нэг кабинетдаа 6 хөтөч болон 120 соронзон хальстай олон хөтөч бүхий соронзон хальсны сан (library) хэлбэрээр байдаг. соронзон хальснууд нь автоматаар солигддог. Сангийн багтаамж 840+ GB хүрнэ.

Exabyte "Mammoth" загвар нь нэг соронзон хальс дээр 12 GB ((24 GB шахалттайгаар) дэмждэг бөгөөд ердийн соронзон хальсны хөтчөөс ойролцоогоор хоёр дахин үнэтэй байдаг.

Өгөгдөл нь соронзон хальс уруу мушгиа скан ашиглагдан бичигддэг, толгойнууд нь зөөвөрлөгч уруу өнцгөөр байрладаг (ойролцоогоор 6 градус). Соронзон хальс нь толгойнуудыг барьж байдаг дамрын 270 градус орчим ороодог. Соронзон хальс дамар дээгүүр гулгаж байхад дамар нь эргэж байдаг. Үр дүнд нь өгөгдлийн өндөр нягтрал болон соронзон хальсны дагуу нэг ирмэгээс нөгөө уруу өнцөгдсөн ойрхон багцалсан замууд үүсэх болно.

### 19.9.3. QIC

QIC-150 соронзон хальснууд болон хөтчүүд нь магадгүй хамгийн нийтлэг соронзон хальсны хөтөч, зөөвөрлөгч юм. QIC соронзон хальсны хөтчүүд нь хамгийн хямд "нухацтай" нөөцлөлтийн хөтчүүд юм. Сул тал нь зөөвөрлөгчийн үнэ байдаг. QIC соронзон хальснууд нь 8мм болон 4мм соронзон хальснуудтай харьцуулахад GB өгөгдлийн хадгалалтын хувьд 5 дахин үнэтэй байдаг. Гэхдээ таны хэрэглээнд цөөн (half-dozen) соронзон хальснууд хангалттай бол QIC нь магадгүй зөв сонголт болж болох юм. QIC нь *хамгийн* нийтлэг соронзон хальсны хөтөч юм. Сайт бүр ямар нэг хэмжээний QIC хөтөчтэй байдаг. QIC нь физикийн хувьд төстэй (заримдаа адил) соронзон хальснуудад их хэмжээний нягтралтай байдаг. QIC хөтчүүд нь чимээгүй биш юм. Эдгээр хөтчүүд нь өгөгдлийг бичиж эхлэхээсээ өмнө дуутайгаар хайдаг бөгөөд унших, бичих эсвэл хайхдаа мэдэгдэхүйц дуутай байдаг. QIC соронзон хальснууд нь 6 x 4 x 0.7 инч (152 x 102 x 17 мм) хэмжээтэй байдаг.

Өгөгдлийн дамжуулах чадвар ~150 kB/s-с ~500 kB/s хүртэл байна. Өгөгдлийн багтаамж 40 MB-с 15 GB хүртэл байна. Шинэ QIC хөтчүүдийн ихэнхэд тоног төхөөрөмжийн шахалт байдаг. QIC хөтчүүд нь бага суулгагддаг; тэдгээр нь DAT хөтчүүдээр шахагдсан юм.

Өгөгдөл нь соронзон хальс уруу замаар бичигддэг. Замууд нь соронзон хальсны зөөвөрлөгчийн нэг төгсгөлөөс нөгөө уруу урт тэнхлэгийн дагуу байдаг. Замуудын тоо болон замын өргөн соронзон хальсны багтаамжаас хамаарч өөр өөр байдаг. Бүх шинэ хөтчүүдийн ихэнх нь хамгийн багадаа бодоход уншилтын (ихэнхдээ бас бичилтийн хувьд)

хуучинтайгаа нийцтэй байдаг. QIC нь өгөгдлийн аюулгүй байдлын хувьд нэлээн нэр хүндтэй байдаг (механизм нь хөтчүүдийг мушгиа скан хийснээс илүү хялбар бөгөөд хүчирхэг байдаг).

5,000 нөөцлөлтийн дараа соронзон хальснуудыг ашиглахаа болих шаардлагатай.

### 19.9.4. DLT

DLT нь энд жагсаагдсан бүх хөтчийн төрлүүдээс хамгийн хурдан өгөгдөл дамжуулах чадвартай байдаг. 1/2" (12.5мм) соронзон хальс нь ганц дамартай хайрцагт (4 x 4 x 1 инч; 100 x 100 x 25 мм) байдаг. Хайрцаг нь нэг талаараа ганхах хаалгатай байдаг. Хөтчийн механизм соронзон хальсны тэргүүнийг гаргаж авахын тулд энэ хаалгыг онгойлгодог. Соронзон хальсны тэргүүн нь зууван нүхтэй байх бөгөөд хөтөч соронзон хальсыг "дэгээдэхдээ" үүнийг ашигладаг. Авах (take-up) дамар нь соронзон хальсны хөтчийн дотор байрладаг. Энд жагсаагдсан бусад соронзон хальсны хайрцагнууд (9 замтай соронзон хальснууд нь жич юм) нь соронзон хальсны хайрцган дотор байрлах хангах болон take-up дамруудтай байдаг.

Өгөгдөл дамжуулах чадвар нь ойролцоогоор 1.5 MB/s бөгөөд 4мм, 8м, QIC соронзон хальсны хөтчүүдийн дамжуулах чадвараас 3 дахин их байдаг. Өгөгдлийн багтаамж нь нэг хөтчийн хувьд 10 GB-аас 20 GB хүрдэг. Хөтчүүд нь олон соронзон хальс солигчид болон олон соронзон хальс хэлбэрээр байдаг, олон хөтөч бүхий соронзон хальсны сангууд нь нийтдээ 50 GB-с 9 TB хүртэл хадгалалт бүхий 5-аас 900 хүртэл соронзон хальснуудыг 1-ээс 20 хүртэлх хөтчүүд дээр агуулдаг.

Шахалттай бол DLT Төрөл 4 хэлбэршүүлэлт нь 70 GB багтаамжийг дэмждэг.

Өгөгдөл нь соронзон хальсны замууд дээр аялалын чиглэлийн дагуу (QIC соронзон хальснуудын адил) зэрэгцээгээр бичигддэг. Хоёр зам нэг удаа бичигддэг. Унших/бичих толгойны амьдрах хугацаа харьцангуй урт байдаг; соронзон хальс хөдлөхөө болиход толгой болон соронзон хальсны хоорондын хамаатай хөдөлгөөн байхгүй болно.

### 19.9.5. AIT

AIT нь Sony-гоос гаргасан шинэ хэлбэршүүлэлт бөгөөд нэг соронзон хальсны хувьд 50 GB хүртэл (шахалттайгаар) мэдээллийг агуулж чадна. Соронзон хальснууд нь санах ойн бичил схемүүдийг агуулдаг бөгөөд эдгээр нь соронзон хальсны агуулгын индексийг хадгалдаг. Бусад соронзон хальснуудын хувьд хэд хэдэн минут шаардагддаг бол харин энэ индексийг соронзон хальсны хөтөч соронзон хальс дээрх файлуудын байрлалыг тодорхойлохын тулд маш хурдан уншдаг. SAMS:Alexandria зэрэг програм нь соронзон хальсны санах ойн бичил схемтэй шууд холбогдон агуулгыг дэлгэцэд гаргаж, ямар файлууд аль соронзон хальс уруу нөөцлөгдсөнийг тодорхойлж, зөв соронзон хальсыг олон ачаалж соронзон хальснаас өгөгдлийг сэргээн дөч болон түүнээс дээш AIT соронзон хальсны сангуудыг ажиллуулж чаддаг.

Үүнтэй адил сангууд $20,000 хавьцаа үнэ хүрч тэдгээрийг сонирхогчдын зах зээлээс бага зэрэг шахдаг байна.

### 19.9.6. Шинэ соронзон хальсыг анх удаа ашиглах нь

Анхны удаа шинэ, хов хоосон соронзон хальсыг уншихыг оролдвол амжилтгүй болно. Консолын мэдэгдлүүд үүнтэй төстэй байна:

```
sa0(ncr1:4:0): NOT READY asc:4,1
sa0(ncr1:4:0):  Logical unit is in process of becoming ready
```

Соронзон хальс нь Identifier Block буюу Танигч Блокийг (block number 0) агуулаагүй байна. Бүх QIC соронзон хальснууд нь QIC-525 стандартыг хэрэглэж эхэлснээс хойш Танигч Блокийг соронзон хальсанд бичдэг. Хоёр шийдэл байдаг:

- `mt fsf 1` тушаал нь Танигч Блокийг соронзон хальс уруу бичихийг соронзон хальсны хөтчид хэлнэ.
- Нүүрний товчлуурыг ашиглан соронзон хальсыг гаргаж авна.

  Соронзон хальсыг дахин хийгээд түүн уруу өгөгдлийг `dump` хийнэ.

  `dump` тушаал нь `DUMP: End of tape detected` гэж мэдэгдэх бөгөөд консол `HARDWARE FAILURE info:280 asc:80,96` гэж харуулна.

  `mt rewind` тушаал ашиглан соронзон хальсыг буцаана.

  Дараа дараачийн соронзон хальсны үйлдлүүд амжилттай болно.

## 19.10. Уян диск уруу нөөцлөх

### 19.10.1. Өөрийн өгөгдлийг нөөцлөхийн тулд би уян дискнүүдийг ашиглаж болох уу?

Уян дискнүүд нь нөөц хийхэд тийм ч тохиромжтой зөөвөрлөгч биш юм, учир нь:

- Энэ зөөвөрлөгч нь найдваргүй, ялангуяа урт хугацааны туршид найдваргүй байдаг.
- Нөөцлөх болон буцааж сэргээх нь их удаан байдаг.
- Тэдгээр нь маш хязгаарлагдмал багтаамжтай (бүхэл бүтэн хатуу дискийг хэдэн арван уян дискэнд нөөцлөх нь их олон өдөр шаардана).

Гэхдээ хэрэв танд өөрийн өгөгдлийг нөөцлөх өөр ямар ч арга байхгүй бол уян дискнүүдэд нөөцлөх нь нөөц хийхгүй байснаас хамаагүй дээр юм.

Хэрэв та уян дискнүүд ашиглах шаардлагатай болсон бол сайн чанарынхыг ашигласан эсэхээ шалгах хэрэгтэй. Оффис дээр чинь хоёр жил хэвтсэн уян дискнүүд муу сонголт болох юм. Нэр хүндтэй үйлдвэрлэгчээс гаргасан шинэ дискнүүдийг ашиглах нь зүйтэй юм.

### 19.10.2. Тэгэхээр би өөрийн өгөгдлийг уян диск уруу хэрхэн нөөцлөх вэ?

Уян диск уруу нөөцлөх хамгийн шилдэг арга нь tar(1) тушаалыг `-M` (олон эзлэхүүн)

тохируулгатайгаар ашиглах явдал юм. Энэ нь олон уян дискнүүдэд нөөцлөх боломжийг олгодог.

Тухайн сан болон дэд сан доторх бүх файлуудыг нөөцлөхийн тулд үүнийг ашиглах хэрэгтэй (`root` хэрэглэгчээр):

```
# tar Mcvf /dev/fd0 *
```

Эхний уян диск дүүрсний дараа tar(1) нь дараагийн эзлэхүүнийг оруулахыг хүсэх болно (учир нь tar(1) нь зөөвөрлөгчөөс хамааралгүй бөгөөд эзлэхүүнүүдэд ханддаг; энд уян дискийг хэлж байна).

```
Prepare volume #2 for /dev/fd0 and hit return:
```

Энэ нь заагдсан файлууд архивлагдах хүртэл (эзлэхүүний дугаар нэмэгдэн) давтагдах болно.

### 19.10.3. Би өөрийнхөө нөөцүүдийг шахаж болох уу?

Харамсалтай нь tar(1) нь олон эзлэхүүн бүхий архивуудын хувьд `-z` тохируулгыг ашиглахыг зөвшөөрдөггүй. Мэдээж та бүх файлуудыг gzip(1) хийж тэдгээрийг уян диск уруу tar(1) хийж дараа нь файлуудыг дахин gunzip(1) хийж болно!

### 19.10.4. Би өөрийн нөөцүүдийг хэрхэн сэргээх вэ?

Бүхэл архивыг сэргээхдээ дараах тушаалыг ашиглана:

```
# tar Mxvf /dev/fd0
```

Зөвхөн заагдсан файлуудыг сэргээх хоёр арга байдаг бөгөөд та тэдгээрийг ашиглаж болно. Эхлээд та эхний дискнээс эхлээд дараах тушаалыг ашиглана:

```
# tar Mxvf /dev/fd0 filename
```

tar(1) хэрэгсэл нь шаардлагатай файлыг олох хүртлээ дараа дараагийн уян дискнүүдийг хийхийг танаас хүсэх болно.

Өөрөөр, хэрэв та файл нь яг аль уян диск дээр байгааг мэдэж байвал ердөө л тэр уян дискийг оруулж дээрхтэй адил тушаалыг ашиглах хэрэгтэй. Хэрэв уян диск дээрх эхний файл нь өмнөх диск дээрхийн үргэлжлэл бол таныг асуугаагүй байсан ч гэсэн tar(1) үүнийг сэргээж чадахгүй гэж танд анхааруулах болно!

## 19.11. Нөөцлөх стратегууд

Нөөцлөх төлөвлөгөөг боловсруулах эхний шаардлага нь дараах бүх асуудлуудыг хамарсан эсэхийг шалгах явдал юм:

- Дискний гэмтэл
- Санамсаргүй файл устгалт
- Санамсаргүй файлын эвдрэл
- Газар дээрх нөөцүүд байвал тэдгээрийг оруулаад машины бүрэн сүйрэл (өөрөө хэлбэл гал).

Эдгээр асуудал бүрийг шал өөр техникээр шийдэснээр зарим системүүдийг илүүтэй ажиллуулах төгс боломж байж болох юм. Маш бага үнэ цэнэтэй өгөгдөл бүхий чанга хувийн системүүдийг тооцохгүй юм бол нэг техник нь бүх асуудлуудыг хамрах нь бараг боломжгүй юм.

Хэрэгслийн хайрцаг дахь зарим нэг техникүүдийг дурдвал:

- Бүх системийн архивууд нь сайтаас гадна байнгын зөөвөрлөгчид нөөцлөгдөнө. Энэ нь дээр дурдсан бүх асуудлуудаас хамгаалах боловч сэргээх нь тохиромжгүй удаан байдаг. Та нөөцүүдийн хуулбаруудыг газар дээр нь ба/эсвэл шууд авч болохоор хадгалж болох боловч файлуудыг сэргээх нь ялангуяа зөвшөөрөгдөөгүй хэрэглэгчдэд бас л тохиромжгүй хэвээр байдаг.
- Файлын системийн хормын хувилбарууд. Энэ нь яг үнэндээ зөвхөн санамсаргүйгээр файлыг устгасан тохиолдолд тустай, гэхдээ энэ нь тийм тохиолдолд *маш* тус болохуйц байдаг бөгөөд хурдан, ажиллахад хялбар байдаг.
- Бүх файлын систем ба/эсвэл дискнүүдийн хуулбарууд (бүхэл машины үе үе давтагдах rsync(1)). Энэ нь ерөнхийдөө онцгой шаардлага бүхий сүлжээнүүдэд хамгийн ашигтай байдаг. Дискний эвдрэл, гэмтлийн эсрэг ерөнхий хамгаалалтын хувьд энэ нь ихэвчлэн RAID-с чанарын хувьд муу байдаг. Санамсаргүйгээр устгасан файлуудыг сэргээхэд энэ нь UFS хормын хувилбартай дүйцэхээр боловч та алийг дээдэлдгээс хамаарах юм.
- RAID. Диск эвдрэх, гэмтэх үед зогсох хугацааг багасгаж зайлсхийдэг. Маш бага шаардлагатай хэдий ч дискний эвдрэлүүдтэй илүүтэй зууралдах (учир нь та олон дисктэй) хэрэгтэй болдог.
- Файлуудын байрлалыг (хурууны хээ) шалгах. Үүнд mtree(8) хэрэгсэл их ашигтай байдаг. Энэ нь нөөцлөх техник биш боловч танд өөрийн нөөцүүддээ хандахаар болох үед та хэрэг болох болно. Энэ нь сайтаас гаднах шууд бус нөөцүүдийн хувьд ялангуяа чухал бөгөөд үе үе шалгагдаж байх ёстой.

Үүнээс илүү олон техникийг бодож олох нь амархан бөгөөд тэдгээрийн ихэнх нь дээр дурдсан техникүүдийн өөр хувилбарууд юм. Тусгайлсан шаардлагууд нь ихэвчлэн тусгайлсан техникт хүргэдэг (жишээ нь шууд ажиллаж байгаа мэдээллийн баазыг нөөцлөх нь зөвхөн мэдээллийн санд зориулагдсан аргыг дундын шат болгон ихэвчлэн ашиглахыг шаарддаг). Ямар аюулуудаас та хамгаалахыг хүсэж байгаа болон тэдгээр тус бүртэй хэрхэн ажиллахаа мэдэх нь чухал юм.

## 19.12. Нөөцлөлтийн үндсүүд

Гурван гол нөөцлөх програм бол dump(8), tar(1) болон cpio(1) юм.

### 19.12.1. Dump ба Restore

Уламжлалт UNIX® нөөцлөх програмууд нь dump ба restore юм. Тэдгээр нь файлын системүүдээр үүсгэгдсэн файлууд, холбоосууд болон сангуудын хийсвэр ойлголтуудын доор хөтчүүд дээр дискний блокуудын цуглуулга хэлбэрээр ажилладаг. Бусад нөөцлөх програмуудаас ялгаатай нь dump нь төхөөрөмж дээрх бүхэл файлын системийг нөөцөлдөг. Файлын системийн зөвхөн хэсгийг эсвэл нэгээс илүү файлын систем дагуу байрлах сангийн модыг энэ нь нөөцөлж чаддаггүй. dump нь файлууд болон сангуудыг соронзон хальс уруу бичдэггүй, харин файлууд болон сангуудаас тогтох түүхий өгөгдлийн блокуудыг бичдэг. Өгөгдлийг задлахад хэрэглэгдэхдээ restore нь анхдагчаар түр зуурын файлуудыг /tmp/ санд хадгалдаг. Хэрэв та жижиг хэмжээний /tmp сан бүхий нөөц сэргээх диск дээрээс ажиллаж байгаа бол сэргээлтийг амжилттай гүйцээхийн тулд илүү чөлөөтэй зайтай сан руу TMPDIR орчны хувьсагчийг зааж өгөх хэрэгтэй байж болох юм.

Хэрэв та өөрийн root сандаа dump-г хэрэглэвэл та /home, /usr эсвэл бусад олон сангуудыг нөөцлөхгүй бөгөөд үүний учир бол эдгээр нь ихэвчлэн бусад файлын системүүдийн холбох цэгүүд буюу эсвэл тэдгээр файлын системүүд уруу заасан симболын холбоосууд байдаг.

dump нь өөрийн хөгжлийн эхний өдрүүд болох AT&T UNIX-ийн 6-р хувилбараас (1975 он орчим) үлдсэн кодтой байдаг. Анхдагч параметрүүд нь өнөөдөр байгаа өндөр нягтралтай (62,182 ftpi хүртэл) зөөвөрлөгчид биш 9 зам (6250 bpi) бүхий соронзон хальсанд тохирдог. Одоогийн соронзон хальсны хөтчүүдийн багтаамжийг хэрэглэхийн тулд эдгээр анхдагчуудыг тушаалын мөрөөс дарж өөрчлөх ёстой.

Мөн сүлжээгээр өөр компьютерт холбогдсон соронзон хальсны хөтөч уруу өгөгдлийг rdump болон rrestore тушаал ашиглан нөөцлөх боломжтой байдаг. Энэ хоёр програм нь алсын соронзон хальсны хөтчид хандахдаа rcmd(3) болон ruserok(3)-д тулгуурладаг. Тиймээс нөөцлөлтийг хийж байгаа хэрэглэгч алсын компьютерийн .rhosts файл дотор жагсаагдсан байх ёстой. rdump болон rrestore тушаалуудад өгөх нэмэлт өгөгдлүүд нь алсын компьютер дээр ашиглаж болохоор байх ёстой. komodo гэж нэрлэгдсэн Sun уруу холбогдсон Exabyte соронзон хальсны хөтөч уруу FreeBSD компьютераас rdump хийхдээ дараах тушаалыг ашиглана:

```
# /sbin/rdump 0dsbfu 54000 13000 126 komodo:/dev/nsa8 /dev/da0a 2>&1
```

Болгоомжлоорой: .rhosts танин нэвтрэлтийг зөвшөөрөх нь аюулгүй байдлын асуудлууд үүсгэдэг. Өөрийн нөхцөл байдлыг анхааралтай үнэлэх хэрэгтэй.

Мөн dump болон restore тушаалыг илүү аюулгүй загвараар ssh дээгүүр ашиглах бас боломжтой байдаг.

*Жишээ 24. ssh-ээр dump-г ашиглах нь*

```
# /sbin/dump -0uan -f - /usr | gzip -2 | ssh -c blowfish \
          targetuser@targetmachine.example.com dd of=/mybigfiles/dump-usr-l0.gz
```

Эсвэл dump-ийн дотор байдаг нэг арга болох RSH орчны хувьсагчийг тохируулан ашиглаж болно:

*Жишээ 25. ssh-ээр RSH тохируулан dump-г ашиглах нь*

```
# env RSH=/usr/bin/ssh /sbin/dump -0uan -f
targetuser@targetmachine.example.com:/dev/sa0 /usr
```

### 19.12.2. tar

tar(1) нь бас AT&T UNIX-ийн 6-р хувилбараас (1975 он орчим) эхтэй. tar нь файлын системтэй хамт ажилладаг; энэ нь файлууд болон сангуудыг соронзон хальс уруу бичдэг. tar нь cpio(1) -д байдаг бүрэн хэмжээний тохируулгуудыг дэмждэггүй боловч энэ нь cpio-ийн хэрэглэдэг ховор тушаалын дамжуулах хоолойг шаарддаггүй.

komodo гэж нэрлэгдсэн Sun уруу холбогдсон Exabyte соронзон хальсны хөтөч уруу tar хийхдээ дараах тушаалыг ашиглана:

```
# tar cf - . | rsh komodo dd of=tape-device obs=20b
```

Хэрэв та сүлжээгээр нөөцлөхийн аюулгүй байдалд санаа зовж байгаа бол rsh-ийн оронд ssh тушаалыг ашиглах хэрэгтэй.

### 19.12.3. cpio

cpio(1) нь соронзон зөөвөрлөгчид зориулагдсан UNIX®-ийн анхны файл солилцох соронзон хальсны програм юм. cpio нь (бусад олон тохируулгуудын дотроос) байт солилцохыг хийх, хэд хэдэн төрлийн архивын хэлбэршүүлэлтүүдийг бичих болон өгөгдлийг өөр програм уруу хоолойгоор гаргах зэрэг тохируулгуудтай байдаг. Сүүлийн боломж нь cpio-г суулгацын зөөвөрлөгчийн хувьд сайн сонголт болгодог. cpio нь сангийн модоор хэрхэн явахаа мэддэггүй бөгөөд файлуудын жагсаалтыг stdin-ээс хангаж өгөх ёстой.

cpio нь сүлжээгээр нөөцлөхийг дэмждэггүй. Та дамжуулах хоолой болон rsh-ийг ашиглаж өгөгдлийг алсын соронзон хальсны хөтөч уруу илгээж болно.

```
# for f in directory_list; do
find $f >> backup.list
done
# cpio -v -o --format=newc < backup.list | ssh user@host "cat > backup_device"
```

*directory_list* нь таны нөөцлөхийг хүссэн сангуудын жагсаалт бол *user@host* нь нөөцлөлтийг хийж байгаа хэрэглэгч/хостын нэрийн хослол бөгөөд *backup_device* нь нөөцлөлт хийгдэх төхөөрөмж (өөрөөр хэлбэл /dev/nsa0) юм.

### 19.12.4. pax

pax(1) нь tar болон cpio тушаалын IEEE/POSIX®-ийн хариулт юм. Жил ирэх тутам tar болон cpio тушаалуудын төрөл бүрийн хувилбарууд нь хоорондоо мэдэгдэхүйц нийцгүй болж ирсэн юм. Тэдгээртэй тэмцэж бүр мөсөн стандартчилахын оронд POSIX® нь архивын шинэ хэрэгсэл үүсгэсэн юм. pax нь өөрийн шинэ хэлбэршүүлэлтээс гадна cpio болон tar тушаалуудын төрөл бүрийн хэлбэршүүлэлтийн ихэнхийг уншиж бичихийг оролддог. Энэ тушаалын олонлог нь tar тушаалынхтай биш харин cpio тушаалынхтай илүү төстэй байдаг.

### 19.12.5. Amanda

Amanda (Advanced Maryland Network Disk Archiver) нь нэг програм биш клиент/сервер програм юм. Amanda сервер нь ганц соронзон хальсны хөтөч уруу Amanda клиенттэй, Amanda сервер уруу сүлжээгээр холбогдсон дурын тооны компьютерийг нөөцлөх болно. Хэд хэдэн том дискнүүд бүхий сайтууд дээр байдаг нийтлэг асуудлууд нь өгөгдлийг соронзон хальс уруу нөөцлөхөд шаардагдах хугацаа нь уг үйлдлийн хувьд зориулагдсан хугацаанаас илүү гарах явдал юм. Amanda нь энэ асуудлыг шийддэг. Amanda нь хэд хэдэн файлын системүүдийг нэг зэрэг нөөцлөхдөө "holding disk буюу хүлээгдэх диск"ийг ашигладаг. Amanda нь "архивын олонлогууд" буюу Amanda-ийн тохиргооны файлд жагсаагдсан бүх файлын системүүдийн бүрэн нөөцлөлтүүдийг үүсгэхэд хэсэг хугацаанд хэрэглэгдсэн бүлэг соронзон хальснуудыг үүсгэдэг. "Архивын олонлогууд" нь бас файлын системүүдийн шөнө болгоны нэмэгдсэн (эсвэл ялгаатай) нөөцлөлтүүдийг агуулдаг. Эвдэрсэн файлын системийг сэргээх нь хамгийн сүүлийн бүрэн нөөцлөлт болон нэмэгдсэн нөөцлөлтүүдийг шаарддаг.

Тохиргооны файл нь нөөцлөлтүүд болон Amanda-ийн үүсгэдэг сүлжээний урсгалыг нарийн хянах боломжийг хангадаг. Amanda нь өгөгдлийг соронзон хальс уруу бичихийн тулд дээр дурдсан нөөцлөлтийн програмуудыг ашиглана. Amanda нь порт эсвэл багц хэлбэрээр байдаг бөгөөд анхдагчаар суулгагдаагүй байдаг.

### 19.12.6. Юу ч хийхгүй байх

"Юу ч хийхгүй байх" нь компьютерийн програм биш боловч энэ нь хамгийн их ашиглагддаг нөөцлөлтийн стратеги юм. Ямар ч эхний зардал байхгүй. Мөрдөх нөөцлөлтийн төлөвлөгөө байхгүй. Зүгээр л үгүй гэх хэрэгтэй. Хэрэв таны өгөгдөлд ямар нэг асуудал тохиолдвол шүд зуугаад л тэвчих хэрэгтэй!

Хэрэв таны цаг болон өгөгдөл нь тийм ч чухал биш эсвэл юу ч биш бол "Do nothing буюу юу ч хийхгүй байх" нь таны компьютерийн хувьд хамгийн тохиромжтой нөөцлөлтийн програм болох юм. Гэхдээ UNIX® нь ашигтай хэрэгсэл юм, та зургаан сарын дотор үнэтэй файлуудын цуглуулгатай болж үүнийг мэдрэх болно.

"Юу ч хийхгүй байх" нь /usr/obj болон таны компьютераар яг үүсгэгдэж болох бусад сангийн моднуудын хувьд зөв нөөцлөх арга болдог. Үүний жишээ нь энэ гарын авлагын HTML эсвэл PostScript®-с тогтох хувилбарын файлууд байж болох юм. Эдгээр баримтын хэлбэршүүлэлтүүд нь XML оролтын файлуудаас үүсгэгддэг. HTML эсвэл PostScript® файлуудын нөөцлөлтийг үүсгэх нь шаардлагагүй юм. XML файлууд нь байнга нөөцлөгддөг.

## 19.12.7. Аль нөөцлөлтийн програм хамгийн шилдэг нь вэ?

dump(8) *Ye.* Элизабет Д. Звики энд дурдагдсан бүх нөөцлөлтийн програмуудыг зовоосон тест хийсэн. Өөрийн бүх өгөгдөл болон UNIX® файлын системүүдийн онцлогуудыг хадгалах цэвэр сонголт бол `dump` болсон байна. Элизабет маш их төрөл бүрийн ховор нөхцөлүүдийг (зарим нь тийм ч ховор биш) агуулсан файлын системүүдийг үүсгэж тэдгээр файлын системүүдийг нөөцлөн дараа нь сэргээж програм бүрийг тест хийсэн байна. Онцгой зүйлүүдэд цоорхойтой файлууд, хоосон блоктой болон цоорхойтой файлууд, сонин тэмдэгт бүхий нэрээс тогтсон файлууд, бичигдэх боломжгүй уншигдах боломжгүй файлууд, төхөөрөмжүүд, нөөцлөлтийн үед хэмжээгээ өөрчлөх файлууд, нөөцлөлтийн үед үүсгэгдэж/устгагдсан файлууд зэргийг дурдаж болох юм. Тэрээр үр дүнгээ 1991 оны 10 сард LISA дээр үзүүлсэн байна. Нөөцлөлт болон Архивын програмуудыг зовоох тест холбоосыг үзнэ үү.

## 19.12.8. Яаралтай сэргээх процедур

### 19.12.8.1. Гамшгийн өмнө

Учирч болзошгүй гамшигт бэлтгэхийн тулд таны хийх ёстой дөрвөн үе шат байдаг.

Эхлээд өөрийн диск бүрийн bsdlabel (өөрөөр хэлбэл `bsdlabel da0 | lpr`), өөрийн файлын системийн хүснэгт (/etc/fstab) болон бүх ачаалалтын мэдэгдлүүдийг тус бүр хоёр хувь хэвлэх хэрэгтэй.

Дараа нь "livefs" CDROM шарах хэрэгтэй. Энэ CDROM нь FreeBSD "livefs" аврах горим руу ачаалж орох боломжийг агуулсан байдаг. Ингэснээр хэрэглэгч dump(8),restore(8), fdisk(8), bsdlabel(8), newfs(8), mount(8) гэх мэт олон үйлдлийг хийх боломжтой болох юм. FreeBSD/i386 11.2-RELEASE-д зориулсан Livefs CD дүрс ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/ISO-IMAGES/11.2/FreeBSD-11.2-RELEASE-i386-livefs.iso хаяг дээр бий.

Livefs CD дүрснүүд FreeBSD 12.0-RELEASE болон түүнээс хойшхи хувилбаруудын хувьд байдаггүй. CDROM суулгалтын дүрснээс гадна флэш суулгалтын дүрснүүд ашиглан системийг сэргээж болно. FreeBSD/i386 12.0-RELEASE-д зориулсан "memstick" дүрс ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/i386/ISO-IMAGES/12.0/FreeBSD-12.0-RELEASE-i386-memstick.img хаяг дээр бий.

Гуравдугаарт нөөц соронзон хальснуудыг байнга үүсгэж байх хэрэгтэй. Таны сүүлийн нөөцлөлтөөс хойш хийгдсэн өөрчлөлтүүд буцааж авагдах боломжгүйгээр алга болж болох юм. Нөөц соронзон хальснууд уруу хийж болзошгүй бичилтийг хамгаалах хэрэгтэй.

Дөрөвдүгээрт хоёрдугаар шатанд хийсэн "livefs" CDROM болон нөөцийн соронзон хальснуудыг шалгах ёстой. Процедурынхаа талаар тэмдэглэгээ хийх хэрэгтэй. Эдгээр тэмдэглэгээнүүдээ CDROM, хэвлэсэн зүйл болон нөөц соронзон хальснуудтайгаа цуг хадгалах хэрэгтэй. Сэргээж байх үед таны анхаарал их сарнисан байж болох бөгөөд тэдгээр тэмдэглэгээнүүд чинь таныг өөрийн нөөц соронзон хальснуудаа устгахаас сэргийлэх болно (Яаж? `tar xvf /dev/sa0` тушаалын оронд та санамсаргүйгээр `tar cvf /dev/sa0` гэж бичээд өөрийн нөөц соронзон хальсыг дарж бичиж болох юм).

Аюулгүй байдлын нэмэгдэл арга хэмжээ болгож тухайг бүрт нь ачаалагдах "livefs" CDROM болон нөөц соронзон хальс хоёрыг бэлдэж байх хэрэгтэй. Тус бүрийг нь тусад нь алсад өөр газар хадгалах хэрэгтэй. Тэр өөр газар нь яг тэр оффисийн байрны подвал байх ЁСГҮЙ. Дэлхийн Худалдааны Төвийн хэд хэдэн фирм үүнийг хэцүү замаар мэдэрсэн юм. Тэр өөр газар нь таны компьютерууд болон дискний хөтчүүдээс тусгаарлагдсан физикийн хувьд нэлээд зайтай газар байх ёстой.

#### 19.12.8.2. Гамшгийн дараа

Түлхүүр асуулт бол: таны тоног төхөөрөмж амьд гарч чадсан уу? Байнгын нөөцлөлт хийж байсан болохоор програм хангамжийн талаар та санаа зовох хэрэггүй юм.

Хэрэв тоног төхөөрөмж эвдэрсэн бол компьютерийг ашиглахаасаа өмнө эвдэрсэн хэсгүүдийг солих шаардлагатай.

Хэрэв таны тоног төхөөрөмж зүгээр байгаа бол CDROM хөтөч рүү "livefs" CDROM-оо хийгээд компьютераа ачаалах хэрэгтэй. Эхний суулгах цэс дэлгэц дээр гарах болно. Зөв улсаа сонгоод Fixit — Repair mode with CDROM/DVD/floppy or start a shell. сонголтыг сонгоод CDROM/DVD — Use the live filesystem CDROM/DVD цэсийг сонгоно. Танд хэрэгтэй `restore` болон бусад програмууд /mnt2/rescue санд байрлана.

Файлын систем бүрийг тус тусад нь сэргээнэ.

Өөрийн эхний дискний root хуваалтыг `mount` (өөрөөр хэлбэл `mount /dev/da0a /mnt`) хийхийг оролдоорой. Хэрэв bsdlabel эвдэрсэн бол `bsdlabel` тушаалыг ашиглан дискийг дахин хуваж таны хэвлэж хадгалсан шошготой адил тааруулж хаяглах хэрэгтэй. `newfs` тушаал ашиглан файлын системүүдийг дахин үүсгэнэ. Дискний root хуваалтыг унших-бичихээр дахин холбох хэрэгтэй (`mount -u -o rw /mnt`). Энэ файлын системийн хувьд өгөгдлийг сэргээхийн тулд өөрийн нөөцлөх програм болон соронзон хальснуудыг ашиглана (өөрөөр хэлбэл `restore vrf /dev/sa0`). Файлын системийг салгана (өөрөөр хэлбэл `umount /mnt`). Эвдэрсэн файлын систем бүрийн хувьд давтана.

Таны систем ажилласны дараа өөрийн өгөгдлийг шинэ соронзон хальс уруу нөөцлөх хэрэгтэй. Сүйрэл эсвэл өгөгдлийн алдагдалд хүргэсэн тэр явдал дахин болж болох юм. Одоо зарцуулсан нэг цаг таныг зовлонд учрахаас хожим аварч болох юм.

## 19.13. Сүлжээ, санах ой болон файл дээр тулгуурласан файлын системүүд

Өөрийн компьютер уруу физикээр хийж болдог уян диск, CD-үүд, хатуу хөтчүүд гэх зэрэг дискнүүдээс гадна бусад хэлбэрийн дискнүүд буюу *виртуал дискнүүд*ийг FreeBSD ойлгодог.

Эдгээрт Network File System буюу сүлжээний файлын систем болон Coda зэрэг сүлжээний файлын системүүд, санах ой дээр тулгуурласан файлын системүүд болон файл дээр тулгуурласан файлын системүүд орно.

FreeBSD-ийн ямар хувилбарыг ажиллуулж байгаагаас хамаарч та файл дээр болон санах ой дээр тулгуурласан файлын системүүдийг үүсгэж ашиглахдаа өөр өөр хэрэгслүүдийг хэрэглэх шаардлагатай болно.

Хэрэглэгчийн хувьд мэдэгдэлгүйгээр төхөөрөмжийн цэгүүдийг хуваарилахын тулд devfs(5)-г ашиглана.

### 19.13.1. Файл дээр тулгуурласан файлын систем

mdconfig(8) хэрэгсэл нь FreeBSD дээр санах ой дээр тулгуурласан дискүүд, md(4)-г тохируулж идэвхжүүлэхэд хэрэглэгддэг. mdconfig(8)-г ашиглахын тулд та md(4) модулийг ачаалах хэрэгтэй, эсвэл өөрийн цөмийн тохиргооны файлдаа дэмжлэгийг нэмэх хэрэгтэй болно:

```
device md
```

mdconfig(8) тушаал нь гурван төрлийн санах ой дээр тулгуурласан виртуал дискийг дэмждэг: эдгээр нь malloc(9)-р хуваарилагдсан санах ойн дискүүд, файл эсвэл swap зай ашигласан санах ойн дискүүд юм. Нэг боломжтой хэрэглээ нь файлд хадгалагдсан уян эсвэл CD дүрсийг холбох явдал юм.

Байгаа файлын системийн дүрсийг холбохдоо:

*Жишээ 26. `mdconfig` ашиглан байгаа файлын системийн дүрсийг холбох нь*

```
# mdconfig -a -t vnode -f diskimage -u 0
# mount /dev/md0 /mnt
```

mdconfig(8) ашиглан шинэ файлын системийн дүрсийг үүсгэхдээ:

*Жишээ 27. `mdconfig` тушаал ашиглан шинэ файл дээр тулгуурласан диск үүсгэх нь*

```
# dd if=/dev/zero of=newimage bs=1k count=5k
5120+0 records in
5120+0 records out
# mdconfig -a -t vnode -f newimage -u 0
# bsdlabel -w md0 auto
# newfs md0a
/dev/md0a: 5.0MB (10224 sectors) block size 16384, fragment size 2048
        using 4 cylinder groups of 1.25MB, 80 blks, 192 inodes.
super-block backups (for fsck -b #) at:
 160, 2720, 5280, 7840
# mount /dev/md0a /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md0a       4710    4  4330     0%    /mnt
```

Хэрэв та `-u` тохируулгаар нэгжийн дугаарыг зааж өгөхгүй юм бол mdconfig(8) нь

хэрэглэгдээгүй төхөөрөмжийг сонгохын тулд md(4) автомат хуваарилалтыг ашиглах болно. Хуваарилагдсан нэгжийн нэр нь stdout уруу md4 мэтээр гарах болно. mdconfig(8)-ийн тухай дэлгэрэнгүй мэдээллийг гарын авлагын хуудаснаас лавлана уу.

mdconfig(8) хэрэгсэл нь их ашигтай, гэхдээ энэ нь файл дээр тулгуурласан файлын систем үүсгэхийн тулд олон тушаалын мөрийг асуудаг. FreeBSD нь бас mdmfs(8) гэж нэрлэгддэг хэрэгсэлтэй ирдэг, энэ програм нь md(4)-г mdconfig(8) тушаал ашиглан тохируулж newfs(8) ашиглан UFS файлын систем түүн дээр тавьж mount(8) ашиглан түүнийг холбодог. Жишээ нь хэрэв та дээрхтэй ижил файлын системийн дүрсийг үүсгэж холбохыг хүсвэл ердөө л дараах тушаалыг бичээрэй:

*Жишээ 28. `mdmfs` тушаал ашиглан файл дээр тулгуурласан дискийг тохируулж холбох нь*

```
# dd if=/dev/zero of=newimage bs=1k count=5k
5120+0 records in
5120+0 records out
# mdmfs -F newimage -s 5m md0 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md0        4718    4  4338     0%    /mnt
```

Хэрэв та `md` тохируулгыг нэгжийн дугааргүй ашиглавал mdmfs(8) нь ашиглагдаагүй төхөөрөмжийг автоматаар сонгохдоо md(4) авто-нэгж боломжийг ашиглах болно. mdmfs(8) -ийн тухай дэлгэрэнгүй мэдээллийг гарын авлагын хуудаснаас лавлана уу.

### 19.13.2. Санах ой дээр тулгуурласан файлын систем

Санах ой дээр тулгуурласан файлын системийн хувьд "swap backing буюу swap дээрх тулгуурлалтыг" ихэнхдээ ашиглах хэрэгтэй. swap дээрх тулгуурлалтыг ашиглах нь санах ойн диск нь анхдагчаар диск уруу swap хийгдэнэ гэсэн үг биш бөгөөд харин ердөө л санах ойн диск нь санах ойн цөөрмөөс хуваарилагдаж хэрэв хэрэгцээтэй бол диск уруу swap хийгдэж болно гэсэн үг юм. Мөн malloc(9) дээр тулгуурласан санах ойн дискийг үүсгэх боломжтой байдаг. Гэхдээ malloc дээр тулгуурласан санах ойн дискнүүдийг, ялангуяа том хэмжээтэйг нь ашиглах нь цөм санах ойгүй болох тохиолдолд системийг сүйрэлд хүргэж болох юм.

*Жишээ 29. `mdconfig` тушаал ашиглан шинэ санах ой дээр тулгуурласан диск үүсгэх нь*

```
# mdconfig -a -t swap -s 5m -u 1
# newfs -U md1
/dev/md1: 5.0MB (10240 sectors) block size 16384, fragment size 2048
        using 4 cylinder groups of 1.27MB, 81 blks, 192 inodes.
        with soft updates
super-block backups (for fsck -b #) at:
 160, 2752, 5344, 7936
# mount /dev/md1 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
```

```
/dev/md1        4718    4  4338     0%    /mnt
```

*Жишээ 30. `mdmfs` тушаал ашиглан шинэ санах ой дээр тулгуурласан диск үүсгэх нь*

```
# mdmfs -s 5m md2 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md2        4846    2  4458     0%    /mnt
```

### 19.13.3. Системээс санах ойн дискийг салгах нь

Санах ой дээр тулгуурласан эсвэл файл дээр тулгуурласан файлын систем ашиглагдахгүй байгаа бол та бүх эх үүсвэрүүдийг системд суллаж өгөх шаардлагатай. Эхний хийх ёстой зүйл нь файлын системийг салгаад дараа нь mdconfig(8) ашиглан дискийг системээс салгаж эх үүсвэрүүдийг суллана.

Жишээ нь /dev/md4-г салгаж түүний хэрэглэсэн бүх эх үүсвэрүүдийг чөлөөлөхийн тулд дараах тушаалыг ажиллуулна:

```
# mdconfig -d -u 4
```

Тохиргоо хийгдсэн md(4) төхөөрөмжүүдийн тухай мэдээллийг `mdconfig -l` тушаал ашиглан жагсааж болно.

## 19.14. Файлын системийн хормын хувилбарууд

FreeBSD нь Soft Updates буюу зөөлөн шинэчлэлтүүдийн хамт Файлын системийн хормын хувилбарууд боломжийг санал болгодог.

Хормын хувилбарууд нь заасан файлын системүүдийн дүрсийг үүсгэж тэдгээртэй файлтай ханддаг шиг хандах боломжийг хэрэглэгчид зөвшөөрдөг. Хормын хувилбарын файлууд нь үйлдэл хийгдсэн файлын систем дээр үүсгэгдэх ёстой бөгөөд нэг файлын системийн хувьд 20-с ихгүй хормын хувилбаруудыг хэрэглэгч үүсгэж болно. Идэвхтэй хормын хувилбарууд нь супер блокт бичигддэг бөгөөд ингэснээр тэдгээр нь системийн дахин ачаалалтуудын үед салгах болон дахин холбох явцад тэсвэртэй байдаг. Хормын хувилбар нь хэрэггүй болсон үед энэ нь стандарт rm(1) тушаалаар устгагдаж болно. Хормын хувилбарууд нь ямар ч дарааллаар устгагдаж болох бөгөөд гэхдээ ашиглагдсан бүх зайг дахин олж авахгүй байж болох юм. Учир нь өөр нэг хормын хувилбар чөлөөлөгдсөн блокуудын заримыг шаардаж болзошгүй юм.

Өөрчлөгдөхгүй `snapshot` файл тохируулга нь хормын хувилбарыг эхлээд үүсгэсний дараа mksnap_ffs(8) тушаалаар заагддаг. unlink(1) тушаал нь хормын хувилбарын файлуудын хувьд орхидог бөгөөд учир нь энэ нь тэдгээрийг устгахыг зөвшөөрдөг байна.

Хормын хувилбарууд нь mount(8) тушаалаар үүсгэгддэг. /var-ийн хормын хувилбарыг /var/snapshot/snap файлд байрлуулахын тулд дараах тушаалыг ашиглана:

```
# mount -u -o snapshot /var/snapshot/snap /var
```

Өөрөөр та хормын хувилбар үүсгэхийн тулд mksnap_ffs(8)-г ашиглаж болно:

```
# mksnap_ffs /var /var/snapshot/snap
```

Файлын систем (өөрөөр хэлбэл /var) дээр хормын хувилбарын файлуудыг find(1) тушаал ашиглан олж болно:

```
# find /var -flags snapshot
```

Хормын хувилбар үүсгэгдсэний дараа энэ нь хэд хэдэн хэрэглээтэй байдаг:

- Зарим администраторууд хормын хувилбарын файлыг нөөцийн зорилгоор ашигладаг, учир нь хормын хувилбар нь CD-үүд эсвэл соронзон хальс уруу дамжуулагдаж болдог.
- Файлын системийн бүрэн бүтэн байдлыг шалгадаг fsck(8) нь хормын хувилбар дээр ажиллаж болдог. Файлын систем нь холбогдсон үедээ цэвэр байсан гэж үзвэл та үргэлж цэвэр (өөрчлөгдөөгүй) үр дүнг авах болно. Үүнийг гол нь арын fsck(8) процесс хийдэг.
- dump(8) хэрэгслийг хормын хувилбар дээр ажиллуулдаг. Хормын хувилбарын файлын систем болон цагтай нэг янз хуулбар хийгддэг. dump(8) нь бас хормын хувилбарыг авч нөөц дүрсийг үүсгэж чаддаг бөгөөд дараа нь хормын хувилбарыг нэг тушаалаар `-L` тугийг ашиглан устгаж чаддаг.
- Хормын хувилбарыг файлын системийн хөлдөөгдсөн дүрс маягаар mount(8) хийдэг. /var/snapshot/snap хормын хувилбарыг mount(8) хийхийн тулд доор дурдсаныг ажиллуулна:

  ```
  # mdconfig -a -t vnode -f /var/snapshot/snap -u 4
  # mount -r /dev/md4 /mnt
  ```

Та одоо /mnt-д холбогдсон өөрийн хөлдөөгдсөн /var файлын системийн шатлалаар аялж болно. Хормын хувилбар үүсгэх үе дэх төлвийн адил төлөвт бүгд эхлээд байна. Цорын ганц өөр зүйл нь өмнөх хормын хувилбарууд нь тэг урттай файлууд маягаар гарч ирэх болно. Хормын хувилбарын хэрэглээг хязгаарлах үед үүнийг дараах тушаалыг ажиллуулан салгаж болно:

```
# umount /mnt
# mdconfig -d -u 4
```

`softupdates` буюу зөөлөн шинэчлэлтүүд болон файлын системийн хормын хувилбаруудын талаар дэлгэрэнгүй мэдээллийг техникийн баримтуудыг оролцуулаад та Маршал Кирк

МкКюзикийн вэб хуудас http://www.mckusick.com/ уруу зочлон орж үзэж болно.

# 19.15. Файлын системийн ноогдлууд

Quota буюу ноогдол нь хэрэглэгч эсвэл бүлгийн гишүүдийн хувьд дискний зай ба/эсвэл файлуудын тоог файлын систем тус бүрийн хувьд хуваарилж хязгаарлах боломжийг бүрдүүлэх үйлдлийн системийн нэг нэмэлт боломж юм. Энэ нь дурын нэг хэрэглэгч эсвэл бүлэг хэрэглэгчдийн хуваарилж болох эх үүсвэрүүдийг хязгаарлах шаардлагатай цаг хуваалцах систем дээр ихэвчлэн хэрэглэгддэг. Энэ нь нэг хэрэглэгч эсвэл бүлэг хэрэглэгчид бүх байгаа дискний зайг хэрэглэхээс хамгаалдаг юм.

## 19.15.1. Дискний ноогдлуудыг идэвхтэй болгохын тулд өөрийн системийг тохируулах нь

Дискний ноогдлыг ашиглаж эхлэхээсээ өмнө ноогдлууд нь цөмд тохируулагдсан эсэхийг шалгах шаардлагатай. Дараах мөрийг өөрийн цөмийн тохиргооны файлд нэмж үүнийг хийнэ:

```
options QUOTA
```

Анхны GENERIC цөм нь үүнийг анхнаасаа идэвхжүүлээгүй байдаг болохоор та дискний ноогдлуудыг ашиглахын тулд тусгайлсан цөмийг тохируулж бүтээн суулгах хэрэгтэй болох юм. Цөмийн тохиргооны талаар дэлгэрэнгүй мэдээллийг FreeBSD цөмийг тохируулах нь-с лавлана уу.

Дараа нь та дискний ноогдлуудыг /etc/rc.conf файлд идэвхжүүлэх хэрэгтэй. FreeBSD 7.X болон түүнээс өмнөх дээр дараах мөрийг нэмж үүнийг хийнэ:

```
enable_quotas="YES"
```

FreeBSD 8.0-RELEASE болон түүнээс хойшхи дээр дараах мөрийг нэмнэ:

```
quota_enable="YES"
```

Өөрийн ноогдлын эхлүүлэлтэд сайн хяналт тавихад хэрэгтэй нэмэлт тохиргооны хувьсагчууд байдаг. Ердийн үед ачаалалтын үеэр файлын систем бүрийн ноогдлын бүрэн бүтэн байдал quotacheck(8) программаар шалгагддаг. quotacheck(8) боломж нь ноогдлын мэдээллийн бааз дахь өгөгдөл файлын систем дээрх өгөгдлийг зөв тусгаж байгаа эсэхийг шалгаж байдаг. Энэ нь хугацаа их шаардсан процесс бөгөөд таны систем ачаалах хугацаанд мэдэгдэхүйц нөлөөлдөг. Хэрэв та энэ алхмыг алгасахыг хүсвэл энэ зорилгоор хувьсагчийг /etc/rc.conf файлд тохируулах хэрэгтэй:

```
check_quotas="NO"
```

Төгсгөлд нь та файлын систем бүрийн хувьд дискний ноогдлуудыг идэвхжүүлэхийн тулд /etc/fstab файлыг засварлах хэрэгтэй. Энд та хэрэглэгч эсвэл бүлгийн ноогдлууд эсвэп хоёуланг нь өөрийн бүх файлын системийн хувьд идэвхжүүлэх боломжтой юм.

Файлын систем дээр хэрэглэгч бүрийн хувьд ноогдлуудыг идэвхжүүлэхийн тулд өөрийн ноогдлууд идэвхжүүлэхийг хүсэж байгаа файлын системийн хувьд /etc/fstab файлын харгалзах оруулгын options буюу тохируулгыг заах талбарт `userquota` тохируулгыг нэмж өгөх хэрэгтэй. Жишээ нь:

```
/dev/da1s2g   /home    ufs rw,userquota 1 2
```

Үүнтэй төстэйгээр бүлгийн ноогдлуудыг идэвхжүүлэхийн тулд `userquota` тохируулгын оронд `groupquota` тохируулгыг ашиглах хэрэгтэй. Хэрэглэгч болон бүлгийн ноогдлуудыг хоёуланг нь идэвхжүүлэхийн тулд оруулгыг ингэж өөрчилнө:

```
/dev/da1s2g    /home    ufs rw,userquota,groupquota 1 2
```

Анхдагчаар ноогдлын файлууд файлын системийн root санд хэрэглэгчийн ноогдлын хувьд quota.user, бүлгийн ноогдлын хувьд quota.group нэрээр хадгалагддаг. Дэлгэрэнгүй мэдээллийг fstab(5)-с үзнэ үү. fstab(5) гарын авлагын хуудас ноогдлын файлуудын хувьд өөр байрлалыг зааж өгч болохыг хэлдэг боловч ноогдлын төрөл бүрийн хэрэгслүүд үүнтэй зөв ажиллаж чаддаггүй болохоор ингэхийг зөвлөдөггүй юм.

Энэ үед хүрээд та өөрийн системийг шинэ цөмтэй дахин ачаалах хэрэгтэй. /etc/rc нь таны /etc/fstab файлд идэвхжүүлсэн бүх ноогдлуудын хувьд эхний ноогдлын файлуудыг үүсгэдэг тохирох тушаалуудыг автоматаар ажиллуулах бөгөөд ингэснээр ямар ч тэг урттай ноогдлын файлуудыг гараар үүсгэх шаардлагагүй юм.

Ердийн үйлдлүүдийн үед quotacheck(8), quotaon(8), эсвэл quotaoff(8) тушаалуудыг та заавал ажиллуулах ёсгүй юм. Гэхдээ тэдгээрийн хийдэг үйлдлүүдтэй танилцахын тулд тэдгээрийн гарын авлагын хуудаснуудыг уншихыг хүсэж болох юм.

### 19.15.2. Ноогдлын хязгаарлалтуудыг тохируулах нь

Та ноогдлыг идэвхжүүлж өөрийн системийг тохируулсны дараа тэдгээр нь яг жинхэнэдээ идэвхтэй болсон эсэхийг шалгаарай. Үүнийг хийх хялбар арга нь дараах тушаалыг ажиллуулах явдал юм:

```
# quota -v
```

Та ноогдлууд идэвхжүүлэгдсэн файлын систем бүрийн хувьд дискний хэрэглээ болон тухайн үеийн ноогдлын хязгаарлалтуудын ерөнхий нэг мөр үр дүнг харах ёстой.

Та одоо edquota(8) тушаал ашиглан ноогдлын хязгаарлалтуудыг зааж өгч эхлэхэд бэлэн боллоо.

Хэрэглэгч эсвэл бүлгийн хуваарилж болох дискний зай болон тэд хичнээн файл үүсгэж болох хэмжээний хязгаарлалтуудыг хүчлэн зааж өгч болох хэд хэдэн сонголт танд байдаг. Та хуваарилалтыг дискний зай (блок ноогдлууд) эсвэл файлын тоо (inode ноогдлууд) эсвэл хоёулангийн хослолоор хязгаарлан өгч болно. Эдгээр хязгаарлалт бүр дараа нь хоёр зэрэглэлт хуваагддаг: энэ нь хатуу ба зөөлөн хязгаарлалтууд юм.

Хатуу хязгаарлалтаас илүү гарч болохгүй. Хэрэглэгч өөрийн хатуу хязгаарлалтад хүрэхэд асуудалтай байгаа файлын систем дээр хэрэглэгч ямар ч хуваарилалт хийж чадахгүй болно. Жишээ нь хэрэв хэрэглэгч файлын систем дээр 500 кбайт хатуу хязгаарлалттай бөгөөд 490 кбайтыг одоогоор ашиглаж байгаа бол хэрэглэгч зөвхөн 10 кбайтыг нэмж хуваарилж чадна. 11 кбайт нэмж хуваарилахыг оролдох нь амжилтгүй болох болно.

Нөгөө талаас зөөлөн хязгаарлалтууд нь тодорхой хугацааны туршид илүү гарч болно. Энэ хугацааг энэрэнгүй хугацаа гэдэг бөгөөд анхдагчаар нэг долоо хоног байдаг. Хэрэв хэрэглэгч өөрийн зөөлөн хязгаарлалтаас илүү гарсан хэвээр энэрэнгүй хугацааг өнгөрөх юм бол зөөлөн хязгаарлалт нь хатуу хязгаарлалт болж дахин хуваарилалтыг зөвшөөрдөггүй. Хэрэглэгч зөөлөн хязгаарлалтаас доош орвол энэрэнгүй хугацаа дахин тавигдах болно.

Дараах нь edquota(8) тушаалыг ажиллуулах үед таны харж болох гаралтын жишээ юм. edquota(8) тушаалыг ажиллуулахад `EDITOR` орчны хувьсагчид заагдсан засварлагч уруу эсвэл `EDITOR` хувьсагч заагдаагүй бол vi засварлагч уруу таныг оруулах бөгөөд энэ нь ноогдлын хязгаарлалтуудыг засварлах боломжийг танд олгоно.

```
# edquota -u test
```

```
Quotas for user test:
/usr: kbytes in use: 65, limits (soft = 50, hard = 75)
        inodes in use: 7, limits (soft = 50, hard = 60)
/usr/var: kbytes in use: 0, limits (soft = 50, hard = 75)
        inodes in use: 0, limits (soft = 50, hard = 60)
```

Та ноогдлууд идэвхжүүлэгдсэн файлын систем бүрийн хувьд хоёр мөрийг ердийн үед харах болно. Нэг мөр нь блокийн хязгаарлалтад, нөгөө мөр нь inode хязгаарлалтад зориулагддаг. Ноогдлын хязгаарлалтуудыг өөрчлөхийн тулд та ердөө л утгыг шинэчлэхийг хүсэж байгаа утгаараа солих хэрэгтэй. Жишээ нь энэ хэрэглэгчийн блок хязгаарлалтыг зөөлөн хязгаарлалт 50, хатуу хязгаарлалт 75 байсныг зөөлөн хязгаарлалт 500 болон хатуу хязгаарлалт 600 болгон өсгөхийн тулд дараах:

```
/usr: kbytes in use: 65, limits (soft = 50, hard = 75)
```

мөрийг доорх шиг болгож өөрчлөх хэрэгтэй:

```
/usr: kbytes in use: 65, limits (soft = 500, hard = 600)
```

Таныг засварлагчаас гарсны дараа шинэ ноогдлын хязгаарлалтууд байрандаа орох болно.

Заримдаа ноогдлын хязгаарлалтуудыг зарим UID-ийн хүрээний хувьд зааж өгөх шаардлага гардаг. Үүнийг edquota(8) тушаалд `-p` тохируулгыг ашиглан хийдэг. Эхлээд хэрэглэгчид хүссэн ноогдлын хязгаарлалтаа зааж өгөөд дараа нь `edquota -p protouser startuid-enduid` тушаалыг ажиллуулна. Жишээ нь хэрэв хэрэглэгч `test` хүссэн ноогдлын хязгаарлалттай байгаа бол энэ ноогдлын хязгаарлалтыг 10,000-с 19,999 хүртэлх UID-уудад хувилж хамааруулахын тулд дараах тушаалыг ажиллуулна:

```
# edquota -p test 10000-19999
```

Дэлгэрэнгүй мэдээллийг edquota(8) гарын авлагын хуудаснаас үзнэ үү.

### 19.15.3. Ноогдлын хязгаарлалтууд болон дискний хэрэглээг шалгах нь

Ноогдлын хязгаарлалтууд болон дискний хэрэглээг шалгахын тулд та quota(1) эсвэл repquota(8) тушаалын нэгийг ашиглаж болно. quota(1) тушаалыг хэрэглэгч эсвэл бүлгийн ноогдлууд болон дискний зайг шалгахад хэрэглэж болно. Хэрэглэгч нь зөвхөн өөрийн ноогдол болон өөрийн гишүүн бүлгийн ноогдлыг шалгаж чадна. Зөвхөн супер хэрэглэгч бүх хэрэглэгч болон бүлгийн ноогдлуудыг харж болно. repquota(8) тушаалыг ноогдлууд идэвхжүүлэгдсэн файлын системүүдийн хувьд бүх ноогдлууд болон дискний хэрэглээний ерөнхий дүнг авахад хэрэглэж болно.

Дараах нь хоёр файлын систем дээр ноогдлын хязгаарлалттай хэрэглэгчийн хувьд `quota -v` тушаалын жишээ гаралт юм.

```
Disk quotas for user test (uid 1002):
     Filesystem  usage    quota   limit   grace   files   quota   limit   grace
           /usr      65*     50      75   5days       7      50      60
       /usr/var       0      50      75               0      50      60
```

Дээрх жишээний /usr файлын систем дээр хэрэглэгч 50 кбайтын зөөлөн хязгаарлалтыг 15 кбайтаар илүүдүүлсэн байгаа бөгөөд 5 хоногийн энэрэнгүй хугацаа үлдсэн байна. Од `*` тэмдэгт нь хэрэглэгч өөрийн ноогдлын хязгаарлалтаас илүү гарсныг харуулж байна.

Ердийн үед хэрэглэгч ямар ч дискний зай ашиглаагүй байгаа тийм файлын системүүд, файлын системийн хувьд ноогдлын хязгаарлалт байсан ч гэсэн quota(1) тушаалын гаралтад харагддаггүй байна. `-v` тохируулга нь тэдгээр файлын системүүдийг харуулах бөгөөд дээрх жишээн дээр /usr/var файлын системийг харуулж байна.

### 19.15.4. NFS дээгүүрх ноогдлууд

Ноогдлууд нь NFS сервер дээр ноогдлын дэд системээр хянагддаг. rpc.rquotad(8) демон нь NFS клиентүүд дээр өөрсдийн ноогдлын статистикуудыг хэрэглэгчдэд үзэх боломжийг олгож ноогдлын мэдээллийг тэдгээр машинууд дээр quota(1) тушаалд бэлэн болгож өгдөг.

`rpc.rquotad`-г /etc/inetd.conf файлд ингэж идэвхжүүлнэ:

```
rquotad/1      dgram rpc/udp wait root /usr/libexec/rpc.rquotad rpc.rquotad
```

Одоо `inetd`-г дахин ачаал:

```
# /etc/rc.d/inetd restart
```

# 19.16. Дискний хуваалтуудыг шифрлэх нь

FreeBSD нь өгөгдөлд зөвшөөрөгдөөгүй хандах хандалтын эсрэг маш сайн шууд хамгаалалтуудыг санал болгодог. Файлын зөвшөөрлүүд болон Mandatory Access Control (MAC) буюу Албадмал Хандалт Хяналт (Mandatory Access Control буюу Албадмал Хандалтын хяналт-г үзнэ үү) нь компьютер ассан, үйлдлийн систем идэвхтэй байх үед зөвшөөрөлгүй гуравдагч нөхдүүдийг өгөгдөлд хандахаас хамгаалахад тусалдаг. Гэхдээ хэрэв халдагч компьютерт физик хандалт олж аван компьютерийн хатуу хөтчийг өөр систем уруу зөөн эмзэг өгөгдлийг хуулж шинжилгээ хийж чадах бол үйлдлийн системийн мөрдүүлдэг зөвшөөрлүүд нь хамааралгүй болох юм.

Халдагч хэрхэн хатуу диск эсвэл унтраасан компьютерийг эзэмшсэнээс үл хамаарч FreeBSD дээрх GEOM Based Disk Encryption (gbde) буюу GEOM дээр тулгуурласан дискний шифрлэлт болон `geli` криптограф дэд системүүд нь компьютерийн файлын системүүд дээрх өгөгдлийг бүр чухал эх үүсвэрүүд бүхий, зорилгодоо хүрэхээр ихээхэн чармайсан халдагчдаас хамгаалж чадах юм. Зөвхөн тухайн файл бүрийг шифрлэдэг ярвигтай шифрлэлтийн аргуудаас ялгаатай нь `gbde` болон `geli` нь бүх файлын системүүдийг харагдаж мэдэгдэхгүйгээр шифрлэдэг. Хатуу дискний хавтгайд цэвэр текст хэзээ ч бичигддэггүй.

## 19.16.1. gbde-ээр диск шифрлэх

1. `root` болно

   gbde-г тохируулахад супер хэрэглэгчийн эрхүүдийг шаарддаг.

   ```
   % su -
   Password:
   ```

2. Цөмийн тохиргооны файлдаа gbde(4) дэмжлэгийг нэмнэ

   Дараах мөрийг цөмийн тохиргооны файлдаа нэмнэ:

   ```
   options GEOM_BDE
   ```

   Цөмөө FreeBSD цөмийг тохируулах нь-д тайлбарласны дагуу дахин бүтээнэ.

   Шинэ цөм уруугаа дахин ачаална.

3. Цөмийг дахин бүтээхээс өөр нэг арга нь `kldload` ашиглаж gbde(4)-г ачаалах явдал юм:

```
# kldload geom_bde
```

#### 19.16.1.1. Шифрлэгдсэн хатуу хөтчүүдийг бэлдэх нь

Дараах жишээ нь таныг өөрийн систем уруу шинэ хатуу хөтөч нэмж ганц шифрлэгдсэн хуваалтыг агуулна гэж тооцдог. Энэ хуваалт нь /private гэж холбогдох болно. gbde нь /home болон /var/mail-г бас шифрлэхэд ашиглагддаг боловч энэ нь энэ танилцуулгын хүрээнээс хальж илүү төвөгтэй зааврууд шаарддаг.

1. Шинэ хатуу хөтчийг нэмнэ

   Шинэ хөтчийг систем уруу Диск нэмэх-д тайлбарласны дагуу нэмнэ. Энэ жишээн дээр шинэ хатуу хөтчийн хуваалт /dev/ad4s1c гэж нэмэгдсэн байгаа. /dev/ad0s1* төхөөрөмжүүд нь жишээ систем дээр байгаа стандарт FreeBSD хуваалтуудыг харуулж байна.

   ```
   # ls /dev/ad*
   /dev/ad0        /dev/ad0s1b     /dev/ad0s1e     /dev/ad4s1
   /dev/ad0s1      /dev/ad0s1c     /dev/ad0s1f     /dev/ad4s1c
   /dev/ad0s1a     /dev/ad0s1d     /dev/ad4
   ```

2. gbde цоож файлуудыг агуулах санг үүсгэнэ

   ```
   # mkdir /etc/gbde
   ```

   gbde цоож файл нь шифрлэгдсэн хуваалтуудад хандахад gbde-д шаардагдах мэдээллийг агуулдаг. Цоож файлд хандах боломжгүй бол gbde нь шифрлэгдсэн хуваалтын агуулсан өгөгдлийг хөндлөнгийн их хүч шаардсан оролцоогүйгээр буцааж тайлж чадахгүй юм. Үүнийг програм хангамж дэмждэггүй. Шифрлэгдсэн хуваалт бүр тусдаа цоож файл ашигладаг.

3. gbde хуваалтыг эхлүүлнэ

   gbde хуваалт ашиглагдаж эхлэхээсээ өмнө эхэлж тохируулагдсан байх шаардлагатай. Энэ эхний тохируулга нь зөвхөн нэг удаа хийгдэх ёстой:

   ```
   # gbde init /dev/ad4s1c -i -L /etc/gbde/ad4s1c.lock
   ```

   gbde(8) нь загварт төрөл бүрийн тохиргооны сонголтуудыг зааж өгөх боломжийг танд өгч таны засварлагчийг онгойлгох болно. UFS1 эсвэл UFS2-той ашиглахын тулд sector_size-г 2048 гэж заана:

   ```
   # $FreeBSD: src/sbin/gbde/template.txt,v 1.1.36.1 2009/08/03 08:13:06 kensmith
   ```

```
Exp $
#
# Sector size is the smallest unit of data which can be read or written.
# Making it too small decreases performance and decreases available space.
# Making it too large may prevent filesystems from working.  512 is the
# minimum and always safe.  For UFS, use the fragment size
#
sector_size     =       2048
[...]
```

gbde(8) нь өгөгдлийг аюулгүй болгоход хэрэглэгдэх ёстой нэвтрэх үгийг хоёр удаа бичихийг танаас хүснэ. Нэвтрэх үг нь хоёуланд нь адил байх ёстой. gbde-ийн таны өгөгдлийг хамгаалах чадвар таны сонгосон нэвтрэх үгийн чанараас бүхэлдээ хамаарна.

`gbde init` тушаал нь таны gbde хуваалтад зориулж цоож файлыг үүсгэх бөгөөд энэ жишээн дээр /etc/gbde/ad4s1c.lock гэж хадгалагдсан байна. gbde цоож файлууд нь /etc/rc.d/gbde эхлүүлэх скриптээр зөв танигдахын тулд ".lock" гэж төгсөх ёстой.

gbde цоож файлууд нь шифрлэгдсэн хуваалтуудын агуулгатай цуг нөөцлөгдөх *ёстой*. Зөвхөн цоож файлыг устгах нь шийдсэн халдагч gbde хуваалтыг буцааж тайлахаас хамгаалж чаддаггүй бөгөөд цоож файлгүйгээр хууль ёсны эзэмшигч нь шифрлэгдсэн хуваалт дээрх өгөгдөлд gbde(8) болон түүнийг зохиогчийн ерөөсөө дэмждэггүй, их хүч шаардсан ажиллагаагүйгээр хандаж чадахгүй болох юм.

4. Шифрлэгдсэн хуваалтыг цөмд залгана

```
# gbde attach /dev/ad4s1c -l /etc/gbde/ad4s1c.lock
```

Шифрлэгдсэн хуваалтыг эхэлж тохируулж байх үед таны сонгосон нэвтрэх үгийг оруулахыг танаас асуух болно. Шинэ шифрлэгдсэн төхөөрөмж /dev дотор /dev/device_name.bde гэж гарч ирнэ:

```
# ls /dev/ad*
/dev/ad0        /dev/ad0s1b     /dev/ad0s1e     /dev/ad4s1
/dev/ad0s1      /dev/ad0s1c     /dev/ad0s1f     /dev/ad4s1c
/dev/ad0s1a     /dev/ad0s1d     /dev/ad4        /dev/ad4s1c.bde
```

5. Шифрлэгдсэн төхөөрөмж дээр файлын систем үүсгэнэ

Шифрлэгдсэн төхөөрөмж цөмд залгагдсаны дараа та энэ төхөөрөмж дээр файлын систем үүсгэж болно. Шифрлэгдсэн төхөөрөмж дээр файлын систем үүсгэхийн тулд newfs(8)-г ашиглана. Хуучин UFS1 файлын системийг эхлүүлснээс шинэ UFS2 файлын системийг эхлүүлэх нь хамаагүй хурдан учраас newfs(8)-г `-O2` тохируулгатай хэрэглэхийг зөвлөдөг.

```
# newfs -U -O2 /dev/ad4s1c.bde
```

newfs(8) тушаал нь төхөөрөмжийн нэрэндээ *.bde гэж өргөтгөлөөр танигдах залгагдсан gbde хуваалт дээр хийгдэх ёстой.

6. Шифрлэгдсэн хуваалтыг холбоно

   Шифрлэгдсэн файлын системд зориулж холбох цэгийг үүсгэнэ.

   ```
   # mkdir /private
   ```

   Шифрлэгдсэн файлын системийг холбоно.

   ```
   # mount /dev/ad4s1c.bde /private
   ```

7. Шифрлэгдсэн файлын систем байгаа эсэхийг шалгана

   Шифрлэгдсэн файлын систем одоо df(1)-д харагдаж ашиглахад бэлэн болох ёстой.

   ```
   % df -H
   Filesystem        Size   Used  Avail Capacity  Mounted on
   /dev/ad0s1a      1037M    72M   883M     8%    /
   /devfs            1.0K   1.0K     0B   100%    /dev
   /dev/ad0s1f       8.1G    55K   7.5G     0%    /home
   /dev/ad0s1e      1037M   1.1M   953M     0%    /tmp
   /dev/ad0s1d       6.1G   1.9G   3.7G    35%    /usr
   /dev/ad4s1c.bde   150G   4.1K   138G     0%    /private
   ```

#### 19.16.1.2. Байгаа шифрлэгдсэн файлын системүүдийг холбох нь

Ачаалалт бүрийн дараа ямар ч шифрлэгдсэн файлын системүүдийг ашиглаж эхлэхээсээ өмнө цөмд дахин залгаж, алдааг нь шалгаж холбож болно. Шаардлагатай тушаалуудыг `root` хэрэглэгч ажиллуулах ёстой гэж үздэг.

1. `gbde` хуваалтыг цөмд залгана

   ```
   # gbde attach /dev/ad4s1c -l /etc/gbde/ad4s1c.lock
   ```

   Шифрлэгдсэн gbde хуваалтыг эхэлж тохируулж байх үед таны сонгосон нэвтрэх үгийг оруулахыг танаас асуух болно.

2. Файлын систем дээр алдааг шалгана

Шифрлэгдсэн файлын системүүд нь автоматаар холбогдохын тулд /etc/fstab файлд одоохондоо жагсаагдах боломжгүй учир файлын системүүд дээр алдаа байгаа эсэхийг шалгаж тэдгээрийг холбохоос өмнө fsck(8)-г гараар ажиллуулах ёстой.

```
# fsck -p -t ffs /dev/ad4s1c.bde
```

3. Шифрлэгдсэн файлын системийг холбоно

```
# mount /dev/ad4s1c.bde /private
```

Шифрлэгдсэн файлын систем одоо ашиглахад бэлэн боллоо.

#### 19.16.1.2.1. Шифрлэгдсэн хуваалтуудыг автоматаар холбох нь

Шифрлэгдсэн хуваалтыг автоматаар залган шалгаж холбох скриптийг үүсгэж болох боловч аюулгүй байдлын шалтгаанаас болоод скрипт нь gbde(8) нууц үгийг агуулсан байх ёсгүй. Харин эдгээр скриптүүдийг гараар ажиллуулж нууц үгийг консол эсвэл ssh(1)-ээр оруулахыг зөвлөдөг.

Өөр нэг арга нь rc.d скрипт байдаг. Энэ скриптийн нэмэлт өгөгдлүүдийг rc.conf(5)-оор дамжуулж болно, жишээ нь:

```
gbde_autoattach_all="YES"
gbde_devices="ad4s1c"
gbde_lockdir="/etc/gbde"
```

Энэ нь gbde нэвтрэх үгийг ачаалах үед оруулахыг шаарддаг. Зөв нэвтрэх үгийг бичсэний дараа gbde шифрлэгдсэн хуваалт автоматаар холбогдох болно. Энэ нь gbde-г зөөврийн компьютер дээр хэрэглэж байгаа үед их ашигтай байдаг.

### 19.16.1.3. gbde-ийн ажиллуулдаг криптограф хамгаалалтууд

gbde(8) нь 128-bit AES-ийг CBC горимд ашиглан сектор ачааг шифрлэдэг. Диск дээрх сектор бүр өөр өөр AES түлхүүрээр шифрлэгддэг. Секторын түлхүүрүүд хэрэглэгчийн оруулсан нэвтрэх үгээс хэрхэн гаргагддаг зэрэг gbde-ийн криптограф дизайны талаар дэлгэрэнгүйг gbde(4)-ээс үзнэ үү.

### 19.16.1.4. Нийцтэй байдлын асуудлууд

sysinstall(8) нь gbde-шифрлэгдсэн төхөөрөмжүүдтэй нийцгүй байдаг. Бүх *.bde төхөөрөмжүүдийг sysinstall(8) эхлүүлэхээс өмнө цөмөөс салгасан байх ёстой бөгөөд ингэхгүй бол энэ нь төхөөрөмжүүдийг шалгаж эхлэхдээ сүйрэх болно. Бидний жишээн дээр ашиглагдсан шифрлэгдсэн төхөөрөмжийг салгахдаа дараах тушаалыг ашиглана:

```
# gbde detach /dev/ad4s1c
```

Мөн vinum(4) нь geom(4) дэд системийг ашигладаггүйг анхааараарай, та gbde-г vinum эзлэхүүнүүдтэй ашиглаж болохгүй.

### 19.16.2. geli-ээр диск шифрлэх нь

Шинэ криптограф GEOM ангилал geli гэж бий. Үүнийг одоогоор Paweł Jakub Dawidek <pjd@FreeBSD.org> хөгжүүлж байгаа болно. geli нь gbde-ээс өөр бөгөөд энэ нь өөр боломжуудыг санал болгож криптограф үйлдэлдээ өөр схемийг ашигладаг.

geli(8)-ийн хамгийн чухал боломжууд бол:

- crypto(9) тогтолцоог ашигладаг - криптограф тоног төхөөрөмж бэлэн болоход geli түүнийг автоматаар ашигладаг.
- Олон криптограф алгоритмуудыг дэмждэг (одоогоор AES, Blowfish, болон 3DES).
- root хуваалтыг шифрлэх боломжийг олгодог. Шифрлэгдсэн root хуваалтад хандахад хэрэглэгддэг нэвтрэх үгийг систем ачаалахад асуудаг.
- Бие биеэндээ хамааралгүй хоёр түлхүүрүүдийг ашиглахыг зөвшөөрдөг (өөрөөр хэлбэл "key" болон "company key").
- geli нь хурдан байдаг. Энгийн сектороос сектор уруу шифрлэлтийг хийдэг.
- Мастер түлхүүрүүдийг нөөцлөх ба сэргээхийг зөвшөөрдөг. Хэрэглэгч өөрийн түлхүүрүүдийг устгах хэрэгтэй болоход нөөцөөс түлхүүрүүдийг сэргээж өгөгдөлд дахин хандах боломжтой болох юм.
- Дискийг санамсаргүй, нэг удаагийн түлхүүрээр залгахыг зөвшөөрдөг - энэ нь swap хуваалтууд болон түр зуурын файлын системүүдэд ашигтай байдаг.

geli-ийн илүү боломжуудыг geli(8)-ийн гарын авлагын хуудаснаас олж болно.

Дараагийн алхмууд нь geli-ийн дэмжлэгийг FreeBSD цөмд хэрхэн идэвхжүүлэхийг тайлбарлах бөгөөд шинэ geli шифрлэлт үзүүлэгчийг хэрхэн үүсгэх болон ашиглах талаар өгүүлэх болно.

Цөмд өөрчлөлтүүд шаардлагатай учраас супер хэрэглэгчийн эрхүүд хэрэгтэй болно.

1. Цөмд geli дэмжлэгийг нэмэх нь

   Дараах мөрийг цөмийн тохиргооны файлд нэмнэ:

   ```
   options GEOM_ELI
   device crypto
   ```

   Цөмөө FreeBSD цөмийг тохируулах нь-д тайлбарласны дагуу дахин бүтээнэ.

   Мөн geli модулийг ачаалах үед дуудаж болно. Дараах мөрийг /boot/loader.conf файлд нэмнэ:

```
geom_eli_load="YES"
```

geli(8) нь одоо цөмд дэмжигдсэн байх ёстой.

2. Мастер түлхүүр үүсгэх нь

   Дараах жишээ нь түлхүүр файлыг хэрхэн үүсгэх талаар тайлбарлах бөгөөд энэ нь /private доор холбогдсон шифрлэгдсэн үзүүлэгчийн Мастер түлхүүрийн хэсэг болон ашиглагдах болно. Түлхүүр файл нь Мастер түлхүүрийг шифрлэхэд хэрэглэгдэх зарим санамсаргүй өгөгдлийг өгөх болно. Мастер түлхүүр нь нэвтрэх үгээр бас хамгаалагдах болно. Үзүүлэгчийн секторын хэмжээ 4kB том байх болно. Мөн энэ хэлэлцүүлэг нь `geli` үзүүлэгчийг хэрхэн залгаж түүн дээр файлын систем үүсгэх, хэрхэн түүнийг холбож ажиллах болон төгсгөлд нь хэрхэн салгах талаар тайлбарлах болно.

   Илүү сайн ажиллагааг хангахын тулд секторын том хэмжээг (4kB ч юм уу) ашиглахыг зөвлөдөг.

   Мастер түлхүүр нь нэвтрэх үгээр хамгаалагдах бөгөөд түлхүүр файлд зориулагдсан өгөгдлийн эх нь /dev/random байх болно. Бидний үзүүлэгч гэж нэрлэдэг /dev/da2.eli -ийн секторын хэмжээ 4kB байх болно.

   ```
   # dd if=/dev/random of=/root/da2.key bs=64 count=1
   # geli init -s 4096 -K /root/da2.key /dev/da2
   Enter new passphrase:
   Reenter new passphrase:
   ```

   Нэвтрэх үг болон түлхүүр файлыг хоёуланг нь заавал ашиглах шаардлагагүй; Мастер түлхүүрийг нууцлах аль ч аргыг тусад нь ашиглаж болно.

   Хэрэв түлхүүр файл "-" гэж өгөгдсөн бол стандарт оролтыг ашиглана. Энэ жишээ нь нэгээс олон түлхүүр файлыг хэрхэн ашиглаж болох талаар харуулж байна.

   ```
   # cat keyfile1 keyfile2 keyfile3 | geli init -K - /dev/da2
   ```

3. Үзүүлэгчийг үүсгэсэн түлхүүрийн тусламжтай залгах нь

   ```
   # geli attach -k /root/da2.key /dev/da2
   Enter passphrase:
   ```

   Шинэ цэвэр текст төхөөрөмж /dev/da2.eli гэж нэрлэгдэх болно.

   ```
   # ls /dev/da2*
   /dev/da2  /dev/da2.eli
   ```

4. Шинэ файлын системийг үүсгэх нь

```
# dd if=/dev/random of=/dev/da2.eli bs=1m
# newfs /dev/da2.eli
# mount /dev/da2.eli /private
```

Шифрлэгдсэн файлын систем df(1)-д харагдаж ашиглахад бэлэн болох ёстой:

```
# df -H
Filesystem     Size   Used  Avail Capacity  Mounted on
/dev/ad0s1a    248M    89M   139M    38%    /
/devfs         1.0K   1.0K     0B   100%    /dev
/dev/ad0s1f    7.7G   2.3G   4.9G    32%    /usr
/dev/ad0s1d    989M   1.5M   909M     0%    /tmp
/dev/ad0s1e    3.9G   1.3G   2.3G    35%    /var
/dev/da2.eli   150G   4.1K   138G     0%    /private
```

5. Үзүүлэгчийг холбосныг салгаж залгасныг болиулах

Шифрлэгдсэн хуваалт дээрх ажил хийгдэж дуусаад /private хуваалт хэрэггүй болох үед `geli` шифрлэгдсэн хуваалтыг холбосноо салган цөмд залгаснаа бас салгах нь зүйтэй юм.

```
# umount /private
# geli detach da2.eli
```

geli(8)-ийг хэрэглэх тухай дэлгэрэнгүй мэдээллийг гарын авлагын хуудаснаас олж болно.

#### 19.16.2.1. geli rc.d скриптийг ашиглах нь

`geli` нь `geli`-ийн хэрэглээг хялбаршуулахад ашиглаж болох rc.d скрипттэй цуг ирдэг. `geli`-г rc.conf(5)-оор тохируулах жишээг дараахаас харж болно:

```
geli_devices="da2"
geli_da2_flags="-p -k /root/da2.key"
```

Энэ нь /dev/da2-г Мастер түлхүүр файл нь /root/da2.key-д байрлах `geli` үзүүлэгч гэж тохируулах бөгөөд үзүүлэгчийг залгаж байхдаа `geli` нь нэвтрэх үгийг ашиглахгүй (хэрэв `-P` сонголт `geli init` хийгдэх үед өгөгдсөн тохиолдолд энэ нь ашиглагдах боломжтой). Систем нь `geli` үзүүлэгчийг цөмөөс систем унтрахаас өмнө салгах болно.

rc.d-г тохируулах талаар дэлгэрэнгүй мэдээлэл энэхүү гарын авлагын rc.d хэсэгт байгаа болно.

# 19.17. Swap зайг шифрлэх

FreeBSD дээрх swap-ийн шифрлэлт нь тохируулахад амархан байдаг. FreeBSD-ийн аль хувилбарыг ашиглаж байгаагаас хамаараад өөр өөр тохируулгууд байдаг бөгөөд тохиргоо нь нэлээн өөр өөр байдаг. gbde(8) эсвэл geli(8) шифрлэх системүүдийг swap-ийг шифрлэхэд ашиглаж болно. Энэ хоёр систем нь хоёулаа encswap rc.d скриптийг ашигладаг.

Өмнөх Дискний хуваалтуудыг шифрлэх хэсэг нь өөр өөр шифрлэх системүүдийн тухай богино хэлэлцүүлгийг агуулдаг.

## 19.17.1. Swap яагаад шифрлэгдэх ёстой вэ?

Дискний хуваалтуудын шифрлэлтийн нэгэн адил swap зайн шифрлэлт нь эмзэг мэдээллийг хамгаалахын тулд хийгддэг. Нууц үгстэй ажилладаг програм байлаа гэж бодъё. Эдгээр нууц үгс нь физик санах ойд байгаа үед бүгд зүгээр байна. Гэхдээ хэрэв үйлдлийн систем нь бусад програмуудад зориулж зайг чөлөөлөхийн тулд санах ойн хуудаснуудыг swap хийж эхэлбэл нууц үгс нь дискний хавтгай дээр шифрлэгдэлгүй бичигдэж болох бөгөөд энэ нь өрсөлдөгчийн хувьд олж авахад амархан болох юм. Swap зайг шифрлэх нь энэ тохиолдолд шийдэл болж болох юм.

## 19.17.2. Бэлтгэл

Энэ хэсгийн үлдсэн хэсэгт ad0s1b нь swap хуваалт байх болно.

Энэ хүртэл swap нь шифрлэгдээгүй байгаа билээ. Аль хэдийн нууц үгс эсвэл бусад эмзэг өгөгдөл дискний хавтгайнууд дээр цэвэр текстээр байж болзошгүй юм. Үүнийг засварлахын тулд swap хуваалт дээрх өгөгдлийг санамсаргүй хаягдлаар дарж бичих хэрэгтэй:

```
# dd if=/dev/random of=/dev/ad0s1b bs=1m
```

## 19.17.3. gbde(8) ашиглан swap шифрлэлт хийх

.bde дагаварыг /etc/fstab-ийн тохирох swap мөр дэх төхөөрөмжид нэмэх шаардлагатай:

```
# Device                Mountpoint      FStype  Options         Dump    Pass#
/dev/ad0s1b.bde         none            swap    sw              0       0
```

## 19.17.4. geli(8) ашиглан swap шифрлэлт хийх

Өөрөөр, swap шифрлэлтэд зориулж geli(8)-г ашиглах процедур нь gbde(8) ашиглахтай төстэй. .eli дагаварыг /etc/fstab-ийн тохирох swap мөр дэх төхөөрөмжид нэмэх шаардлагатай:

```
# Device                Mountpoint      FStype  Options         Dump    Pass#
/dev/ad0s1b.eli         none            swap    sw              0       0
```

geli(8) нь AES алгоритмийг анхдагчаар 128 битийн урттай түлхүүртэй ашигладаг.

Мөн эдгээр анхдагчуудыг /etc/rc.conf файлд `geli_swap_flags` тохируулгыг ашиглан өөрчилж болно. Дараах мөр нь encswap rc.d скриптийг "detach on last close" тохируулгыг зааж 4 килобайтын секторын хэмжээтэй, 128 бит түлхүүрийн урттайгаар Blowfish алгоритм ашиглан geli(8) swap хуваалт үүсгэ гэж хэлж байна:

```
geli_swap_flags="-e blowfish -l 128 -s 4096 -d"
```

geli(8) гарын авлагын хуудсан дахь `onetime` тушаалын тайлбараас боломжит тохируулгуудын жагсаалтыг лавлана уу.

### 19.17.5. Ажиллаж байгаа эсэхийг шалгах

Систем дахин ачаалагдсаны дараа шифрлэгдсэн swap-ийн зөв үйлдлийг `swapinfo` тушаал ашиглан шалгаж болно.

Хэрэв gbde(8) ашиглагдаж байвал:

```
% swapinfo
Device          1K-blocks     Used    Avail Capacity
/dev/ad0s1b.bde    542720        0   542720     0%
```

Хэрэв geli(8) ашиглагдаж байвал:

```
% swapinfo
Device          1K-blocks     Used    Avail Capacity
/dev/ad0s1b.eli    542720        0   542720     0%
```

## 19.18. Highly Available Storage (HAST)

### 19.18.1. Ерөнхий зүйл

Байнгын бэлэн ажиллагаатай байх (High-availability) нь бизнесийн чухал програм хангамжуудын хувьд хамгийн гол шаардлага бөгөөд тийм хадгалалтын төхөөрөмж нь уг орчны түлхүүр хэсэг юм. Highly Available STorage буюу HAST-г Paweł Jakub Dawidek <pjd@FreeBSD.org> хөгжүүлсэн бөгөөд энэ нь өгөгдлийг TCP/IP сүлжээнд холбогдсон хэд хэдэн тусдаа байгаа машинууд дээр тунгалгаар хадгалах боломжийг бүрдүүлдэг. HAST-г сүлжээний RAID1 (толь) гэж ойлгож болох бөгөөд GNU/Linux® тавцангийн хувьд байдаг DRBD® хадгалалтын системтэй төстэй юм. CARP зэрэг FreeBSD-н бусад байнгын бэлэн ажиллагааг хангагч боломжуудын хамтаар HAST нь тоног төхөөрөмжийн эвдрэлээс ангид байнгын бэлэн ажиллагаатай хадгалалтын кластер бүтээх боломжийг бүрдүүлдэг.

Энэ хэсгийг уншаад та дараах зүйлсийг мэдэх болно:

- HAST гэж юу болох, хэрхэн ажилладаг болон ямар боломжуудыг хангадаг талаар.

- FreeBSD дээр HAST-г хэрхэн тохируулж ашиглах талаар.
- CARP болон devd(8)-г хэрхэн тохируулж уян хатан хадгалалтын систем бүтээх талаар.

Энэ хэсгийг уншихаасаа өмнө та дараах зүйлсийг мэдсэн байх шаардлагатай:

- UNIX® болон FreeBSD-ийн үндсийг ойлгосон байх (Юниксийн үндэс).
- Сүлжээний интерфэйсүүд болон FreeBSD-н бусад гол дэд системүүдийг хэрхэн тохируулах талаар мэддэг байх (Тохиргоо ба Тааруулалт).
- FreeBSD-н сүлжээний талаар сайн мэддэг байх (Сүлжээний Холболт).
- FreeBSD 8.1-RELEASE буюу түүнээс хойшхи хувилбарыг ашиглах.

HAST төслийг FreeBSD сан OMCnet Internet Service GmbH болон TransIP BV-н дэмжлэгтэйгээр санхүүжүүлжээ.

## 19.18.2. HAST-н боломжууд

HAST системийн гол боломжуудад:

- Локал хатуу дискний хөтчүүд дээрх I/O алдаануудыг халхлахад ашиглаж болох боломж.
- FreeBSD-н дэмждэг дурын файлын системийг ашиглах боломжийг бүрдүүлдэг файлын системээс үл хамаарсан байдал.
- Тухайн цэг унасан байх үед зөвхөн өөрчлөгдсөн блокуудыг синхрончлох замаар үр ашигтай, хурдан дахин синхрончлох боломж.
- Нэмэлт нөөц бололцоог нэмэхийн тулд аль хэдийн бий болгосон орчинд ашиглах боломж.
- CARP, Heartbeat эсвэл бусад хэрэгслүүдийн хамтаар уян хатан, бөх бат хадгалалтын системийг бүтээхэд цуг ашиглаж боломж.

## 19.18.3. HAST-н ажиллагаа

HAST нь дурын хадгалалтын төхөөрөмжийн хувьд блок түвшний синхрон олшруулалтыг хэд хэдэн машин уруу хийдэг болохоор хамгийн багадаа хоёр цэг (физик машинууд) шаарддаг - Эдгээр нь `анхдагч` (бас `мастер` гэгддэг) цэг болон `хоёрдогч` (`боол`) цэг юм. Энэ хоёр машиныг хамтад нь кластер гэж дуудах болно.

HAST нь одоогоор хоёр кластерын цэгээр хязгаарлагдсан байгаа.

HAST нь анхдагч-хоёрдогч тохиргоогоор ажилладаг болохоор тухайн үед зөвхөн нэг кластерын цэгийг идэвхтэй байхыг зөвшөөрдөг. `анхдагч` цэг буюу `актив` нь HAST-р удирдуулсан төхөөрөмжүүдэд хандах I/O хүсэлтүүдтэй ажилладаг цэг юм. `хоёрдогч` цэг нь дараа нь `primary` цэгээсээ автоматаар синхрончлогддог.

HAST системийн физик хэсгүүд нь:

- локал диск (анхдагч цэг дээрх)
- алсын машин дээрх диск (хоёрдогч цэг)

HAST нь блок түвшинд синхроноор ажилладаг бөгөөд энэ нь файлын системүүд болон програм хангамжуудын хувьд тунгалаг болгодог. HAST нь бусад хэрэгслүүд эсвэл програм хангамжуудад ашиглах боломжтойгоор /dev/hast/ санд ердийн GEOM үйлчилгээ үзүүлэгчдээр хангадаг бөгөөд ингэснээр HAST-р хангагдсан төхөөрөмжүүд болон түүхий дискнүүд, хуваалтууд гэх зэргүүдийн хооронд ялгаа байхгүй болдог байна.

Бичих, устгах, эсвэл flush үйлдэл бүрийг локал диск болон алсын диск рүү TCP/IP ашиглан илгээдэг. Унших үйлдэлд локал диск дээрх мэдээлэл шинэ биш эсвэл I/O алдаа гараагүй л бол локал дискнээс хариу өгдөг. Хэрэв тийм тохиолдол байвал унших үйлдлийг хоёрдогч цэг рүү илгээдэг.

#### 19.18.3.1. Синхрончлол ба Олшруулалтын горимууд

HAST нь унасан байдлаас хурдан сэргээх боломжийг бий болгохыг оролддог. Энэ зорилгоор цэг унасны дараа синхрончлох хугацааг багасгах нь хамгийн чухал юм. Синхрончлолыг хурдан хангахын тулд HAST нь диск дээрх бохир өгөгдлүүдийн битмапыг зохицуулж байдаг бөгөөд ердийн синхрончлолын үед зөвхөн тэдгээрийг синхрончилдог (эхний синхрончлолыг тооцолгүйгээр).

Синхрончлолтой ажиллах олон аргууд байдаг. HAST нь янз бүрийн синхрончлолын аргуудтай (доор дурдсан) ажиллахын тулд хэд хэдэн олшруулалтын горимыг хийдэг:

- *memsync*: локал бичих үйлдэл дууссан ба алсын цэг өгөгдөл ирснийг баталгаажуулсан бөгөөд өгөгдлийг яг хадгалахаас өмнөх үед бичих үйлдэл дууссан гэж үздэг. Алсын цэг дээрх өгөгдөл баталгаажуулалт илгээгдсэний дараа хадгалагддаг. Энэ горим нь саатлыг багасгах зориулалттай бөгөөд маш сайн найдвартай байдлыг хангадаг. *memsync* олшруулалтын горим нь одоогоор хийгдээгүй байгаа.
- *fullsync*: локал болон алсын бичих үйлдэл хийгдэж дууссаны дараа бичих үйлдэл дууссан гэж үздэг. Энэ нь хамгийн найдвартай бөгөөд удаан горим юм. Энэ горим нь анхдагч байдаг.
- *async*: локал бичих үйлдэл дууссан үед бичих үйлдлийг дууссан гэж үздэг. Энэ нь хамгийн хурдан бөгөөд аюултай олшруулалтын горим юм. Энэ горимыг бусад горимын хувьд саатал хэтэрхий өндөр байгаа алсын цэг рүү олшруулахдаа ашиглах ёстой. *async* олшруулалтын горим одоогоор хийгдээгүй байгаа.

Зөвхөн *fullsync* олшруулалтын горим одоогоор дэмжигдсэн.

### 19.18.4. HAST-н тохиргоо

HAST нь ажиллахын тулд `GEOM_GATE` дэмжлэгийг шаарддаг. `GENERIC` цөм нь анхдагчаар `GENERIC` -г *агуулдаггүй* боловч FreeBSD-н анхдагч суулгацад дуудагдах боломжтой geom_gate.ko модуль байдаг. Мөн дараах мөрийг цөмийн тохиргооны файлд нэмэн `GEOM_GATE` дэмжлэгийг цөмд оруулан бүтээж болдог:

```
options GEOM_GATE
```

HAST тогтолцоо нь үйлдлийн системийн өнцгөөс харахад хэд хэдэн хэсгээс тогтдог:

- өгөгдлийн синхрончлолд зориулсан hastd(8) демон,
- хэрэглэгчийн талбарын удирдлагын хэрэгсэл hastctl(8),
- hast.conf(5) тохиргооны файл.

Дараах жишээ хоёр цэгийг `мастер-боол` / `анхдагч-хоёрдогч` үйлдлийн хувьд өгөгдлийг уг хоёр цэгийн хооронд олшруулахын тулд HAST ашиглан тохируулах талаар тайлбарлана. Цэгүүдийг *172.16.0.1* IP хаягтай `hasta` болон *172.16.0.2* IP хаягтай `hastb` гэж нэрлэе. Эдгээр цэгүүд нь HAST үйлдлийн хувьд ижил хэмжээтэй өөр өөрийн /dev/ad6 хатуу дисктэй байна. HAST нөөцийг (заримдаа эх үүсвэр гэгддэг, өөрөөр хэлбэл /dev/hast/ дахь GEOM үйлчилгээ үзүүлэгч) test гэж нэрлэнэ.

HAST-н тохиргоог /etc/hast.conf файлд хийнэ. Энэ файл нь хоёр цэг дээр ижил байх ёстой. Боломжит хамгийн хялбар тохиргоо дараах байдалтай байна:

```
resource test {
    on hasta {
        local /dev/ad6
        remote 172.16.0.2
    }
    on hastb {
        local /dev/ad6
        remote 172.16.0.1
    }
}
```

Илүү дэлгэрэнгүй тохиргооны талаар hast.conf(5) гарын авлагаас үзнэ үү.

Мөн `remote` гэсэн хэсэгт хостын нэрийг ашиглаж бас болно. Тэр тохиолдолд тэдгээр хостуудыг таних боломжтой байх ёстойг анхаараарай, өөрөөр хэлбэл тэдгээр нь /etc/hosts файл юм уу эсвэл локал DNS дээр тодорхойлогдсон байх ёстой.

Одоо хоёр цэг дээр тохиргоо байгаа болохоор HAST нөөцийг үүсгэх боломжтой. Хоёр цэг дээр дараах тушаалыг ажиллуулж эхний мета өгөгдлийг локал диск дээр байрлуулж hastd(8) демонг ажиллуулна:

```
# hastctl create test
# /etc/rc.d/hastd onestart
```

GEOM үйлчилгээ үзүүлэгчдийг одоо байгаа файлын систем дээр ашиглаж *болохгүй* (жишээ нь HAST-р удирдуулсан нөөц рүү одоо байгаа хадгалалтын төхөөрөмжийг хувиргах). Учир нь энэ процедур нь ямар нэг мета өгөгдлийг үйлчилгээ үзүүлэгч дээр хадгалах хэрэгтэй байдаг бөгөөд шаардлагатай зай хангалттай байхгүй байх болно.

HAST цэгийн үүргийг (`анхдагч` эсвэл `хоёрдогч`) администратор тохируулах юм уу эсвэл

hastctl(8) хэрэгсэл ашиглан Heartbeat зэрэг бусад програмаар тохируулж болно. Анхдагч цэг рүү (hasta) шилжээд дараах тушаалыг өгнө:

```
# hastctl role primary test
```

Үүнтэй адилаар хоёрдогч цэг дээр (hastb) дараах тушаалыг ажиллуулна:

```
# hastctl role secondary test
```

Хоёр цэг хоорондоо холбогдож чадахгүй бөгөөд хоёулаа анхдагч гэж тохируулагдсан бол үүнийг split-brain гэж нэрлэдэг. Энэ тохиолдолд алдааг олж засварлахын тулд Split-brain нөхцлөөс сэргэж гарах хэсэгт зааснаар ажиллана.

Цэг бүр дээр hastctl(8) хэрэгсэл ашиглан үр дүнг шалгаж болно:

```
# hastctl status test
```

Хамгийн чухал текст нь status мөр бөгөөд хоёр цэг дээр complete гэж байх ёстой. Хэрэв энэ нь degraded гэж байх юм бол ямар нэг юм болохоо больсныг илтгэнэ. Энэ үед цэгүүдийн хооронд синхрончлол явагдаад эхэлчихсэн байна. hastctl status тушаал dirty гэдэг дээр 0 байт харуулж байвал синхрончлол дууссан гэсэн үг юм.

Дараагийн алхам бол /dev/hast/test GEOM үйлчилгээ үзүүлэгч дээр файлын систем үүсгэж холбох явдал юм. Үүнийг анхдагч цэг (учир нь /dev/hast/test зөвхөн primary цэг дээр харагдана) дээр хийх ёстой. Хатуу дискийн хэмжээнээс хамаарч хэдэн минут үргэлжилж болох юм:

```
# newfs -U /dev/hast/test
# mkdir /hast/test
# mount /dev/hast/test /hast/test
```

HAST тогтолцоог зөв тохируулсны дараа хамгийн сүүлийн алхам бол HAST систем ачаалах үед автоматаар эхэлсэн байхыг шалгах явдал юм. Дараах мөрийг /etc/rc.conf файлд нэмж өгөх шаардлагатай:

```
hastd_enable="YES"
```

#### 19.18.4.1. Нэгээс нөгөөд шилжих тохиргоо

Энэ жишээний зорилго нь өгөгдсөн дурын цэг ажиллахаа больсон тохиолдолд ажиллаж байх уян хатан хадгалалтын систем бүтээх явдал юм. Кластерын анхдагч цэг ажиллахаа болих тохиолдол байж болно. Хэрэв ийм явдал боллоо гэхэд хоёрдогч цэг сааталгүйгээр авч ажиллан файлын системийг шалган холбож өгөгдлийн нэг ч битийг алдалгүйгээр үргэлжлүүлэн ажиллах ёстой.

Энэ зорилтод хүрэхийн тулд FreeBSD-ийн IP давхарга дээр автоматаар шилжих боломжийг бүрдүүлдэг CARP-г ашиглах шаардлагатай байдаг. CARP (Common Address Redundancy Protocol) ижил сүлжээнд олон хостууд IP хаяг хуваалцаж хэрэглэх боломжийг бүрдүүлдэг. CARP-г кластерын цэг бүр дээр Common Address Redundancy Protocol (CARP) хэсэгт зааснаар тохируулна. Үүнийг хийсний дараа цэг бүр хуваалцсан IP *172.16.0.254* гэсэн хаягтай carp0 гэсэн интерфэйстэй байх болно. Кластерын анхдагч HAST цэг мастер CARP цэг байх ёстой.

Өмнөх хэсэгт үүсгэсэн HAST нөөц сүлжээн дэх бусад хостууд руу экспорт хийгдэхэд бэлэн болсон байна. Үүнийг NFS, Samba гэх мэтүүдийн тусламжтайгаар хуваалцсан IP хаяг *172.16.0.254* ашиглан экспорт хийх замаар хийнэ. Шийдэгдээгүй цор ганц асуудал бол анхдагч цэг унахад автоматаар шилжих асуудал юм.

Хэрэв CARP интерфэйсүүд унаж босоод байвал FreeBSD үйлдлийн систем devd(8) үйл явдал гаргах бөгөөд энэ нь CARP интерфэйсүүд дээр төлвийг харах боломжийг бүрдүүлдэг. CARP интерфэйс дээрх төлвийн өөрчлөлт нь аль нэг цэг ажиллахаа байсан эсвэл ажиллаж эхэлснийг харуулна. Эдгээр төлвийн өөрчлөлт нь HAST шилжилтийг автоматаар зохицуулах тусгай скриптийг ажиллуулах боломжтой юм.

CARP интерфэйсүүд дээрх төлвийн өөрчлөлтүүдийг барьж чадахын тулд дараах тохиргоог цэг бүр дээр /etc/devd.conf файлд хийж өгөх ёстой:

```
notify 30 {
    match "system" "IFNET";
    match "subsystem" "carp0";
    match "type" "LINK_UP";
    action "/usr/local/sbin/carp-hast-switch master";
};

notify 30 {
    match "system" "IFNET";
    match "subsystem" "carp0";
    match "type" "LINK_DOWN";
    action "/usr/local/sbin/carp-hast-switch slave";
};
```

Шинэ тохиргоог идэвхтэй болгохын тулд devd(8)-г цэг бүр дээр дахин ажиллуулна:

```
# /etc/rc.d/devd restart
```

carp0 интерфэйс унаж босож байгаа тохиолдолд (өөрөөр хэлбэр интерфэйсийн төлөв өөрчлөлгдсөн тохиолдолд) систем мэдээлэл өгөх бөгөөд энэ нь devd(8) дэд системд скрипт ажиллуулах боломжийг бүрдүүлэх бөгөөд энэ тохиолдолд /usr/local/sbin/carp-hast-switch байна. Энэ скрипт нь автоматаар шилжих шилжилтийг зохицуулах юм. Дээрх devd(8) тохиргооны талаар дэлгэрэнгүйг devd.conf(5) гарын авлагаас үзнэ үү.

Ийм скриптийн жишээ дараах байдлаар байж болох юм:

```
#!/bin/sh

# Original script by Freddie Cash <fjwcash@gmail.com>
# Modified by Michael W. Lucas <mwlucas@BlackHelicopters.org>
# and Viktor Petersson <vpetersson@wireload.net>

# The names of the HAST resources, as listed in /etc/hast.conf
resources="test"

# delay in mounting HAST resource after becoming master
# make your best guess
delay=3

# logging
log="local0.debug"
name="carp-hast"

# end of user configurable stuff

case "$1" in
    master)
        logger -p $log -t $name "Switching to primary provider for ${resources}."
        sleep ${delay}

        # Wait for any "hastd secondary" processes to stop
        for disk in ${resources}; do
            while $( pgrep -lf "hastd: ${disk} \(secondary\)" > /dev/null 2>&1 ); do
                sleep 1
            done

            # Switch role for each disk
            hastctl role primary ${disk}
            if [ $? -ne 0 ]; then
                logger -p $log -t $name "Unable to change role to primary for resource
${disk}."
                exit 1
            fi
        done

        # Wait for the /dev/hast/* devices to appear
        for disk in ${resources}; do
            for I in $( jot 60 ); do
                [ -c "/dev/hast/${disk}" ] && break
                sleep 0.5
            done

            if [ ! -c "/dev/hast/${disk}" ]; then
                logger -p $log -t $name "GEOM provider /dev/hast/${disk} did not
appear."
                exit 1
```

```
            fi
        done

        logger -p $log -t $name "Role for HAST resources ${resources} switched to
primary."

        logger -p $log -t $name "Mounting disks."
        for disk in ${resources}; do
            mkdir -p /hast/${disk}
            fsck -p -y -t ufs /dev/hast/${disk}
            mount /dev/hast/${disk} /hast/${disk}
        done

    ;;

    slave)
        logger -p $log -t $name "Switching to secondary provider for ${resources}."

        # Switch roles for the HAST resources
        for disk in ${resources}; do
            if ! mount | grep -q "^/dev/hast/${disk} on "
            then
            else
                umount -f /hast/${disk}
            fi
            sleep $delay
            hastctl role secondary ${disk} 2>&1
            if [ $? -ne 0 ]; then
                logger -p $log -t $name "Unable to switch role to secondary for
resource ${disk}."
                exit 1
            fi
            logger -p $log -t $name "Role switched to secondary for resource ${disk}."
        done
    ;;
esac
```

Товчхондоо скрипт нь эдгээр алхмуудыг цэг `мастер` / `анхдагч` болох үед хийнэ:

- Өгөгдсөн цэг дээр HAST нөөцийг анхдагч болгоно.
- HAST нөөц доор байгаа файлын системийг шалгана.
- Тохирох газарт нь нөөцүүдийг холбоно.

`нөөц` / `хоёрдогч` болсон тохиолдолд:

- HAST нөөцүүдийг салгана.
- HAST нөөцүүдийг хоёрдогч болгож бууруулна.

Энэ нь зөвхөн боломжой шийдэл байдаг гэдгийг харуулсан жишээ скрипт

гэдгийг санаарай. Энэ нь бүх л боломжит нөхцлийг тооцоогүй бөгөөд шаардлагатай үйлчилгээг эхлүүлэх/зогсоох гэх мэтээр сайжруулж өргөтгөх юм уу эсвэл дурааараа өөрчлөх боломжтой юм.

Энэ жишээн дээр бид стандарт UFS файлын системийг ашигласан. Сэргээхэд шаардлагатай цагийг багасгахын тулд журнал дэмжигдсэн UFS эсвэл ZFS файлын системийг ашиглаж болох юм.

Нэмэлт жишээнүүд бүхий илүү дэлгэрэнгүй мэдээллийг HAST Wiki хуудаснаас олж болно.

## 19.18.5. Алдааг олж засварлах

### 19.18.5.1. Алдааг олж засварлах ерөнхий аргууд

HAST ерөнхийдөө асуудалгүй ажиллах ёстой. Гэхдээ бусад програмуудын нэгэн адил ажиллах ёстойгоосоо өөрөөр ажиллах тохиолдол байдаг. Асуудлын шалтгаан өөр өөр байж болох боловч гол шалгах юм нь кластерын цэгүүдийн хооронд цаг синхрончлогдсон байгаа эсэхийг мэдэх явдал юм.

HAST-н асуудлуудыг шалгаж байхдаа hastd(8)-г `-d` өгөгдөлтэйгээр ажиллуулан hastd(8)-н дибаг түвшинг ихэсгэж үзэх ёстой. Дибаг түвшинг дахин дахин ихэсгэхийн тулд энэ аргументыг олон удаа тавьж өгч болно. Энэ замаар маш их хэрэгтэй мэдээллийг олж авч болно. Мөн `-F` аргументыг ашиглан hastd(8) демонг ил ажиллуулах нь зүйтэй байдаг.

### 19.18.5.2. Split-brain нөхцлөөс сэргэж гарах

`split-brain` гэдэг нь хоёр цэг нэг нэгэнтэйгээ холбогдож чадахгүй бөгөөд хоёул анхдагч цэг гэж тохируулагдсан үе юм. Энэ нь аюултай нөхцөл бөгөөд учир нь энэ тохиолдолд өгөгдөлд хоёр цэг хоёул нийцгүй өөрчлөлтийг хийх боломжийг олгодог. Энэ нөхцлийг системийн администратор гараар засварлах шаардлагатай.

Администратор аль цэг хамгийн чухал өөрчлөлттэй байгаа дээр шийдвэр гаргаж (эсвэл гараар нийлүүлэн) HAST-д эвдэрсэн өгөгдөл бүхий цэгийн бүтэн синхрончлол хийх боломжийг олгох ёстой. Ингэхийн тулд дараах тушаалуудыг дахин синхрончлол шаардлагатай байгаа цэг дээр ажиллуулна:

```
# hastctl role init <resource>
# hastctl create <resource>
# hastctl role secondary <resource>
```

# Chapter 20. GEOM: Διαχείριση Συστοιχιών Δίσκων

## 20.1. Ерөнхий агуулга

Энэ бүлэг нь FreeBSD дээр GEOM тогтолцооны хүрээнд дискийг хэрхэн хэрэглэх талаар хамарсан. Энэ нь тохиргоонд зориулж тогтолцоог ашигладаг гол RAID хянагч хэрэгслүүдийг багтаадаг. Энэ бүлэг нь I/O, доор нь байгаа дэд систем, эсвэл кодыг GEOM хэрхэн зохицуулж эсвэл хянадаг талаар гүнзгий хэлэлцэхгүй. Энэхүү мэдээлэл нь geom(4) болон төрөл бүрийн `SEE ALSO` баримтуудын тусламжтай хангагддаг. Бас энэ бүлэг нь RAID тохиргоонуудын дэлгэрэнгүй заавар биш бөгөөд зөвхөн GEOM-дэмжигдсэн RAID ангиллуудын талаар хэлэлцэх болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Ямар төрлийн RAID дэмжлэг GEOM-ийн хүрээнд байдаг талаар.
- Хэрхэн үндсэн хэрэгслүүдийг ашиглаж тохиргоо хийх, ажиллагааг хангах, болон төрөл бүрийн RAID түвшнүүдийг удирдах талаар.
- Хэрхэн толин тусгал болон судал үүсгэх, шифрлэх, алсаас GEOM-ийн хүрээнд диск төхөөрөмжүүдийг холбох талаар.
- GEOM тогтолцоонд хавсаргасан дискнүүдийн алдааг хэрхэн олж засварлах талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- FreeBSD диск төхөөрөмжүүдийг хэрхэн үздэг талаар ойлгох.
- Шинэ FreeBSD цөм хэрхэн тохируулж суулгах талаар мэдэх.

## 20.2. GEOM-ийн Танилцуулга

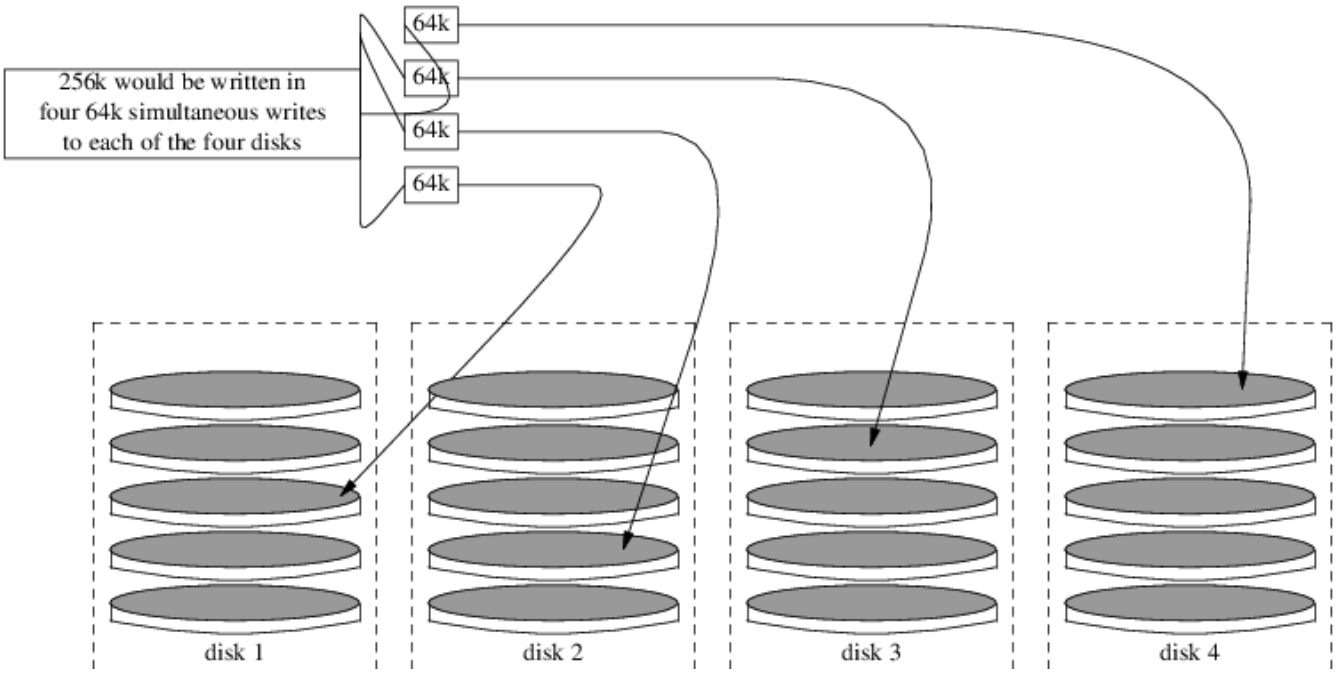
GEOM нь Мастер Ачаалалтын Бичлэгүүд (MBR), BSD хаягууд, гэх мэт - ангиллуудад тухайн ангиллын дэмжигчид (providers) эсвэл /dev дахь тусгай файлуудын тусламжтайгаар хандах хандалт ба хяналтыг уг ангилалд зөвшөөрдөг. GEOM нь төрөл бүрийн програм хангамжийн RAID тохиргоонуудыг ашиглаж үйлдлийн систем болон үйлдлийн системийн хэрэгслүүдэд саадгүйгээр хандах боломж олгодог.

## 20.3. RAID0 - Судал үүсгэх

Судал үүсгэх нь хэд хэдэн дискнүүдийг нэгтгэж нэг эзлэхүүн болгодог. Олон тохиолдолд энэ нь тоног төхөөрөмжийн хянагчуудыг ашиглан хийгддэг. GEOM дэд систем нь диск судал үүсгэх гэж бас нэрлэгддэг RAID0-д зориулсан програмыг хангаж өгдөг.

RAID0 системд, дараалсан дискнүүдэд байгаа өгөгдлүүд нь багц болж хуваагддаг. Систем дээр нэг диск рүү 256k-ийг бичихийг хүлээснээс 64k-ийг 4 өөр диск рүү зэрэг бичих дээд зэргийн I/O буюу оруулах гаргах ажиллагааг RAID0 систем нь санал болгодог. Энэ ажиллагааг олон дискний хянагчуудыг ашиглан нэмэгдүүлж болдог.

Багцалсан олон дискнүүдээс I/O хүсэлтүүд нь зэрэгцэж унших болон бичигддэг учраас RAID0 дискний хуваагдал нь бүгд ижил хэмжээтэй байх ёстой.




**Procedure: Хэлбэржүүлэгдээгүй ATA дискнүүдийн судал үүсгэх**

1. geom_stripe.ko модулийг ачаална:

```
# kldload geom_stripe
```

2. Тохирох холболтын цэг байгааг шалгаарай. Хэрэв энэ эзлэхүүн нь root хуваалт болох ёстой бол түр зуур өөр /mnt гэх мэт холболтын цэгийг ашиглаарай:

```
# mkdir /mnt
```

3. Судал үүсгэх дискнүүдэд зориулсан төхөөрөмжийн нэрийг тодорхойлоод шинэ судал төхөөрөмж үүсгэ. Жишээ нь хоёр ашиглагдаагүй, хуваалт хийгдээгүй /dev/ad2 болон /dev/ad3 гэсэн ATA дискнүүдэд судал үүсгэхийн тулд:

```
# gstripe label -v st0 /dev/ad2 /dev/ad3
Metadata value stored on /dev/ad2.
Metadata value stored on /dev/ad3.
Done.
```

4. Хуваалтын хүснэгт гэгддэг стандарт хаягийг шинэ эзлэхүүн дээр бичээд анхдагч ачаалагдах кодыг суулгаарай:

```
# bsdlabel -wB /dev/stripe/st0
```

5. Энэ процесс хоёр төхөөрөмжийг st0 төхөөрөмжийн хамт /dev/stripe санд үүсгэх ёстой. Тэдгээр нь st0a болон st0c юм. Ингэсний дараа файлын системийг st0a төхөөрөмж дээр `newfs` хэрэгслийн тусламжтайгаар үүсгэж болно:

```
# newfs -U /dev/stripe/st0a
```

Олон тоонууд дэлгэц дээр урсан өнгөрөх бөгөөд хэдхэн хормын дараа процесс төгсөнө. Ингээд эзлэхүүн үүсэж холболт хийгдэхэд бэлэн болно.

Гараар үүсгэсэн дискний судлаа холбохын тулд:

```
# mount /dev/stripe/st0a /mnt
```

Энэ судал үүсгэсэн файлын системдээ ачаалах үед автоматаар холболт хийхийн тулд эзлэхүүний мэдээллийг /etc/fstab файлд хийгээрэй. Энэ жишээн дээр stripe гэсэн байнгын холбох цэгийг үүсгэж байна:

```
# mkdir /stripe
# echo "/dev/stripe/st0a /stripe ufs rw 2 2" \
    >> /etc/fstab
```

/boot/loader.conf файлд мөр нэмж geom_stripe.ko модулийг систем эхлэхэд автоматаар ачаалагдахаар болгох ёстой:

```
# echo 'geom_stripe_load="YES"' >> /boot/loader.conf
```

# 20.4. RAID1 - Толин тусгал үүсгэх

RAID1, буюу *mirroring* нь нэг өгөгдлийг нэгээс илүү диск дээр бичих техникийг хэлдэг. Толин тусгал нь диск ажиллахаа байсан үед өгөгдлийг алдагдахаас хамгаалахад ихэвчлэн хэрэглэгддэг. Толин тусгал дахь диск бүр өгөгдлийн ижил хуулбарыг агуулж байдаг. Тухайн диск ажиллахаа больсон үед ажиллаж байгаа дискнүүд дэх өгөгдлөөр ханган толин тусгал ажилласан хэвээр байна. Компьютер энэ үед ажилласаар байх бөгөөд администратор нь ажиллахаа больсон дискийг хэрэглэгчийн оролцоогүйгээр солих боломжтой болдог.

Эдгээр жишээнүүд дээр хоёр түгээмэл тохиолдлыг харуулсан байгаа. Эхнийх дээр нь хоёр шинэ дискнээс бүтэх толин тусгал үүсгэж одоо байгаа нэг дискний оронд ашиглаж байгаа талаар бичсэн. Хоёр дахь жишээ нь нэг шинэ диск дээр толин тусгал үүсгэж хуучин дискний өгөгдлийг хуулж дараа нь хуучин дискийг толин тусгал руу оруулж байгаа талаар болно. Энэ процедур нь илүү төвөгтэй боловч зөвхөн ганц диск шаарддаг.

Уламжлалаар толин тусгал дахь хоёр диск нь модел болон хэмжээний хувьд ижил байдаг боловч gmirror(8) нь тийм байхыг шаарддаггүй. Өөр дискнүүдээс тогтсон толин тусгал нь хамгийн бага дискний хэмжээтэй ижил байдаг. Илүү том диск дээрх илүүдэл зай нь

ашиглагддаггүй. Толин тусгал руу сүүлд нэмэх диск нь толин тусгал дахь хамгийн бага хэмжээтэй дискний хэмжээтэй ижил байх ёстой.

Энд үзүүлсэн толин тусгал болгож байгаа процедур эвдлэх шинж чанаргүй боловч дискний үйл ажиллагаатай холбоотой учраас бүрэн хэмжээний нөөцлөлтийг эхлээд хийгээрэй.

### 20.4.1. Мета өгөгдөлтэй холбоотой асуудлууд

Дискний олон систем мета өгөгдлийг диск бүрийн төгсгөлд хадгалдаг. Дискийг толин тусгалд дахин ашиглахын тулд хуучин мета өгөгдлийг устгах ёстой. Ихэнх асуудал хуучин үлдсэн мета өгөгдлийн хоёр төрлөөс болж үүсдэг: GPT хуваалтын хүснэгтүүд, болон өмнөх толин тусгалын хуучин gmirror(8) мета өгөгдөл юм.

GPT мета өгөгдлийг gpart(8) ашиглан устгаж болно. Энэ жишээн дээр анхдагч болон нөөц GPT хуваалтын хүснэгтийг ada8 дискнээс устгаж байна:

```
# gpart destroy -F ada8
```

gmirror(8) нь ажиллаж байгаа толин тусгалаас дискийг салгаж мета өгөгдлийг арилгахыг нэг алхмаар хийж чаддаг. Энд жишээ диск ada8-г ажиллаж байгаа толин тусгал gm4-с салгаж байна:

```
# gmirror remove gm4 ada8
```

Хэрэв толин тусгал ажиллахгүй байгаа бөгөөд хуучин толин тусгалын мета өгөгдөл диск дээр байгаа хэвээр бол `gmirror clear` тушаал ашиглан устгаж болно:

```
# gmirror clear ada8
```

gmirror(8) нь мета өгөгдлийн нэг блокийг дискний төгсгөлд хадгалдаг. GPT хуваалтын схем нь мета өгөгдлийг бас дискний төгсгөлд хадгалдаг учраас бүтэн GPT дискүүдийг gmirror(8) ашиглан толин тусгал хийхийг зөвлөдөггүй. Энд MBR хуваалтыг ашигладаг, учир нь энэ хуваалт нь хуваалтын хүснэгтийг дискний зөвхөн эхэнд хадгалдаг бөгөөд gmirror(8)-той харшилддаггүй.

### 20.4.2. Хоёр шинэ дискнээс толин тусгал үүсгэх

Энэ жишээн дээр FreeBSD нь ada0 гэсэн нэг диск дээр суулгасан байгаа. ada1 ба ada2 гэсэн хоёр шинэ диск систем рүү холбогдсон байгаа. Энэ хоёр диск дээр шинэ толин тусгал үүсгэх бөгөөд нэг хуучин дискийг солиход ашиглагдах болно.

gmirror(8) нь цөмд цуг бүтээгдсэн эсвэл ачаалах буюу ажиллах үед дуудагдах geom_mirror.ko цөмийн модулийг шаарддаг. Цөмийн модулийг гараар одоо дуудна:

```
# gmirror load
```

Хоёр шинэ дискээс тогтох толин тусгал үүсгэнэ.

```
# gmirror label -v gm0 /dev/ada1 /dev/ada2
```

gm0 нь шинэ толин тусгалд өгсөн хэрэглэгчийн сонгосон төхөөрөмжийн нэр юм. Толин тусгал эхэлсний дараа төхөөрөмжийн нэр /dev/mirror/ санд гарч ирнэ.

MBR болон bsdlabel хуваалтын хүснэгтүүдийг толин тусгал дээр gpart(8)-н тусламжтай үүсгэж болно. Энд бид /, swap, /var, /tmp, ба /usr гэсэн хуваалтуудтай уламжлалт хуваалттай файлын системийн бүтцийг харуулж байна. Ганц / болон swap хуваалт байж бас болно.

Толин тусгал дахь хуваалтууд нь одоо байгаа дискнүүд шиг ижил хэмжээтэй байх албагүй боловч ada0 дээр байгаа бүх өгөгдлийг агуулах хэмжээтэй байх ёстой.

```
# gpart create -s MBR mirror/gm0
# gpart add -t -a 4k mirror/gm0
# gpart show mirror/gm0
=>       63  156301423  mirror/gm0  MBR  (74G)
         63         63                    - free -  (31k)
        126  156301299                1  freebsd  (74G)
  156301425         61                    - free -  (30k)
```

```
# gpart create -s BSD mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 2g mirror/gm0s1
# gpart add -t freebsd-swap -a 4k -s 4g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 2g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 1g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k       mirror/gm0s1
# gpart show mirror/gm0s1
=>        0  156301299  mirror/gm0s1  BSD  (74G)
          0          2                     - free -  (1.0k)
          2    4194304                   1  freebsd-ufs  (2.0G)
    4194306    8388608                   2  freebsd-swap  (4.0G)
   12582914    4194304                   4  freebsd-ufs  (2.0G)
   16777218    2097152                   5  freebsd-ufs  (1.0G)
   18874370  137426928                   6  freebsd-ufs  (65G)
  156301298          1                     - free -  (512B)
```

Ачаалах кодыг MBR болон bsdlabel-д суулган идэвхтэй зүсмэлийг тохируулж толин тусгалыг ачаалах боломжтой болгоно:

```
# gpart bootcode -b /boot/mbr mirror/gm0
# gpart set -a active -i 1 mirror/gm0
```

```
# gpart bootcode -b /boot/boot mirror/gm0s1
```

Зөөлөн шинэчлэлийг идэвхжүүлэн шинэ толин тусгал дээрх файлын системийг формат хийнэ.

```
# newfs -U /dev/mirror/gm0s1a
# newfs -U /dev/mirror/gm0s1d
# newfs -U /dev/mirror/gm0s1e
# newfs -U /dev/mirror/gm0s1f
```

Эх диск ada0 дээрх файлын системийг толин тусгал руу dump(8) ба restore(8) тушаалыг ашиглан хуулж болно.

```
# mount /dev/mirror/gm0s1a /mnt
# dump -C16 -b64 -0aL -f - / | (cd /mnt && restore -rf -)
# mount /dev/mirror/gm0s1d /mnt/var
# mount /dev/mirror/gm0s1e /mnt/tmp
# mount /dev/mirror/gm0s1f /mnt/usr
# dump -C16 -b64 -0aL -f - /var | (cd /mnt/var && restore -rf -)
# dump -C16 -b64 -0aL -f - /tmp | (cd /mnt/tmp && restore -rf -)
# dump -C16 -b64 -0aL -f - /usr | (cd /mnt/usr && restore -rf -)
```

/mnt/etc/fstab-г шинэ толин тусгал файлын систем рүү заагдсан байхаар засварлах ёстой:

```
# Device        Mountpoint  FStype  Options Dump    Pass#
/dev/mirror/gm0s1a  /       ufs rw  1   1
/dev/mirror/gm0s1b  none        swap    sw  0   0
/dev/mirror/gm0s1d  /var        ufs rw  2   2
/dev/mirror/gm0s1e  /tmp        ufs rw  2   2
/dev/mirror/gm0s1f  /usr        ufs rw  2   2
```

Хэрэв gmirror(8) цөмийн модул цөмд цуг бүтээгдээгүй бол модулийг ачаалахад дуудахын тулд /mnt/boot/loader.conf файлыг засах ёстой:

```
geom_mirror_load="YES"
```

Шинэ толин тусгалыг тест хийхийн тулд системийг дахин ачаалж бүх өгөгдөл хуулагдсан эсэхийг шалгаарай. BIOS толин тусгалыг толин тусгал гэж харахгүй, харин хоёр тусдаа диск гэж харна. Дискнүүд ижил учраас аль нь ачаалахаар сонгогдсон нь чухал биш юм.

Ачаалахад асуудалтай байгаа бол Алдааг олж засварлах хэсгийг үзээрэй. Унтраагаад эхний ada0 дискийг салгаснаар уг дискийг нөөц маягаар байх боломжийг бүрдүүлдэг.

Ашиглаж байгаа үед толин тусгал нь яг л анхны ганц диск шиг л ажиллах болно.

### 20.4.3. Байгаа дискүүд ашиглан толин тусгал үүсгэх

Энэ жишээн дээр FreeBSD ada0 гэсэн нэг диск дээр суусан байгаа. Шинэ диск ada1-г системд холбосон байгаа. Нэг дискээс тогтох толин тусгалыг шинэ диск дээр үүсгэж байгаа системийг түүн рүү хуулж дараа нь хуучин дискийг толин тусгал руу оруулна. Энэ илүү төвөгтэй процедур нь зайлшгүй шаардлагатай байдаг, учир нь gmirror(8) нь 512 байт блок бүхий мета өгөгдлийг диск бүрийн төгсгөлд нэмэх хэрэгтэй байдаг бөгөөд байгаа ada0-н бүх зай ихэвчлэн ашиглагдчихсан байдаг.

gmirror(8) цөмийн модулийг дуудна.

```
# gmirror load
```

diskinfo(8) ашиглан эхний дискний хэмжээг шалгана.

```
# diskinfo -v ada0 | head -n3
/dev/ada0
    512             # sectorsize
    1000204821504   # mediasize in bytes (931G)
```

Шинэ диск дээр толин тусгалыг үүсгэнэ. Эхний дискнээс толин тусгалын хэмжээ их биш гэдгийг шалгахын тулд gnop(8) ашиглан ижил хэмжээтэй хуурамч диск үүсгэж үзнэ. Энэ диск нь ямар ч өгөгдөл хадгалахгүй бөгөөд толин тусгалын хэмжээг хязгаарлахад зөвхөн ашиглагдана. gmirror(8) нь толин тусгал үүсгэхдээ хэдийгээр шинэ диск (ada1) илүү зайтай байсан ч гэсэн gzero.nop-н хэмжээгээр багтаамжийг хязгаарлана. Хоёр дахь мөрөн дээр байгаа *1000204821504* утга нь дээрх diskinfo(8)-р харуулсантай адил ada0-н хэмжээтэй ижил байх ёстой.

```
# geom zero load
# gnop create -s 1000204821504 gzero
# gmirror label -v gm0 gzero.nop ada1
# gmirror forget gm0
```

gzero.nop нь ямар ч өгөгдөл хадгалдаггүй учраас толин тусгал холбогдсон гэж хардаггүй. gzero.nop-той холбоотой мэдээллийг устган холбогдоогүй байгаа зүйлийг "март" гэж толин тусгалд хэлдэг гэсэн үг юм. Үүний үр дүнд толин тусгал нь зөвхөн ганц диск ada1-с тогтоно.

gm0-г үүсгэсний дараа ada0 дээрх хуваалтын хүснэгтийг харна.

Эндэх гаралт нь 1 TB дискнээс юм. Дискний төгсгөлд хуваарилагдаагүй зай байгаа бол өгөгдөл ada0-с шууд шинэ толин тусгал руу хуулагдаж болно.

Гэхдээ гаралт нь дараах жагсаалт дээрх шиг диск дээрх бүх зай бүхлээрээ хуваарилагдсан байгаагаар харуулж байгаа бол 512 байт gmirror(8) мета өгөгдлийн хувьд дискний төгсгөлд зай байхгүй гэсэн үг юм.

```
# gpart show ada0
=>        63  1953525105        ada0  MBR  (931G)
          63  1953525105           1  freebsd  [active]  (931G)
```

Энэ тохиолдолд mirror/gm0 дээрх багтаамжийг нэг секторoop багасгахын тулд хуваалтын хүснэгтийг засварлах хэрэгтэй. Уг процедурыг дараа тайлбарлах болно.

Аль ч тохиолдолд анхдагч диск дээрх хуваалтын хүснэгтүүдийг эхэлж хуулах хэрэгтэй бөгөөд gpart(8) `backup` ба `restore` дэд тушаалуудыг ашиглан гүйцэтгэнэ.

```
# gpart backup ada0 > table.ada0
# gpart backup ada0s1 > table.ada0s1
```

Эдгээр тушаал нь table.ada0 ба table.ada0s1 гэсэн хоёр файл үүсгэнэ. Энэ жишээ нь 1 TB дискнээс юм:

```
# cat table.ada0
MBR 4
1 freebsd         63 1953525105   [active]
```

```
# cat table.ada0s1
BSD 8
1  freebsd-ufs          0    4194304
2 freebsd-swap    4194304   33554432
4  freebsd-ufs   37748736   50331648
5  freebsd-ufs   88080384   41943040
6  freebsd-ufs  130023424  838860800
7  freebsd-ufs  968884224  984640881
```

Хэрэв `gpart show` тушаал дискний төгсгөлд зай байхгүй байгааг харуулж байгаа бол зүсмэл болон сүүлийн хуваалтыг нэг секторoop багасгах хэрэгтэй. Зүсмэл болон сүүлийн хуваалтын зайг нэгээр багасган хоёр файлыг засварлах хэрэгтэй. Эдгээр нь жагсаалт бүрийн сүүлийн тоонууд юм.

```
# cat table.ada0
MBR 4
1 freebsd         63 1953525104   [active]
```

```
# cat table.ada0s1
BSD 8
1  freebsd-ufs          0    4194304
2 freebsd-swap    4194304   33554432
4  freebsd-ufs   37748736   50331648
5  freebsd-ufs   88080384   41943040
```

```
6  freebsd-ufs  130023424  838860800
7  freebsd-ufs  968884224  984640880
```

Дискний төгсгөлд хамгийн багадаа нэг сектор хуваарилагдаагүй байгаа бол эдгээр хоёр файлыг засварлалгүйгээр ашиглаж болно.

Одоо хуваалтын хүснэгтийг mirror/gm0 руу сэргээнэ:

```
# gpart restore mirror/gm0 < table.ada0
# gpart restore mirror/gm0s1 < table.ada0s1
```

`gpart show` тушаалаар хуваалтын хүснэгтийг шалгана. Энэ жишээ нь gm0s1a-г /, gm0s1d-г /var, gm0s1e-г /usr, gm0s1f-г /data1, ба gm0s1g-г /data2-д ашиглана.

```
# gpart show mirror/gm0
=>        63  1953525104  mirror/gm0  MBR  (931G)
          63  1953525042           1  freebsd  [active]  (931G)
  1953525105          62              - free -  (31k)

# gpart show mirror/gm0s1
=>         0  1953525042  mirror/gm0s1  BSD  (931G)
           0     2097152             1  freebsd-ufs  (1.0G)
     2097152    16777216             2  freebsd-swap  (8.0G)
    18874368    41943040             4  freebsd-ufs  (20G)
    60817408    20971520             5  freebsd-ufs  (10G)
    81788928   629145600             6  freebsd-ufs  (300G)
   710934528  1242590514             7  freebsd-ufs  (592G)
  1953525042          63                - free -  (31k)
```

Зүсмэл болон сүүлийн хуваалт диск бүрийн төгсгөлд сул зайтай байх ёстой.

Энэ шинэ хуваалтууд дээр файлын систем үүсгэнэ. Хэдэн хуваалт байх нь эх диск ada0 дээрх хуваалтуудаас хамаарч янз бүр байна.

```
# newfs -U /dev/mirror/gm0s1a
# newfs -U /dev/mirror/gm0s1d
# newfs -U /dev/mirror/gm0s1e
# newfs -U /dev/mirror/gm0s1f
# newfs -U /dev/mirror/gm0s1g
```

Идэвхтэй зүсмэлийг тохируулж MBR болон bsdlabel дээр ачаалах кодыг суулгаж толин тусгалыг ачаалах боломжтой болгоно:

```
# gpart bootcode -b /boot/mbr mirror/gm0
# gpart set -a active -i 1 mirror/gm0
```

```
# gpart bootcode -b /boot/boot mirror/gm0s1
```

Толин тусгал дээрх шинэ хуваалтыг ашиглахаар болгож /etc/fstab файлыг тохируулна. /etc/fstab.orig руу хуулж энэ файлыг эхлээд нөөцөлж авах хэрэгтэй.

```
# cp /etc/fstab /etc/fstab.orig
```

/dev/ada0-г mirror/gm0-р сольж /etc/fstab файлыг засварлана.

```
# Device        Mountpoint  FStype  Options Dump    Pass#
/dev/mirror/gm0s1a  /       ufs rw  1   1
/dev/mirror/gm0s1b  none        swap    sw  0   0
/dev/mirror/gm0s1d  /var        ufs rw  2   2
/dev/mirror/gm0s1e  /usr        ufs rw  2   2
/dev/mirror/gm0s1f  /data1      ufs rw  2   2
/dev/mirror/gm0s1g  /data2      ufs rw  2   2
```

Хэрэв gmirror(8) цөмийн модул цөмд цуг бүтээгдээгүй бол /boot/loader.conf-г засварлаж түүнийг дуудна:

```
geom_mirror_load="YES"
```

Эх диск дээрх файлын системүүдийг одоо dump(8) ба restore(8) тушаалуудыг ашиглан толин тусгал руу хуулж болно. `dump -L` тушаалаар авагдсан файл систем бүрийн хувьд хормын хувилбар үүсгэхэд хугацаа шаардагдаж магадгүйг санаарай.

```
# mount /dev/mirror/gm0s1a /mnt
# dump -C16 -b64 -0aL -f - /    | (cd /mnt && restore -rf -)
# mount /dev/mirror/gm0s1d /mnt/var
# mount /dev/mirror/gm0s1e /mnt/usr
# mount /dev/mirror/gm0s1f /mnt/data1
# mount /dev/mirror/gm0s1g /mnt/data2
# dump -C16 -b64 -0aL -f - /usr | (cd /mnt/usr && restore -rf -)
# dump -C16 -b64 -0aL -f - /var | (cd /mnt/var && restore -rf -)
# dump -C16 -b64 -0aL -f - /data1 | (cd /mnt/data1 && restore -rf -)
# dump -C16 -b64 -0aL -f - /data2 | (cd /mnt/data2 && restore -rf -)
```

ada1-с ачаалж системийг дахин ачаална. Бүгд ажиллаж байгаа бол систем mirror/gm0-с ачаалах бөгөөд энэ нь ada0-н өмнө агуулж байсан өгөгдлийг бас агуулах болно. Ачаалахтай холбоотой асуудал байгаа бол Алдааг олж засварлах хэсгийг харна уу.

Энд хүрэхэд толин тусгал нь зөвхөн ганц диск ada1-с тогтох болно.

mirror/gm0-с амжилттай ачаалсны дараа сүүлийн алхам бол ada0-г толин тусгал руу оруулах явдал юм.

ada0-г толин тусгал руу оруулах үед хуучин агуулга нь толин тусгал дээрх агуулгаар дарагдаж бичигддэг. ada0-г толин тусгал руу нэмэхээсээ өмнө mirror/gm0 нь ada0-н өгөгдлийг яг агуулж байгаа эсэхийг шалгаарай. dump(8) ба restore(8)-г ашиглан хуулсан агуулгатай холбоотой ямар нэг асуудал үүссэн бол файлын системийг ada0-с холбохоор болгож /etc/fstab-г тохируулж дахин ачаалан дээрх алхмуудыг дахин эхнээс нь хийж үзэх хэрэгтэй.

```
# gmirror insert gm0 ada0
GEOM_MIRROR: Device gm0: rebuilding provider ada0
```

Хоёр дискний хоорондох синхрончлол шууд эхэлнэ. gmirror(8) `status` тушаал явцыг харуулна.

```
# gmirror status
      Name    Status  Components
mirror/gm0  DEGRADED  ada1 (ACTIVE)
                      ada0 (SYNCHRONIZING, 64%)
```

Тодорхой хугацааны дараа синхрончлол дуусна.

```
GEOM_MIRROR: Device gm0: rebuilding provider ada0 finished.
# gmirror status
      Name    Status  Components
mirror/gm0  COMPLETE  ada1 (ACTIVE)
                      ada0 (ACTIVE)
```

mirror/gm0 нь одоо ada0 ба ada1 гэсэн хоёр дискээс тогтох бөгөөд агуулгууд нь нэг нэгэнтэйгээ автоматаар синхрончлогдсон байна. Ашиглагдаж байгаа үед mirror/gm0 нь яг л эхний ганц диск шиг ажиллах болно.

## 20.4.4. Алдааг олж засварлах

### 20.4.4.1. Ачаалахтай холбоотой асуудлууд

#### 20.4.4.1.1. BIOS тохиргоо

BIOS тохиргоог шинээр толин тусгал болгосон дискнүүдийн аль нэгнээс ачаалахаар болгож тохируулах хэрэгтэй байж магадгүй юм. Толин тусгалын нэг хэсэг учраас тэдгээр нь ижил өгөгдөлтэй байдаг бөгөөд аль ч дискийг ачаалахад ашиглаж болно.

#### 20.4.4.1.2. Ачаалах үеийн асуудлууд

Хэрэв ачаалалт ийм мэдэгдэлтэй зогссон бол толин тусгал төхөөрөмж асуудалтай байж болох юм:

```
Mounting from ufs:/dev/mirror/gm0s1a failed with error 19.
```

```
Loader variables:
  vfs.root.mountfrom=ufs:/dev/mirror/gm0s1a
  vfs.root.mountfrom.options=rw

Manual root filesystem specification:
  <fstype>:<device> [options]
      Mount <device> using filesystem <fstype>
      and with the specified (optional) option list.

    eg. ufs:/dev/da0s1a
        zfs:tank
        cd9660:/dev/acd0 ro
          (which is equivalent to: mount -t cd9660 -o ro /dev/acd0 /)

  ?               List valid disk boot devices
  .               Yield 1 second (for background tasks)
  <empty line>    Abort manual input

mountroot>
```

geom_mirror модулийг /boot/loader.conf-т оруулж дуудахаа мартсан бол энэ асуудал гарна. Үүнийг засахын тулд FreeBSD 9.0 эсвэл түүнээс хойшхи хувилбарын CD эсвэл USB дискнээс ачаалаад эхний асуулт дээр Shell-г сонгох хэрэгтэй. Дараа нь толин тусгалын модулийг дуудаж толин тусгал төхөөрөмжийг холбох хэрэгтэй:

```
# gmirror load
# mount /dev/mirror/gm0s1a /mnt
```

Толин тусгалын модулийг дуудах мөрийг /mnt/boot/loader.conf-д оруулж засварлана:

```
geom_mirror_load="YES"
```

Файлыг хадгалж дахин ачаална.

error 19 гаргаж байгаа бусад асуудлуудыг засварлахад илүү хүч зарах шаардлагатай. Ачаалах үеийн асууж лавлах дэлгэц дээр ufs:/dev/ada0s1a гэж оруулж өгнө. Систем хэдийгээр ada0-с ачаалах ёстой боловч бүрхүүл сонгох асууж лавлах дэлгэц дахин гарч ирнэ. /etc/fstab файл буруу байгаа учраас тэр юм. Оруулах товчийг энэ үед дарах хэрэгтэй. Толин тусгалын оронд эх дискнээс (ada0) файлын системийг холбохоор болгож /etc/fstab-г тохируулж өөрчлөлтийг буцаах хэрэгтэй. Системийг дахин ачаалж алхмуудыг дахин оролдож үзэх хэрэгтэй.

```
Enter full pathname of shell or RETURN for /bin/sh:
# cp /etc/fstab.orig /etc/fstab
# reboot
```

### 20.4.5. Диск ажиллахаа болиход сэргээх

Толин тусгалын ашигтай тал бол аль нэг диск ажиллахаа байхад толин тусгал дээрх өгөгдөл алдагддаггүй явдал юм. Дээрх жишээн дээр ada0 ажиллахаа болиход ажиллаж байгаа диск болох ada1-с өгөгдлөө өгөн толин тусгал ажилласан хэвээр байх болно.

Ажиллахаа больсон дискийг солихын тулд компьютерыг унтрааж эвдэрсэн дискийг ижил эсвэл илүү хэмжээтэй дискээр солих хэрэгтэй. Үйлдвэрлэгчид дискний хэмжээг гигабайтаар хэмжихдээ дурын утга ашигладаг бөгөөд итгэлтэй байхын тулд зөвхөн `diskinfo -v` тушаалаар харагдах секторын тоог харьцуулах явдал юм. Толин тусгалын хэмжээнээс том диск болох боловч шинэ диск дээрх илүү зай ашиглагдахгүй.

Компьютер ассаны дараа толин тусгал "degraded" буюу "ажиллагаа саарсан" горимд нэг дисктэйгээр ажиллах болно. Толин тусгал нь залгагдаагүй байгаа дискийг мартаж ажиллана:

```
# gmirror forget gm0
```

Хуучин мета өгөгдлийг сольсон дискээс арилгах хэрэгтэй. Дараа нь энэ жишээн дээрх ada4 дискийг толин тусгал руу оруулна:

```
# gmirror insert gm0 /dev/ada4
```

Шинэ диск толин тусгалд ороход дахин синхрончлол хийгдэж эхэлнэ. Толин тусгалын өгөгдлийг шинэ диск рүү хуулах процесс удаж болно. Толин тусгалын үйл ажиллагаа хуулж байх явцад удаашрах учраас компьютерын хэрэглээ бага үед шинэ дискийг хийх нь зүйтэй.

Үйл явцыг `gmirror status` тушаалаар монитор хийж болох бөгөөд синхрончлол хийгдэж байгаа дискнүүд болон ажиллаж байгаа хувийг энэ тушаал харуулдаг. Дахин синхрончлолын үед төлөв `DEGRADED` байж байгаад процесс дуусахдаа `COMPLETE` болж өөрчлөгдөнө.

## 20.5. RAID3 - Parity бүхий байтын түвшний тусдаа судалжуулалт (Striping)

RAID3 нь хэд хэдэн дискийн хөтчийг нэг эзлэхүүн рүү тусгайлан зориулагдсан parity бүхий дискийн хамтаар нэгтгэхэд хэрэглэгддэг арга юм. RAID3 систем дээр өгөгдөл нь хуваагдан тодорхой тооны байтаар массив дахь бүх хөтчүүд рүү бичигддэг бөгөөд ингэхдээ тусгайлан зориулагдсан нэг parity диск рүү бичигддэггүй. Энэ нь RAID3-аас 1024KB уншихад массив дахь бүх диск рүү хандана гэсэн үг юм. Үйл ажиллагааг сайжруулахын тулд олон диск контроллер ашиглаж болно. RAID3 массив нь массив дахь бүх хөтчийн хэмжээний 1 - 1/n бүхий хэмжээгээр хангаж 1 хөтчийн найдвартай ажиллагааг хангадаг. n нь массив дахь хатуу хөтчүүдийн тоо юм. Ийм тохиргоо нь том хэмжээтэй өгөгдөл буюу мультимедиа файлууд хадгалахад зохимжтой байдаг.

RAID3 массивыг бүтээхэд хамгийн багадаа 3 хатуу хөтчийг шаарддаг. Диск бүр ижил

хэмжээтэй байх ёстой бөгөөд учир нь I/O хүсэлт бүр олон диск рүү зэрэг хандалт хийн унших бичих хэрэгтэй байдаг. Бас RAID3-н онцлогоос хамааraад хөтчийн тоо 3, 5, 9, 17, гэх мэтээр (2^n + 1) байх ёстой байдаг.

### 20.5.1. Тусдаа RAID3 массив үүсгэх

FreeBSD дээр RAID3-н дэмжлэг graid3(8) GEOM классаар шийдэгдсэн байдаг. Тусдаа RAID3 массивыг FreeBSD дээр үүсгэхийн тулд дараах алхмууд шаардлагатай.

FreeBSD дээр RAID3 массиваас онолын хувьд ачаалах боломжтой хэдий ч тохиргооны хувьд ховор бас тэгж хийхийг зөвлөдөггүй.

1. Эхдээд дараах тушаалыг ашилаад geom_raid3.ko цөмийн модулийг ачаална:

```
# graid3 load
```

   Мөн өөрөөр гараар geom_raid3.ko модулийг ачаалж болно:

```
# kldload geom_raid3.ko
```

2. Тохирох холбох цэг үүсгэх юм уу эсвэл байгаа эсэхийг шалгана:

```
# mkdir /multimedia/
```

3. Массив руу нэмэгдэх дискүүдийн хувьд хөтчийн нэрийг тогтоогоод RAID3 төхөөрөмж үүсгэнэ. Сүүлд жагсаагдсан төхөөрөмж тусгайлан зориулагдсан parity диск байх болно. Энэ жишээн дээр гурван хуваагдаагүй ATA хөтчийг ашиглаж байна: ada1 ба ada2-г өгөгдөлд, ada3-г parity-г зориулж байна.

```
# graid3 label -v gr0 /dev/ada1 /dev/ada2 /dev/ada3
Metadata value stored on /dev/ada1.
Metadata value stored on /dev/ada2.
Metadata value stored on /dev/ada3.
Done.
```

4. Шинэ үүсгэсэн gr0 төхөөрөмжийг хуваагаад түүн дээр UFS файлын систем үүсгэнэ:

```
# gpart create -s GPT /dev/raid3/gr0
# gpart add -t freebsd-ufs /dev/raid3/gr0
# newfs -j /dev/raid3/gr0p1
```

   Дэлгэц дээгүүр олон тоо урсах бөгөөд хэсэг хугацааны дараа процесс дуусах болно. Эзлэхүүн үүсч холбох боломжтой болно.

```
# mount /dev/raid3/gr0p1 /multimedia/
```

RAID3 массивыг одоо ашиглаж болно.

Дээрх тохиргоог систем дахин ачаалахад алдагдуулахгүйн тулд нэмэлт тохиргоо хэрэгтэй.

1. Массивыг холбохоос өмнө geom_raid3.ko модулийг ачаалах ёстой. Систем эхлэхэд цөмийн модулийг автоматаар ачаалахын тулд дараах мөрийг /boot/loader.conf файлд нэмж өгнө:

```
geom_raid3_load="YES"
```

2. Систем ачаалах процессын явцад массивын файлын системийг автоматаар холбохын тулд дараах эзлэхүүний мэдээллийг /etc/fstab файлд нэмэх хэрэгтэй:

```
/dev/raid3/gr0p1    /multimedia ufs rw  2   2
```

## 20.6. GEOM Хаалга Сүлжээний Төхөөрөмжүүд

GEOM нь хаалга хэрэгслүүдийг ашиглан дискнүүд, CD-ROM-ууд, файлуудыг алсаас ашиглахыг дэмждэг. Энэ нь NFS-тэй адил юм.

Экспортын файл эхэлж үүсгэх шаардлагатай. Энэ файл нь экспорт хийгдсэн эх үүсвэрүүдэд хэнийг хандахыг зөвшөөрсөн болон ямар түвшний хандалтыг тэд өгч байгааг тусгадаг. Жишээ нь эхний SCSI диск дээр 4 дэх зүсмэлийг экспорт хийхийн тулд дараах /etc/gg.exports нь хангалттай:

```
192.168.1.0/24 RW /dev/da0s4d
```

Энэ нь дотоод сүлжээний бүх хостууд da0s4d хуваалт дээрх файлын системд хандах хандалтыг зөвшөөрнө.

Энэ төхөөрөмжийг экспорт хийхдээ тухайн үед холболт хийгдээгүй эсэхийг шалгаад ggated(8) сервер дэмонг ажиллуулаарай:

```
# ggated
```

Хэрэглэх машинаас уг экспортлогдсон төхөөрөмжид `mount` хийхдээ дараах тушаалыг өгнө үү:

```
# ggatec create -o rw 192.168.1.1 /dev/da0s4d
ggate0
```

```
# mount /dev/ggate0 /mnt
```

Одоо төхөөрөмжид /mnt холболтын цэгийг ашиглан хандаж болно.

Хэрэв төхөөрөмж тухайн үед сервер машин эсвэл сүлжээн дэх өөр бусад машин дээр холболт хийгдсэн байсан бол энэ нь амжилтгүй болохыг сануулъя.

Төхөөрөмж дахин шаардлагагүй болоход бусад дискний төхөөрөмжүүдийн нэгэн адил umount(8) тушаалын тусламжтайгаар салгаж болно.

## 20.7. Дискний төхөөрөмжүүдийг хаяглах нь

Системийг эхлүүлэх явцад FreeBSD цөм төхөөрөмжүүдийг олж төхөөрөмжийн цэгүүдийг үүсгэдэг. Төхөөрөмжүүдийг илрүүлэх энэ арга нь зарим нэг асуудлуудтай байдаг, жишээ нь шинэ дискний төхөөрөмж USB-ээр нэмэгдсэн бол яах вэ? Флэш төхөөрөмжид da0 гэсэн төхөөрөмжийн нэр өгөгдөж хуучин байсан da0 нь da1 болж өөрчлөгдөх болов уу. Энэ нь хэрэв файлын системүүд /etc/fstab-д орсон бол тэдгээрийг холбоход асуудал гаргах бөгөөд бүр системийг ачаалахгүй байдалд хүргэж болох юм.

Нэг шийдэл нь SCSI карт руу нэмэгдсэн шинэ төхөөрөмжид төхөөрөмжийн ашиглагдаагүй дугаарууд өгч SCSI төхөөрөмжүүдийг гинжин цуваа болгох явдал юм. Гэхдээ анхдагч SCSI дискийг сольж болзошгүй USB төхөөрөмжүүдийн хувьд яах вэ? Яагаад гэвэл USB төхөөрөмжүүд нь ихэвчлэн SCSI картнуудаас өмнө шалгагддаг. Үүний нэг шийдэл нь эдгээр төхөөрөмжүүдийг системийг ачаалсны дараа залгах явдал юм. Өөр нэг арга нь зөвхөн ганц ATA хөтөч ашиглаж SCSI төхөөрөмжүүдийг /etc/fstab файлд хэзээ ч жагсаахгүй байх тийм шийдэл байж болох юм.

Илүү сайн шийдэл нь `glabel` хэрэгсэл ашиглан дискний төхөөрөмжүүдийг хаяглаж эдгээр хаяг/шошгонуудыг /etc/fstab файлд ашиглаж болох юм. `glabel` нь өгөгдсөн үзүүлэгчийн сүүлийн секторт хаяг/шошгыг хадгалдаг болохоор хаяг/шошго нь дахин ачаалахад хэвээр байх болно. Энэ хаяг/шошгыг төхөөрөмж маягаар ашигласнаар ямар төхөөрөмжийн цэгээр дамжин хандаж байгаагаас үл хамааран файлын систем нь үргэлж холбогдох юм.

`glabel` хэрэгсэл нь түр зуурын болон байнгын хаяг/шошгыг үүсгэж чадна. Хаяг/шошгонуудын ялгаануудын тухай дэлгэрэнгүй мэдээллийг glabel(8) гарын авлагын хуудаснаас үзнэ үү.

### 20.7.1. Хаяг/шошгоны төрлүүд болон жишээнүүд

Байнгын хаяг нь ердийн ба файлын системийн гэсэн хоёр төрлийн байдаг. Файлын системийн байнгын хаягийг tunefs(8) эсвэл newfs(8) тушаалаар үүсгэж болно. Эдгээр хаяг/шошгонуудын төрлүүд /dev сангийн дэд санд үүсгэгддэг бөгөөд файлын системийн төрлийн дагуу нэртэй байна. Жишээ нь UFS2 файлын системийн хаяг/шошгонууд /dev/ufs санд үүсгэгддэг. Ердийн хаягийг `glabel label` тушаалаар үүсгэж болдог. Эдгээр нь файлын системтэй хамааралгүй бөгөөд /dev/label санд үүсгэгдэх болно.

Түр зуурын хаяг/шошго нь дараагийн удаа ачаалахад арилдаг. Эдгээр хаяг/шошгонууд нь /dev/label санд үүсгэгдэх бөгөөд туршилтад тохирсон байдаг. Түр зуурын хаягийг `glabel create` тушаалаар үүсгэж болно.

Өгөгдлийг устгалгүйгээр UFS2 файлын системийн хувьд байнгын хаяг/шошго үүсгэхийн тулд доор дурдсан тушаалуудыг ажиллуулна:

```
# tunefs -L home /dev/da3
```

Хэрэв файлын систем дүүрэн бол энэ нь өгөгдлийн эвдрэлд хүргэж болох юм.

Хаяг/шошго одоо /dev/ufs санд үүссэн байх ёстой бөгөөд түүнийг /etc/fstab файлд нэмж болох юм:

```
/dev/ufs/home       /home           ufs     rw              2       2
```

`tunefs`-г ажиллуулахдаа файлын системийг холбох ёсгүй.

Одоо файлын системийг холбож болно:

```
# mount /home
```

Эндээс эхлэн ачаалах явцад /boot/loader.conf файлын тусламжтайгаар geom_label.ko цөмийн модуль дуудагдсан байхад эсвэл `GEOM_LABEL` цөмийн тохируулга байхад төхөөрөмжийн цэг системд ямар нэгэн буруу нөлөөлөлгүйгээр өөрчлөгдөж болох юм.

Файлын систем нь `newfs` тушаалд өгөгдсөн `-L` тугны тусламжтайгаар анхдагч хаяг/шошготой үүсгэгдэж бас болно. Дэлгэрэнгүй мэдээллийг newfs(8)-с үзнэ үү.

Доор дурдсан тушаалыг хаяг/шошгыг устгахдаа ашиглаж болно:

```
# glabel destroy home
```

Ачаалах дискийн хуваалтуудыг хэрхэн хаяглахыг дараах жишээ харуулж байна.

*Жишээ 31. Ачаалах диск дээрх хуваалтуудыг хаяглах нь*

Ачаалах диск дээрх хуваалтуудыг байнгын байдлаар хаягласнаар дискийг өөр хянагч руу эсвэл өөр систем рүү шилжүүлсэн ч гэсэн систем хэвийнээр ачаалж чаддаг болох юм. Энэ жишээний хувьд системээр ad0 гэж танигдсан ганц ATA диск хэрэглэгдэж байгаа болно. Бас /, /var, /usr ба /tmp болон swap хуваалттай FreeBSD-ийн стандарт хуваалтын хэлбэрийг ашиглаж байгаа болно.

Системийн дахин ачаалж ганц хэрэглэгчийн горимд ачаалж орохын тулд loader(8) тушаал хүлээх мөрөнд `4` товчлуурыг дарна. Дараа нь дараах тушаалуудыг ажиллуулна:

```
# glabel label rootfs /dev/ad0s1a
GEOM_LABEL: Label for provider /dev/ad0s1a is label/rootfs
# glabel label var /dev/ad0s1d
GEOM_LABEL: Label for provider /dev/ad0s1d is label/var
# glabel label usr /dev/ad0s1f
GEOM_LABEL: Label for provider /dev/ad0s1f is label/usr
# glabel label tmp /dev/ad0s1e
GEOM_LABEL: Label for provider /dev/ad0s1e is label/tmp
# glabel label swap /dev/ad0s1b
GEOM_LABEL: Label for provider /dev/ad0s1b is label/swap
# exit
```

Систем нь олон хэрэглэгчийн ачаалах горимоор үргэлжлэх болно. Ачаалж дууссаны дараа /etc/fstab файлыг засварлаж ердийн төхөөрөмжийн нэрсийг тохирох хаягуудаар солиорой. Төгсгөлийн /etc/fstab файл доорхтой адил харагдах болно:

```
# Device                Mountpoint      FStype  Options         Dump    Pass#
/dev/label/swap         none            swap    sw              0       0
/dev/label/rootfs       /               ufs     rw              1       1
/dev/label/tmp          /tmp            ufs     rw              2       2
/dev/label/usr          /usr            ufs     rw              2       2
/dev/label/var          /var            ufs     rw              2       2
```

Системийг одоо дахин ачаалж болно. Хэрэв бүгд зөв болсон бол энэ нь хэвийн гарч ирэх бөгөөд `mount` тушаал дараахийг харуулах болно:

```
# mount
/dev/label/rootfs on / (ufs, local)
devfs on /dev (devfs, local)
/dev/label/tmp on /tmp (ufs, local, soft-updates)
/dev/label/usr on /usr (ufs, local, soft-updates)
/dev/label/var on /var (ufs, local, soft-updates)
```

FreeBSD 7.2-оос эхлэн glabel(8) ангилал UFS файлын системийн хувьд давхардахгүй файлын системийн id `ufsid` дээр тулгуурлан шинэ хаягийн төрлийг дэмждэг. Эдгээр хаягууд нь /dev/ufsid санд байрлах бөгөөд системийн эхлэх явцад автоматаар үүсдэг. Хуваалтуудыг /etc/fstab боломжийг ашиглан холбохын тулд `ufsid` хаягийг ашиглах боломжтой байдаг. Файлын системийн жагсаалт болон тэдгээр файл системийн харгалзах `ufsid` хаягуудыг авахын тулд `glabel status` тушаал ашиглана:

```
% glabel status
                  Name  Status  Components
ufsid/486b6fc38d330916     N/A  ad4s1d
ufsid/486b6fc16926168e     N/A  ad4s1f
```

Дээрхи жишээн дээр ad4s1d нь /var файлын системийг илэрхийлж байгаа бол ad4s1f нь /usr файлын системийг илэрхийлж байна. Үзүүлсэн `ufsid` утгуудыг ашиглан эдгээр хуваалтуудыг /etc/fstab дахь дараах оруулгуудын тусламжтайгаар холбож болох юм:

```
/dev/ufsid/486b6fc38d330916        /var        ufs        rw        2      2
/dev/ufsid/486b6fc16926168e        /usr        ufs        rw        2      2
```

`ufsid` хаягтай ямар ч хуваалтыг энэ аргаар холбож болох юм. Ингэснээр тэдгээрийн хувьд байнгын хаягийг үүсгэх шаардлагагүйгээс гадна төхөөрөмжийн нэрнээс үл хамаарах холболтын ашигтай талыг авсан хэвээр байх болно.

# 20.8. GEOM-ийн тусламжтай UFS тэмдэглэл

FreeBSD 7.0 хувилбараас эхлээд UFS journals буюу тэмдэглэлийн боломж бүрдсэн билээ. Энэхүү шийдэл нь GEOM дэд системийн тусламжтайгаар хийгдсэн бөгөөд gjournal(8) хэрэгслийн тусламжтай хялбар тохируулж болдог.

Journaling буюу тэмдэглэх чадвар нь мета өгөгдөл болон файлын бичилтүүд дискэнд зөвөөр хийгдэхээс өмнө дискний бүрэн гүйцэд бичих үйлдлийг бүтээдэг өөрчлөлтүүдийн бүртгэл зэрэг файлын системийн шилжүүлгүүдийн бүртгэлийг хадгалдаг. Файлын системийн бүрэн бус байдлаас хамгаалахын тулд файлын системийн шилжүүлгүүдийг дахин гүйцэтгэхийн тулд энэхүү шилжүүлгүүдийн бүртгэлийг дараа нь дахин ажиллуулж болдог.

Энэхүү арга нь файлын системийн бүрэн бус байдал болон өгөгдлийн алдагдлын эсрэг хамгаалах өөр нэг арга зам юм. Мета өгөгдлийн шинэчлэлүүд болон файлын системийн дүрс болох Хормын хувилбаруудыг хянаж мөрдүүлж байдаг Зөөлөн Шинэчлэлүүдээс ялгаатай нь бүртгэлийн файл нь энэ үйлдэлд тусгайлан зориулсан дискийн талбарт хадгалагддаг бөгөөд зарим тохиолдолд өөр диск дээр тэр чигээрээ хадгалагдаж болдог.

Өөр бусад файлын системийн тэмдэглэлийн шийдлүүдээс ялгаатай нь `gjournal` арга нь блок дээр тулгуурласан бөгөөд файлын системийн хэсэг болж хийгдээгүй. Энэ нь GEOM өргөтгөл юм.

`gjournal`-д зориулсан дэмжлэгийг идэвхжүүлэхийн тулд FreeBSD цөм нь 7.0 болон түүнээс хойшхи системүүд дээр анхдагчаар байдаг доор дурдсан тохируулгатай байх ёстой:

```
options UFS_GJOURNAL
```

Хэрэв тэмдэглэл үүсгэгдсэн эзлэхүүнүүдийг ачаалж эхлэхэд холбох шаардлагатай бол /boot/loader.conf файлд дараах мөрийг нэмж цөмийн geom_journal.ko модулийг бас дуудах хэрэгтэй болно:

```
geom_journal_load="YES"
```

Мөн дараах мөрийг цөмийн тохиргооны файлд нэмж үүнийг өөрчлөн тохируулсан цөмд бас бүтээж болох юм:

```
options GEOM_JOURNAL
```

Чөлөөтэй файлын систем дээр тэмдэглэл үүсгэхийн тулд одоо доор дурдсан алхмуудыг ашиглан хийж болно. Энэ жишээн дээр da4 нь шинэ SCSI диск байна:

```
# gjournal load
      # gjournal label /dev/da4
```

Энэ үед /dev/da4 төхөөрөмжийн цэг болон /dev/da4.journal төхөөрөмжийн цэг үүссэн байх ёстой. Одоо энэ төхөөрөмж дээр файлын системийг үүсгэж болно:

```
# newfs -O 2 -J /dev/da4.journal
```

Энэ тушаал UFS2 файлын системийг тэмдэглэл идэвхжсэн төхөөрөмж дээр үүсгэх болно.

Төхөөрөмжийг хүссэн цэгтээ доор дурдсан тушаалаар `mount` хийнэ:

```
# mount /dev/da4.journal /mnt
```

Хэд хэдэн зүсмэлүүдийн тухайд тэмдэглэл нь зүсмэл бүр дээр үүсгэгдэх болно. Жишээ нь хэрэв ad4s1 болон ad4s2 нь хоёулаа зүсмэл бол `gjournal` нь ad4s1.journal болон ad4s2.journal үүсгэх болно.

Хурдан ажиллагаатай байхын тулд тэмдэглэлийг өөр диск дээр байлгаж болно. Энэ тохиолдолд тэмдэглэл хангагч эсвэл хадгалалтын төхөөрөмж нь тэмдэглэлийг идэвхжүүлэхийн тулд уг төхөөрөмжийнхөө дараа жагсаагдах ёстой. `tunefs` тушаал ашиглан тухайн байгаа файлын систем дээр тэмдэглэлийг идэвхжүүлж болдог. Гэхдээ файлын системийг өөрчлөхөөсөө өмнө нөөц *үргэлж* хийж байх хэрэгтэй. Ихэнх тохиолдолд `gjournal` нь яг тэмдэглэлийг үүсгэж чадаагүй бол амжилтгүй болох бөгөөд гэхдээ энэ нь `tunefs`-ийг буруугаар ашигласнаас болоод үүссэн өгөгдлийн алдагдлаас хамгаалж чадахгүй юм.

FreeBSD системийн эхлүүлэх дискийг тэмдэглэх бас боломжтой байдаг. Үүний талаарх дэлгэрэнгүй зааврыг Ширээний PC дээр UFS тэмдэглэлийг хийх нь гэсэн нийтлэлээс үзнэ үү.

# Chapter 21. Файлын системийн дэмжлэг

## 21.1. Ерөнхий агуулга

Файлын системүүд нь үйлдлийн системийн бүрэлдэхүүн хэсэг юм. Тэдгээр нь хэрэглэгчдэд файл хуулж хадгалах, өгөгдөлд хандах боломжийг бүрдүүлэх, тэгээд мэдээж хатуу хөтчүүдийг ашигтай болгодог. Өөр өөр үйлдлийн системүүд нь нийтлэг нэг гол үзэлтэй бөгөөд тэр нь тэдгээрийн эх файлын систем юм. FreeBSD дээр энэ файлын систем нь бас UFS гэгддэг эх Unix™ Файлын Систем дээр бүтээгдсэн Түргэн Файлын Систем буюу FFS юм. Энэ нь өгөгдөлд хандахад зориулж хатуу дискүүд дээр байрлуулагддаг FreeBSD дээрх эх файлын систем юм.

FreeBSD нь бас бусад үйлдлийн системийн өгөгдөлд өөр дээрээсээ өөрөөр хэлбэл холбогдсон USB хадгалалтын төхөөрөмжүүд, флэш хөтчүүд болон хатуу дискүүд дээр хадгалагдсан өгөгдөлд хандах боломжийг бүрдүүлэх зорилгоор төрөл бүрийн олон файлын системийг дэмждэг. Бас зарим эх бус файлын системүүдэд зориулсан дэмжлэг байдаг. Эдгээр нь Linux®-ийн Extended буюу Өргөтгөсөн Файлын Систем (EXT), болон Sun™-ий Z Файлын Систем (ZFS) зэрэг өөр үйлдлийн системүүд дээр хөгжүүлэгдсэн файлын системүүд юм.

FreeBSD дээр төрөл бүрийн файлын системүүдэд зориулсан дэмжлэгийн янз бүрийн түвшингүүд байдаг. Зарим нь цөмийн модулийг дуудахыг шаарддаг бол зарим нь хэрэгслүүд суулгахыг шаардаж болох юм. Энэ бүлэг нь FreeBSD-ийн хэрэглэгчдэд өөрийн систем дээрээ Sun™-ий Z файлын системээс эхлээд бусад файлын системүүдэд хандахад туслах зорилгоор хийгдсэн.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Эх болон дэмжигдсэн файлын системүүдийн ялгаа.
- FreeBSD дээр ямар файлын системүүд дэмжигдсэн талаар.
- Эх бус файлын системүүдийг хэрхэн идэвхжүүлж, тохируулж хандан ашиглах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- UNIX® болон FreeBSD-ийн үндсүүдийг (Юниксийн үндэс) ойлгох.
- Цөмийн тохиргоо/эмхэтгэлтийн (FreeBSD цөмийг тохируулах нь) үндсийг мэддэг байх.
- FreeBSD дээр гуравдагчдын програм хангамжийг суулгаж (Програм суулгах. Багцууд болон портууд) чаддаг байх.
- FreeBSD дээрх диск, хадгалалт болон төхөөрөмжийн нэрстэй танилцсан байх (Хадгалалт).

## 21.2. Z Файлын Систем (ZFS)

Sun™-ий хөгжүүлсэн Z файлын систем нь цөөрөм хэлбэрийн хадгалалтын аргыг ашиглахаар хийгдсэн шинэ технолог юм. Энэ нь зай зөвхөн өгөгдөл хадгалалтад хэрэгтэй нөхцөлд ашиглагдана гэсэн үг юм. Энэ нь өгөгдлийн хормын хувилбарууд, олон хуулбарууд, болон өгөгдлийн хяналтын нийлбэрийг дэмжин өгөгдлийн максимум бүрэн бүтэн байдалд зориулагдан бас хийгдсэн юм. RAID-Z гэгддэг өгөгдлийн шинэ хуулбарлах загвар нэмэгдсэн.

RAID-Z загвар нь RAID5-тай төстэй боловч өгөгдлийн бичилтийн эвдрэлтээс хамгаалахаар хийгдсэн.

### 21.2.1. ZFS тааруулалт

ZFS дэд систем нь системийн эх үүсвэрүүдийн ихэнхийг хэрэглэгдэг учир өдөр тутмын хэрэглээнд максимум үр ашгийг хангахад шаардлагатай зарим нэг тааруулалт шаардлагатай байдаг. FreeBSD дээр туршилтын боломж гэгдэж байгаа боловч энэ нь ойрын ирээдүйд өөрчлөгдөж болох юм. Гэхдээ одоогоор дараах алхмуудыг зөвлөдөг.

#### 21.2.1.1. Санах ой

Нийт системийн санах ой хамгийн багаа бодоход нэг гигабайт байх ёстой бөгөөд хоёр болон түүнээс дээшхийг зөвлөдөг. Энд байгаа бүх жишээнүүдэд систем нь нэг гигабайт санах ойтой, хэд хэдэн бусад тааруулах аргын хамтаар байх болно.

Зарим хүмүүс нэг гигабайт санах ойгоос бага санах ойг амжилттай ашиглаж чадсан боловч тийм хязгаарлагдмал физик санах ойтой систем ачаалал ихтэй үед FreeBSD нь санах ойн шавхалтаас болж үймэх нь маш боломжийн байх болно.

#### 21.2.1.2. Цөмийн тохиргоо

Хэрэгцээгүй драйверууд болон тохиргоонуудыг цөмийн тохиргооны файлаас устгахыг зөвлөдөг. Ихэнх төхөөрөмжүүд модуль хэлбэрээр байдаг болохоор тэдгээрийг /boot/loader.conf файлын тусламжтай дуудаж болно.

i386™ архитектурын хэрэглэгчид өөрсдийн цөмийн тохиргооны файлдаа дараах тохиргоог нэмж цөмөө дахин бүтээж дахин ачаалах хэрэгтэй:

```
options     KVA_PAGES=512
```

Энэ тохиргоо нь `vm.kvm_size` тохируулгыг одоо байгаа 1 GB (PAE-ийн хувьд 2 GB) хязгаарыг давах боломжийг бүрдүүлж цөмийн хаягийн талбарыг өргөтгөх болно. Энэ тохиргооны хувьд хамгийн тохиромжтой утгыг олохын тулд мегабайтаар хүссэн хаягийн талбараа дөрөвт (4) хуваана. Энэ тохиолдолд 2 GB-ийн хувьд `512` байна.

#### 21.2.1.3. Дуудагчийн тохируулгууд

kmem хаягийн талбарыг FreeBSD-ийн бүх архитектур дээр ихэсгэх ёстой. Нэг гигабайт физик санах ойтой тест систем дээр /boot/loader.conf файлд хийсэн дараах тохируулгууд болон системийг дахин ачаалсны дараа амжилттай болсон:

```
vm.kmem_size="330M"
vm.kmem_size_max="330M"
vfs.zfs.arc_max="40M"
vfs.zfs.vdev.cache.size="5M"
```

ZFS-тэй холбоотой тохируулгуудын талаар илүү дэлгэрэнгүй зөвлөмжүүдийн жагсаалтыг

http://wiki.freebsd.org/ZFSTuningGuide хаягаас үзнэ үү.

### 21.2.2. ZFS-г ашиглах нь

ZFS цөөрмүүдийг систем эхлэх үед холбох боломжийг FreeBSD-д олгох эхлүүлэх арга зам байдаг. Үүнийг тохируулахын тулд дараах тушаалуудыг ажиллуулна:

```
# echo 'zfs_enable="YES"' >> /etc/rc.conf
# /etc/rc.d/zfs start
```

Энэ баримтын үлдсэн хэсэг 3 SCSI диск байна гэж үзэх бөгөөд тэдгээр төхөөрөмжийн нэрс нь da0, da1 болон da2 юм. IDE тоног төхөөрөмжийн хэрэглэгчид SCSI-ийн оронд ad төхөөрөмжүүдийг ашигладаг.

#### 21.2.2.1. Ганц дискийн цөөрөм

Ганц дискийн төхөөрөмж ашиглан энгийн, нөөцгүй ZFS цөөрөм үүсгэхийн тулд `zpool` тушаалыг ашиглана:

```
# zpool create example /dev/da0
```

Шинэ цөөрмийг харахын тулд `df` тушаалын гаралтыг үзнэ:

```
# df
Filesystem  1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a   2026030  235230  1628718    13%    /
devfs               1       1        0   100%    /dev
/dev/ad0s1d  54098308 1032846 48737598     2%    /usr
example      17547136       0 17547136     0%    /example
```

Энэ гаралт нь `example` цөөрөм зөвхөн үүсгэгдээд зогсохгүй бас *холбогдсон* гэдгийг тодорхой харуулж байна. Энэ нь энгийн файлын систем шиг хандах боломжтой, үүн дээр файлууд үүсгэж хэрэглэгчид дараах жишээн дээрх шиг үзэх боломжтой байна:

```
# cd /example
# ls
# touch testfile
# ls -al
total 4
drwxr-xr-x   2 root  wheel    3 Aug 29 23:15 .
drwxr-xr-x  21 root  wheel  512 Aug 29 23:12 ..
-rw-r--r--   1 root  wheel    0 Aug 29 23:15 testfile
```

Харамсалтай нь энэ цөөрөм ZFS-ийн боломжуудын давуу талыг ашиглахгүй байна. Файлын системийг энэ цөөрөм дээр үүсгэж түүн дээр шахалтыг идэвхжүүлнэ:

```
# zfs create example/compressed
# zfs set compression=gzip example/compressed
```

example/compressed нь одоо ZFS шахагдсан файлын систем боллоо. Зарим нэг том файлуудыг /example/compressed хуулан түүн рүү хуулж үз.

Шахалтыг дараах тушаалаар болиулж болно:

```
# zfs set compression=off example/compressed
```

Файлын системийг салгахын тулд дараах тушаалыг ажиллуулж df хэрэгслээр шалгана:

```
# zfs umount example/compressed
# df
Filesystem  1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a   2026030  235232  1628716    13%    /
devfs               1       1        0   100%    /dev
/dev/ad0s1d  54098308 1032864 48737580     2%    /usr
example      17547008       0 17547008     0%    /example
```

Файлын системийг дахин холбож дахин хандах боломжтой болгоод df тушаалаар шалгана:

```
# zfs mount example/compressed
# df
Filesystem         1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a          2026030  235234  1628714    13%    /
devfs                      1       1        0   100%    /dev
/dev/ad0s1d         54098308 1032864 48737580     2%    /usr
example             17547008       0 17547008     0%    /example
example/compressed  17547008       0 17547008     0%    /example/compressed
```

Цөөрөм болон файлын системийг mount тушаалын гаралтыг шалган бас ажиглаж болно:

```
# mount
/dev/ad0s1a on / (ufs, local)
devfs on /dev (devfs, local)
/dev/ad0s1d on /usr (ufs, local, soft-updates)
example on /example (zfs, local)
example/data on /example/data (zfs, local)
example/compressed on /example/compressed (zfs, local)
```

Ажигласнаар ZFS файлын системүүд нь үүсгэгдсэнийхээ дараагаар жирийн файлын системүүд шигээр ашиглагдаж болно, гэхдээ өөр бусад олон боломжууд бас байдаг. Дараах жишээн дээр шинэ файлын систем data үүсгэгдэж байна. Чухал файлууд энд хадгалагдах учир файлын систем нь өгөгдлийн блок бүрийн хоёр хуулбарыг хадгалахаар

тохируулагдсан:

```
# zfs create example/data
# zfs set copies=2 example/data
```

Одоо df-г дахин ажиллуулж өгөгдөл болон зайн хэрэглээг харах боломжтой:

```
# df
Filesystem         1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a          2026030  235234  1628714    13%    /
devfs                      1       1        0   100%    /dev
/dev/ad0s1d         54098308 1032864 48737580     2%    /usr
example             17547008       0 17547008     0%    /example
example/compressed  17547008       0 17547008     0%    /example/compressed
example/data        17547008       0 17547008     0%    /example/data
```

Цөөрөм дээрх файлын систем бүр ижил хэмжээтэй зайтай байгааг хараарай. Файлын системүүд нь зөвхөн хэрэгцээтэй зайгаа ашиглаж ижил цөөрмөөс авч байгааг харуулахын тулд df тушаалыг энэ жишээнүүд дээр ашигласан шалтгаантай юм. ZFS файлын систем нь эзлэхүүн болон хуваалтууд гэх мэт ойлголтуудаас ангид байж хэд хэдэн файлын систем нэг цөөрмийг эзлэхийг зөвшөөрдөг. Файлын системүүдийг устгаад дараа нь цөөрмийг хэрэгцээгүй болсон болохоор устгана:

```
# zfs destroy example/compressed
# zfs destroy example/data
# zpool destroy example
```

Дискүүд муудаж эвдрэх нь тойрох боломжгүй нөхцөл юм. Энэ диск муудсан үед өгөгдөл алдагдах болно. Эвдэрсэн хатуу дискээс болоод өгөгдлийн алдагдлаас хамгаалах нэг арга нь RAID-г ашиглах явдал юм. ZFS нь энэ боломжийг дараагийн хэсэгт тайлбарлах өөрийн цөөрмийн дизайндаа дэмжсэн байдаг.

#### 21.2.2.2. ZFS RAID-Z

Өмнө хэлсэнчлэн энэ хэсэг нь гурван SCSI диск байгаа гэж үзэх бөгөөд da0 da1 болон da2 (буюу эсвэл IDE диск ашиглагдаж байвал ad0 гэх мэтээр) байна. RAID-Z цөөрөм үүсгэхийн тулд дараах тушаалыг өгнө:

```
# zpool create storage raidz da0 da1 da2
```

Sun™-ий зөвлөснөөр RAID-Z тохиргоонд ашиглагдах төхөөрөмжүүдийн тоо гурваас есийн хооронд байна. Хэрэв та 10 буюу түүнээс олон дискээс тогтсон ганц цөөрөм үүсгэхийг хүсэж байгаа бол түүнийг арай жижиг хэмжээтэй RAID-Z бүлгүүдэд хуваахыг бодох хэрэгтэй. Хэрэв танд зөвхөн хоёр диск байгаа бөгөөд нөөцтэй байхыг хүсэж байгаа бол ZFS толийг ашиглах

> хэрэгтэй. Дэлгэрэнгүйг zpool(8) гарын авлагын хуудаснаас үзнэ үү.

`storage` zpool үүсгэгдсэн байх ёстой. Өмнөх шиг mount(8) болон df(1) тушаалуудыг ашиглан үүнийг шалгаж болно. Дээрх жагсаалтын төгсгөлд өөр илүү дискийн төхөөрөмжүүдийг нэмэн хуваарилж болно. Хэрэглэгчийн файлууд удахгүй байрлах `home` гэгдсэн шинэ файлын системийг цөөрөмд хийнэ:

```
# zfs create storage/home
```

Одоо шахалтыг идэвхжүүлж хэрэглэгчийн гэрийн сангууд болон файлуудын илүү хуулбаруудыг хадгалах боломжтой. Үүнийг өмнөх шиг дараах тушаалуудыг ашиглан хийж болно:

```
# zfs set copies=2 storage/home
# zfs set compression=gzip storage/home
```

Үүнийг хэрэглэгчдийн шинэ гэрийн сан болгохын тулд хэрэглэгчийн өгөгдлийг энэ сан руу хуулж тохирох тэмдэгт холбоосуудыг үүсгэнэ:

```
# cp -rp /home/* /storage/home
# rm -rf /home /usr/home
# ln -s /storage/home /home
# ln -s /storage/home /usr/home
```

Одоо хэрэглэгчдийн хувьд шинээр үүсгэгдсэн /storage/home файлын систем дээр тэдний өгөгдөл хадгалагдсан байх болно. Шинэ хэрэглэгч нэмж тэр хэрэглэгчээр нэвтэрч үзээрэй.

Дараа нь буцааж болох хормын хувилбар үүсгэж үзээрэй:

```
# zfs snapshot storage/home@08-30-08
```

Хормын хувилбар тохиргоо нь гэрийн сан ч биш эсвэл файл ч биш зөвхөн жинхэнэ файлын системийг авахыг анхаараарай. `@` тэмдэгт нь файлын системийн нэр эсвэл эзлэхүүний нэрийн хооронд ашиглагдсан зааглагч юм. Хэрэглэгчийн гэрийн сан устгагдах үед түүнийг доорх тушаалаар сэргээнэ:

```
# zfs rollback storage/home@08-30-08
```

Байгаа хормын хувилбаруудын жагсаалтыг авахын тулд `ls`-г файлын системийн .zfs/snapshot сан дотор ажиллуулна. Жишээ нь өмнө нь авсан хормын хувилбарыг харахын тулд дараах тушаалыг гүйцэтгэнэ:

```
# ls /storage/home/.zfs/snapshot
```

Хэрэглэгчийн өгөгдөл дээр сар бүрийн хормын хувилбар үүсгэх скрипт бичих боломжтой боловч хожим хормын хувилбарууд нь ихээхэн хэмжээний дискийн хэмжээ эзлэх болно. Түрүүчийн хормын хувилбарыг дараах тушаал ашиглан устгаж болно:

```
# zfs destroy storage/home@08-30-08
```

Энэ бүх тестийн дараагаар /storage/home санг бид одоо байгаа төлөвт нь байлгаад байх шалтгаан байх ёсгүй юм. Үүнийг жинхэнэ /home файлын систем болгох хэрэгтэй:

```
# zfs set mountpoint=/home storage/home
```

`df` болон `mount` тушаалуудыг ажиллуулснаар систем бидний файлын системийг жинхэнэ /home сан гэж үзэж байгааг харуулна:

```
# mount
/dev/ad0s1a on / (ufs, local)
devfs on /dev (devfs, local)
/dev/ad0s1d on /usr (ufs, local, soft-updates)
storage on /storage (zfs, local)
storage/home on /home (zfs, local)
# df
Filesystem   1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a    2026030  235240  1628708    13%    /
devfs                1       1        0   100%    /dev
/dev/ad0s1d   54098308 1032826 48737618     2%    /usr
storage       26320512       0 26320512     0%    /storage
storage/home  26320512       0 26320512     0%    /home
```

Ингэснээр RAID-Z тохиргоо дуусаж байна. Шөнө бүрийн periodic(8) ажиллах үед үүссэн файлын системүүдийн талаар төлвийн шинэчлэлтүүдийг авахын тулд дараах тушаалыг ажиллуулна:

```
# echo 'daily_status_zfs_enable="YES"' >> /etc/periodic.conf
```

#### 21.2.2.3. RAID-Z-г сэргээх нь

Програм хангамжийн RAID бүр өөрсдийн `төлвийг` хянах аргатай байдаг. ZFS нь ч мөн адил. RAID-Z төхөөрөмжүүдийн төлвийг дараах тушаалаар харж болно:

```
# zpool status -x
```

Хэрэв бүх цөөрмүүд эрүүл ба бүгд хэвийн бол дараах мэдэгдэл гарна:

```
all pools are healthy
```

Хэрэв асуудал байвал магадгүй диск холболтгүй болчихвол цөөрмийн төлөв иймэрхүү харагдах болно:

```
  pool: storage
 state: DEGRADED
status: One or more devices has been taken offline by the administrator.
    Sufficient replicas exist for the pool to continue functioning in a
    degraded state.
action: Online the device using 'zpool online' or replace the device with
    'zpool replace'.
 scrub: none requested
config:

    NAME        STATE     READ WRITE CKSUM
    storage     DEGRADED     0     0     0
      raidz1    DEGRADED     0     0     0
        da0     ONLINE       0     0     0
        da1     OFFLINE      0     0     0
        da2     ONLINE       0     0     0

errors: No known data errors
```

Энэ нь төхөөрөмжийг администратор салгасан гэдгийг харуулж байна. Ялангуяа энэ жишээний хувьд энэ нь үнэн юм. Дискийг салгахын тулд дараах тушаалыг ашиглажээ:

```
# zpool offline storage da1
```

Одоо системийг унтраасны дараа da1-г солих боломжтой. Систем буцаж ассаны дараа дараах тушаалыг ажиллуулж дискийг солино:

```
# zpool replace storage da1
```

Эндээс төлвийг дахин шалгаж болох бөгөөд энэ удаад төлвийн мэдээллийг авахын тулд `-x` өгөгдөлгүйгээр үзнэ:

```
# zpool status storage
 pool: storage
 state: ONLINE
 scrub: resilver completed with 0 errors on Sat Aug 30 19:44:11 2008
config:

    NAME        STATE     READ WRITE CKSUM
    storage     ONLINE       0     0     0
```

```
      raidz1    ONLINE       0     0     0
        da0     ONLINE       0     0     0
        da1     ONLINE       0     0     0
        da2     ONLINE       0     0     0

errors: No known data errors
```

Энэ жишээнээс харахад бүгд хэвийн харагдаж байна.

#### 21.2.2.4. Өгөгдлийн шалгалт

Өмнө дурдсанчлан ZFS нь хадгалагдсан өгөгдлийн бүрэн бүтэн байдлыг шалгахдаа `хяналтын нийлбэрийг` ашигладаг. Тэдгээр нь файлын системүүдийг үүсгэхэд автоматаар идэвхждэг бөгөөд дараах тушаалыг ашиглан болиулж болно:

```
# zfs set checksum=off storage/home
```

Энэ нь ухаалаг санаа биш боловч хяналтын нийлбэрүүд нь маш бага хадгалалтын зай эзэлдэг бөгөөд идэвжсэн байхдаа илүү ашигтай байдаг. Бас тэдгээрийг идэвхжүүлснээр мэдэгдэхээр зардал бас ажиглагддаггүй. Идэвхжсэн байх үед өгөгдлийн бүрэн бүтэн байдлыг хяналтын нийлбэрийн шалгалт ашиглан ZFS шалгах боломжтой байдаг. Энэ процессийг "цэвэрлэх" гэж хэлдэг. `storage` цөөрмийн өгөгдлийн бүрэн бүтэн байдлыг шалгахын тулд дараах тушаалыг ажиллуулна:

```
# zpool scrub storage
```

Энэ процесс хадгалагдсан өгөгдлийн хэмжээнээс хамаарч ихээхэн хэмжээний хугацаа шаардаж болох юм. Энэ нь бас эдгээр үйлдлүүдийн зөвхөн нэг нь тухайн үед ажиллаж болохоор тийм ихээр I/O-г ашигладаг байна. Цэвэрлэлт дууссаны дараа төлөв шинэчлэгдэж төлвийн хүсэлтийг илгээн харж болно:

```
# zpool status storage
 pool: storage
 state: ONLINE
 scrub: scrub completed with 0 errors on Sat Aug 30 19:57:37 2008
config:

    NAME        STATE     READ WRITE CKSUM
    storage     ONLINE       0     0     0
      raidz1    ONLINE       0     0     0
        da0     ONLINE       0     0     0
        da1     ONLINE       0     0     0
        da2     ONLINE       0     0     0

errors: No known data errors
```

Дууссан хугацаа энэ жишээн дээр цэврээр харуулагдаж байна. Энэ боломж нь өгөгдлийн бүрэн бүтэн байдлыг урт хугацааны турш хадгалахад баталгаа болж тусалдаг.

Z файлын системд зориулсан өөр олон тохиргоонууд байдаг бөгөөд zfs(8) болон zpool(8) гарын авлагын хуудсуудыг үзнэ үү.

#### 21.2.2.5. ZFS квот

ZFS нь өөр төрлийн квотуудыг дэмждэг. Эдгээр нь ref квот, ердийн квот ба бүлгийн квот юм. Энэ хэсэг нь тус бүрийн талаар танилцуулах бөгөөд зарим нэг хэрэглээний зааврууд ыг оруулах болно.

Квот нь өгөгдөл болон түүнээс үүссэн бусад удмын хэрэглэх зайг хязгаарладаг бөгөөд эдгээр удмуудад зориулсан хормын хувилбарууд болон файлын системүүдийн хэрэглэдэг зайнд хязгаарлалт тавьж өгдөг. Хэрэглэгчийн хувьд квот нь тухайн хэрэглэгч ашиглаж болох зайг хязгаарлахад ашигтай байдаг.

Квотыг эзлэхүүн дээр тавьж болдоггүй бөгөөд учир нь `volsize` шинж чанар нь далд квотыг илэрхийлдэг.

ref квот `refquota=size` нь ашиглах зайнд хатуу хязгаар тавин зайнд хязгаарладаг. Гэхдээ энэ хатуу хязгаар нь файлын системүүд эсвэл хормын хувилбарууд зэрэг өгөгдлийн удмуудын хэрэглэдэг зайнд хамааралгүй юм.

storage/home/bob-д зориулж 10 GB ердийн квот тавихын тулд дараахийг ашиглана:

```
# zfs set quota=10G storage/home/bob
```

Тухайн заагдсан хэрэглэгч ашиглаж болох зайг хэрэглэгчийн квот хязгаарлаж өгдөг. Ерөнхий бичих хэлбэр нь `userquota@user=size` байх бөгөөд хэрэглэгчийн нэр дараах хэлбэрийн аль нэг байх ёстой:

- POSIX нийцтэй нэр (жишээ нь *joe*).
- POSIX тоон ID (жишээ нь *789*).
- SID нэр (жишээ нь *joe.bloggs@example.com*).
- SID тоон ID (жишээ нь *S-1-123-456-789*).

Жишээ нь *joe* хэрэглэгчийн хувьд 50 GB квот тавихын тулд дараахийг ашиглана:

```
# zfs set userquota@joe=50G
```

Квотыг устгах эсвэл тавигдаагүй эсэхийг шалгахын тулд дараахийг хэрэглэнэ:

```
# zfs set userquota@joe=none
```

Хэрэглэгчийн квотын шинж чанаруудыг `zfs get all` тушаал харуулдаггүй. `root` биш хэрэглэгчид `userquota` зөвшөөрөл өгөгдөөгүй л бол зөвхөн өөрийн квотуудыг харж болно. Энэ зөвшөөрөлтэй хэрэглэгчид бүгдийн квотыг харж тогтоох боломжтой байна.

Бүлгийн квот нь заагдсан хэрэглэгчийн бүлэг ашиглаж болох зайг хязгаарладаг. Ерөнхий бичиглэл нь `groupquota@group=size` байна.

*firstgroup* бүлэгт 50 GB квот зааж өгөхийн тулд дараахийг ашиглана:

```
# zfs set groupquota@firstgroup=50G
```

*firstgroup* бүлгийн хувьд квотыг устгах эсвэл тавигдаагүй байгаа эсэхийг шалгахын тулд дараахийг хэрэглэнэ:

```
# zfs set groupquota@firstgroup=none
```

Хэрэглэгчийн квот шинж чанарын хувьд `root` биш хэрэглэгчид зөвхөн өөрсдийн харьяалагддаг хэрэглэгчийн бүлэгтэй холбоотой квотуудыг харж чадна, гэхдээ `root` хэрэглэгч эсвэл `groupquota` зөвшөөрөл бүхий хэрэглэгч бүх бүлгийн хувьд квотыг харах болон тавих боломжтой байна.

`zfs userspace` дэд тушаал заагдсан файлын систем эсвэл хормын хувилбар дээр хэрэглэгч бүрийн ашиглаж байгаа зай болон квотын хамтаар харуулдаг. `zfs groupspace` дэд тушаал бүлгийн хувьд адилыг харуулна. Дэмжигдсэн боломжуудын талаар эсвэл тухайн нэг боломжийг зөвхөн харуулъя гэвэл zfs(1) гарын авлагын хуудсыг үзнэ үү.

Хэрэв танд хэрэгтэй зөвшөөрлүүд байгаа, аль эсвэл `root` хэрэглэгч бол storage/home/bob-н хувьд квотыг харуулахын тулд дараахийг ашиглана:

```
# zfs get quota storage/home/bob
```

#### 21.2.2.6. ZFS захиалгууд

ZFS нь хоёр төрлийн зайн захиалгыг дэмждэг. Энэ хэсэг нь тус бүрийн талаар зарим нэг жишээ оролцуулан тайлбарлах болно.

`reservation` шинж чанар нь өгөгдөл болон түүний удамд хэрэгтэй хамгийн бага зайг захиалж нөөцлөх боломжийг бүрдүүлдэг. Энэ нь хэрэв 10 GB зай storage/home/bob дээр тавигдсан бол дискийн зай бага болсон тохиолдолд уг өгөгдөлд хамгийн багадаа 10 GB зай нөөцлөгдсөн байна гэсэн үг юм. `refreservation` шинж чанар нь хормын хувилбарууд зэрэг удмуудыг хасаад зөвхөн уг өгөгдөлд хэрэгтэй хамгийн бага зайг зааж өгөх юм уу тохируулж өгдөг. Жишээ нь storage/home/bob-н хормын хувилбарыг авсан бол үйлдэл амжилттай болохын тулд `refreservation` хэмжээнээс гадна хангалттай дискний зай байх ёстой, яагаад гэвэл гол өгөгдлийн удмууд нь `refreservation` хэмжээнд хамааралгүй бөгөөд өгөгдсөн зайнд баригдахгүй байна.

Аль ч төрлийн захиалгууд ихэнх тохиолдолд ашигтай байдаг бөгөөд жишээлбэл шинэ

систем дээр дискний зайг хуваарилах тохиромжийг турших болон төлөвлөх, эсвэл системийг сэргээх үйл ажиллагаа болон файлуудад зориулсан хангалттай зай файлын системүүд дээр байгаа эсэхэд баттай байхын тулд ашиглаж болно.

`reservation` шинж чанарын ерөнхий бичиглэл нь `reservation=size` байх бөгөөд storage/home/bob-д 10 GB зайг захиалахын тулд дараах тушаалыг ашиглана:

```
# zfs set reservation=10G storage/home/bob
```

Захиалга хийгдээгүй аль эсвэл захиалгыг устгахын тулд дараахийг ашиглана:

```
# zfs set reservation=none storage/home/bob
```

refreservation-г тохируулахад `refreservation=size` гэсэн ерөнхий бичиглэл ижил зарчмаар `refreservation` шинж чанарт үйлчилж болно.

Ямар нэг захиалга эсвэл storage/home/bob-н хувьд захиалга байгаа эсэхийг шалгахдаа дараах тушаалуудын аль нэгийг ажиллуулаарай:

```
# zfs get reservation storage/home/bob
# zfs get refreservation storage/home/bob
```

# 21.3. Linux® файлын системүүд

Энэ хэсэг FreeBSD-н дэмждэг зарим нэг Linux® файлын системүүдийн талаар тайлбарлах болно.

## 21.3.1. Ext2FS

ext2fs(5) файлын системийн цөмийн шийдлийг Годмар Бэк бичсэн бөгөөд драйвер нь анх FreeBSD 2.2 дээр гарч ирсэн. FreeBSD 8 ба түүнээс өмнөх хувилбарууд дээр GNU нийтийн лицензийн доор байдаг бөгөөд харин FreeBSD 9 дээр код бичигдсэн ба BSD лицензээр лицензлэгдсэн байгаа.

ext2fs(5) драйвер нь FreeBSD-н цөмд ext2 файлын систем рүү унших бичих боломжийг олгодог.

Эхлээд цөмийн дуудагдах модулийг ачаална:

```
# kldload ext2fs
```

Дараа нь /dev/ad1s1-д байрлах ext2fs(5) эзлэхүүнийг холбоно:

```
# mount -t ext2fs /dev/ad1s1 /mnt
```

### 21.3.2. XFS

X файлын систем XFS-г анхлан IRIX-д зориулж SGI бичсэн бөгөөд Linux® руу порт хийгдсэн байдаг. Эх код нь GNU нийтийн лицензийн доор гарсан байдаг. Илүү дэлгэрэнгүйг энэ хаягаас үзнэ үү. FreeBSD-н портыг Рассел Каттелан, Alexander Kabaev <kan@FreeBSD.org>, болон Craig Rodrigues <rodrigc@FreeBSD.org> нар эхлүүлсэн.

XFS-г цөмийн дуудагдах модуль хэлбэрээр ачаалахын тулд:

```
# kldload xfs
```

xfs(5) драйвер нь FreeBSD-д XFS файлын систем рүү хандах боломжийг олгодог. Гэхдээ одоогоор зөвхөн унших хандалтыг дэмжсэн байгаа. Эзлэхүүн рүү бичих боломж байхгүй.

/dev/ad1s1-д байрлах xfs(5) эзлэхүүнийг холбохын тулд дараахийг хийнэ:

```
# mount -t xfs /dev/ad1s1 /mnt
```

sysutils/xfsprogs порт нь XFS файлын системийг үүсгэх боломжийг бүрдүүлэх `mkfs.xfs` хэрэгсэл болон бусад анализ хийж засах хэрэгслүүдээс тогддогийг мэдэхэд илүүдэхгүй юм.

`mkfs.xfs`-д `-p` сонголтыг зааж өгөн xfs(5) файлын системийг файлууд болон бусад мета өгөгдлүүдийн хамтаар үүсгэхэд хэрэглэж болно. Ийм замаар зөвхөн унших боложмтой файлын системийг хурдан үүсгэн FreeBSD дээр тест хийхэд хэрэглэж болно.

### 21.3.3. ReiserFS

Reiser файлын систем ReiseFS-г FreeBSD рүү Jean-Sébastien Pédron <dumbbell@FreeBSD.org> порт хийсэн бөгөөд GNU нийтийн лицензийн доор гарсан байдаг.

ReiserFS драйвер нь FreeBSD цөмд ReiserFS файлын системд хандах болон одоогоор зөвхөн түүн дээр байгаа агуулгыг унших (бичиж болохгүй) боломжийг олгодог.

Эхлээд цөмийн дуудагдах модулийг ачаалах хэрэгтэй:

```
# kldload reiserfs
```

Дараа нь /dev/ad1s1-д байгаа ReiserFS эзлэхүүнийг холбохын тулд:

```
# mount -t reiserfs /dev/ad1s1 /mnt
```

# Chapter 22. Vinum Эзлэхүүн Менежер

## 22.1. Ерөнхий агуулга

Та ямар ч дискнүүдтэй байсан гэсэн үргэлж ямар нэгэн болзошгүй асуудлууд байсаар байдаг:

- Тэдгээр дискнүүд нь хэтэрхий жижиг байж болох юм.
- Тэдгээр дискнүүд нь хэтэрхий удаан байж болох юм.
- Тэдгээр дискнүүд нь хэтэрхий найдваргүй байж болох юм.

Эдгээр асуудлуудыг шийдэх төрөл бүрийн шийдлүүд санал болгогдсон бөгөөд хийж гүйцэтгэгдсэн болно. Иймэрхүү асуудлууд гаргахгүйгээр зарим хэрэглэгчдийн өөрсдийгээ хамгаалах нэг арга нь олон болон нөөц дискнүүдийг заримдаа ашиглах явдал юм. Үндсэн FreeBSD систем нь тоног төхөөрөмжөөр дэмжигдсэн RAID системүүдэд зориулсан төрөл бүрийн картууд болон хянагчуудыг дэмжихээс гадна виртуал диск хөтчүүдийг хийдэг блок төхөөрөмжийн драйвер Vinum Эзлэхүүн Менежерийг агуулдаг. *Vinum* нь эдгээр гурван асуудлыг шийддэг *Эзлэхүүн Менежер* гэж нэрлэгддэг виртуал дискний драйвер юм. Vinum нь уян хатан чанар, ажиллагаа болон найдвартай байдлыг уламжлалт диск хадгалалтаас илүүтэй хангадаг бөгөөд RAID-0, RAID-1 болон RAID-5 загваруудыг дангаар нь болон хослолынх нь хувьд шийддэг.

Энэ бүлэг нь уламжлалт диск хадгалалтын болзошгүй асуудлуудыг тоймлон өгүүлэх бөгөөд Vinum Эзлэхүүн Менежерийн талаар танилцуулах болно.

FreeBSD 5-аас эхлэн Vinum-ийг GEOM архитектурт (GEOM. Модульчлагдсан Диск Хувиргах Тогтолцоо) тааруулан оруулахын тулд анхдагч санаанууд, нэр томъёо болон диск дээрх мета өгөгдлийг нь хадгалан дахин бичсэн юм. Энэхүү дахин бичилт нь *gvinum* (*GEOM vinum*-д зориулсан) гэгддэг. Дараах баримт нь шийдлийн хувилбаруудаас үл хамааран *Vinum*-ийг хийсвэр нэр маягаар ихэвчлэн хэрэглэх болно. Ямар ч тушаалыг одоо ажиллуулахдаа `gvinum` тушаалыг ашиглах ёстой бөгөөд цөмийн модулийн нэр нь vinum.ko geom_vinum.ko болж өөрчлөгдсөнөөс гадна төхөөрөмжийн бүх цэгүүд /dev/vinum дотор биш /dev/gvinum дотор байрласан байгаа. FreeBSD 6-аас эхлэн хуучин Vinum шийдэл нь үндсэн кодонд байхаа аль хэдийн больсон байна.

## 22.2. Дискнүүд хэтэрхий жижиг

Дискнүүд нь хэмжээний хувьд улам томорсоор байгаагийн адил бас өгөгдлийн хадгалалтын шаардлагууд ч бас тийм болсоор байна. Ихэнх тохиолдолд танд өөрт чинь байгаа дискнүүдээс илүү том файлын систем хэрэгтэй болж магадгүй юм. Энэ асуудал нь арван жилийн өмнөх шиг тийм хурц асуудал биш болсон боловч одоо болтол байгааг нь хүлээн зөвшөөрөхөөс өөр аргагүй юм. Зарим системүүд нь үүнийг өөрийнхөө өгөгдлийг хэд хэдэн дискнүүд дээр хадгалах хийсвэр төхөөрөмж үүсгэн шийддэг байна.

# 22.3. Хандалтын тагларалтууд

Орчин үеийн системүүдэд маш өндөр зэрэгцээ хандлагаар өгөгдөлд хандах хэрэгцээ байнга гардаг. Жишээ нь том FTP эсвэл HTTP серверүүд нь олон мянган зэрэгцээ сессүүдтэй ажиллаж чаддаг ба ихэнх дискнүүдийн тогтвортой дамжуулах түвшингээс хавьгүй илүү гарсан гадагшаа ертөнц уруу холбогдсон олон 100 Mbit/s-ийн холболтуудтай байдаг.

Одоогийн диск хөтчүүд өгөгдлийг цуваагаар 70 MB/s хүртэл дамжуулж чаддаг боловч энэ утга нь бие биенээсээ хамааралгүй олон чөлөөтэй процессууд хөтчид хандаг тийм орчинд тийм ч чухал биш юм. Тийм орчинд процессууд нь тэдгээр утгуудын зөвхөн бутархайгаар хэмжигдэх тооны хурдад хүрч магадгүй юм. Ийм тохиолдлуудад асуудлыг дискний дэд системийн үүднээс харвал илүү сонирхолтой байх болно: энд чухал параметр нь дэд системд үзүүлж байгаа дамжуулалтын ачаалал бөгөөд өөрөөр хэлбэл дамжуулалтад оролцож байгаа хөтчүүдийг эзэлж байгаа дамжуулалтын тэр хугацаа юм.

Аль ч дискний дамжуулалтын үед хөтөч нь толгойнуудаа эхлээд байрлуулж унших толгойн доор эхний сектор шалгагдан өнгөрөхийг хүлээж дараа нь дамжуулалтыг гүйцэтгэх ёстой. Эдгээр үйлдлүүд нь atomic буюу хуваагдашгүй, салгагдашгүй үйлдлүүд гэж ойлгогддог: тэдгээрийг тасалдуулах нь ямар ч утгагүй юм.

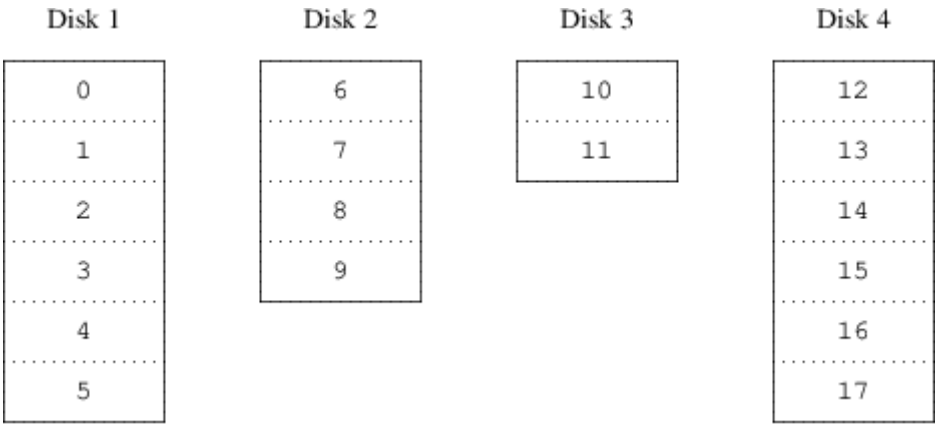
Ердийн 10 kB-ийн дамжуулалтыг авч үзье: өндөр ажиллагаатай одоо үеийн дискнүүд нь толгойнуудаа дунджаар 3.5 ms-д байрлуулж чаддаг. Хамгийн хурдан хөтчүүд 15,000 rpm хурдаар эргэдэг, тийм болохоор дундаж эргэх хожимдол нь (хувьсгалын хагас) 2 ms болох юм. 70 MB/s үед дамжуулалт нь өөрөө 150 μs болох бөгөөд байрлуулах хугацаатай харьцуулахад бараг юу ч биш юм. Ийм тохиолдолд үр дүнтэй дамжуулах түвшин нь 1 MB/s-с бага зэрэг илүү түвшин уруу буурах бөгөөд мэдээж энэ нь дамжуулах хэмжээнээс ихээхэн хамааралтай байна.

Энэ асуудлын уламжлалт, мэдээжийн шийдэл нь "илүү ээрүүлүүд" юм: нэг том диск ашиглахын оронд энэ нь нийт хадгалалтын хэмжээ нь адилхан хэд хэдэн арай жижиг дискнүүдийг ашиглах явдал юм. Диск бүр нь тус тусдаа байрлуулах болон дамжуулах үйлдлийг хийж чадах бөгөөд ингэснээр үр дүнтэй дамжуулах чанар нь ашигласан дискний тоотой ойролцоо хүчин зүйлээр нэмэгдэх болно.

Дамжуулалтын чанарын яг тодорхой сайжруулалт нь мэдээж оролцсон дискнүүдийн тооноос бага байна: хөтөч бүр нь зэрэгцээ дамжуулж чадах боловч хүсэлтүүд нь хөтчүүдийн дагуу тэнцүү түгээгдэхийг батлах аргагүй юм. Нэг хөтөч дээрх ачаалал нь нөгөө дээрхээс илүү өндөр байх нь зайлшгүй юм.

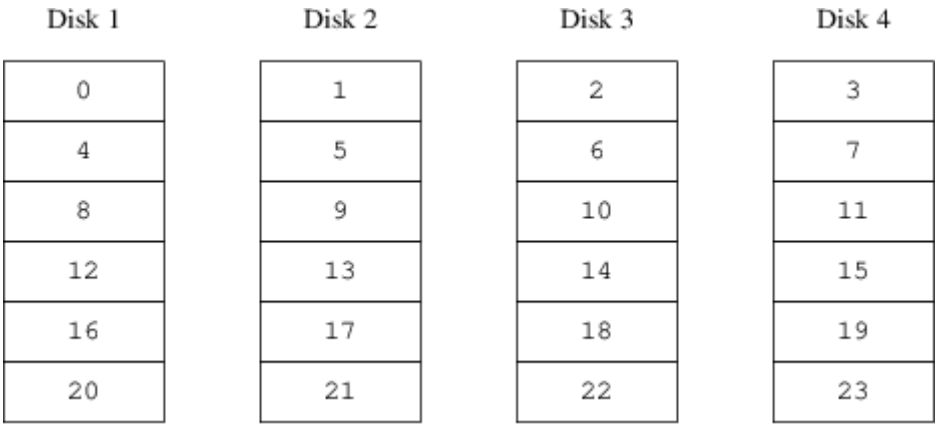
Дискнүүд дээрх ачааллын тэнцүү байдал нь хөтчүүдийн дагуу өгөгдлүүд хуваалцдаг арга замаас маш ихээр хамаардаг. Дараах хэлэлцүүлэг дээр дискний хадгалалтыг номны хуудас шигээр дугаараар нь хандаж болох маш олон тооны өгөгдлийн секторууд гэж ойлгох нь тохиромжтой юм. Хамгийн мэдээжийн арга бол том номыг аваад жижиг хэсгүүд болгон урахын оронд виртуал дискийг физик дискнүүдийн хэмжээтэйгээр дараалсан секторуудын бүлгүүд болгон хувааж энэ маягаар тэдгээрийг хадгалах явдал юм. Энэ аргыг *concatenation буюу нийлүүлэлт* гэж нэрлэдэг бөгөөд дискнүүд нь ямар нэгэн тусгайлсан хэмжээний харилцаанууд шаарддаггүйгээрээ давуу талтай юм. Виртуал диск дэх хандалт өөрийн хаягийн талбараар тэнцүү тархсан үед энэ нь сайн ажилладаг. Хандалт нь жижиг талбарт төвлөрсөн үед сайжруулалт тийм ч сайн мэдэгддэггүй. Нийлүүлэгдсэн зохион байгуулалт нь

хадгалалтын нэгжүүд нийлүүлэгдсэн зохион байгуулалтад хуваарилагдаж байгаа дарааллыг үзүүлж байна.




*Зураг 102. Нийлүүлэгдсэн зохион байгуулалт*

Өөр нэг тааруулалт нь хаягийн талбарыг арай жижиг, тэнцүү хэмжээтэй бүрэлдэхүүн хэсэгт хуваах тэдгээрийг цуваагаар өөр төхөөрөмжүүд дээр хадгалах явдал юм. Жишээ нь эхний 256 сектор эхний диск дээр, дараагийн 256 сектор нь дараагийн диск дээр гэх мэтчилэн байх юм. Сүүлийн дискэнд хүрсний дараа процесс дискнүүдийг дүүртэл давтагдана. Энэ тааруулалтыг *striping буюу судал үүсгэх* эсвэл RAID-0 гэж нэрлэдэг. . Судал үүсгэх нь зарим талаараа өгөгдлийг олохын тулд илүү чармайлт шаарддаг бөгөөд олон дискнүүдийн дагуу тархсан дамжуулалтад нэмэлт I/O ачаалал үүсгэж болох боловч энэ нь бас тогтмол ачааллыг илүүтэйгээр дискнүүдийн дагуу хангаж өгдөг. Судалчлагдсан зохион байгуулалт нь хадгалалтын нэгжүүд судалчлагдсан зохион байгуулалтад хуваарилагдаж байгаа дарааллыг үзүүлж байна.




*Зураг 103. Судалчлагдсан зохион байгуулалт*

# 22.4. Өгөгдлийн бүрэн бүтэн байдал

Одоогийн дискнүүд дэх сүүлийн асуудал нь найдваргүй байдал юм. Диск хөтчүүдийн найдвартай байдал сүүлийн хэдэн жилийн дотор асар ихээр нэмэгдсэн боловч тэдгээр нь сервер унах гол шалтгааны нэг бүрэлдэхүүн хэсэг хэвээр байсаар байна. Ингэсэн тохиолдолд гарах үр дүн нь сүйрлийн байж болох юм: ажиллагаагүй болсон дискний хөтчийг сольж шинэ диск уруугаа өгөгдлийг сэргээх нь хэдэн өдөр болж болох юм.

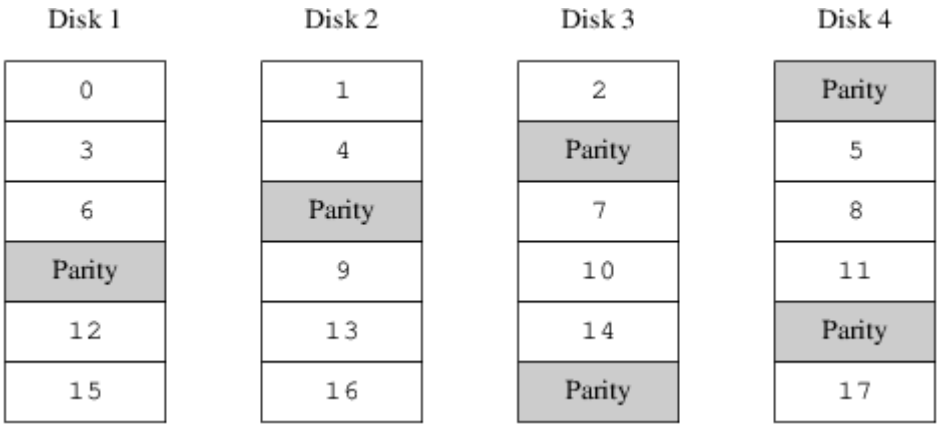
Уламжлалт замаар энэ асуудлыг *толин тусгал үүсгэх* аргаар өгөгдлийн хоёр хуулбарыг өөр физик тоног төхөөрөмж дээр хадгалж шийддэг. RAID түвшингүүд бий болсноор энэ техник

нь бас RAID түвшин 1 буюу RAID-1 гэж нэрлэгддэг. Эзлэхүүн уруу хийгдэх бичилт хоёр байрлал уруу хийгддэг; уншилт нь алинаас нь ч хийгдсэн болдог, хэрэв нэг хөтөч нь ажиллахгүй болбол өгөгдөл нөгөө хөтөч дээр бас байж байх юм.

Толин тусгал үүсгэх нь хоёр асуудалтай байдаг:

- Үнэ. Энэ нь нөөцгүй шийдлийн нэгэн адил хоёр дахин их дискний хадгалалт шаарддаг.
- Ажиллагааны нөлөөлөл. Бичилтүүд нь хоёр хөтчүүдэд хоёуланд нь хийгддэг, тиймээс тэдгээр нь толин тусгал хийгдээгүй эзлэхүүний зурвасын өргөнөөс хоёр дахин ихийг эзэлдэг. Уншилтууд нь харин ажиллагааны хувьд асуудалгүй байдаг: тэд илүү хурдан юм шиг харагддаг.

Өөр нэг шийдэл нь RAID түвшингүүд 2, 3, 4 болон 5-д хийгдсэн *parity* юм. Эдгээрээс RAID-5 нь хамгийн сонирхолтой байдаг. Энэ нь Vinum-д хийгдсэн бөгөөд судал бүрийн нэг блокийг өөр блокуудын parity-д зориулдаг судалчлагдсан зохион байгуулалтын нэг хувилбар юм. RAID-5 plex нь Vinum-д хийгдсэн бөгөөд судалчлагдсан plex-ийн нэгэн адил боловч энэ нь судал бүр дэх parity блокуудыг оруулан RAID-5-г шийдвэрлэдэг. RAID-5-ийн шаарддагаар энэ parity блокийн байрлал нь нэг судлаас дараагийнх уруу өөрчлөгддөг. Өгөгдлийн блокууд дахь тоонууд нь блокийн харьцангуй дугаарыг илэрхийлдэг.




*Зураг 104. RAID-5 зохион байгуулалт*

Толин тусгал үүсгэхтэй харьцуулахад RAID-5 нь хамаагүй бага хадгалалтын зай шаарддагаараа давуу талтай юм. Унших хандалт нь судалчлагдсан зохион байгуулалтуудын адил байдаг боловч бичих хандалт нь хамаагүй удаан ойролцоогоор унших хурдны 25% байдаг. Хэрэв нэг хөтөч ажиллагаагүй болбол массив нь доройтсон горимд үргэлжлэн ажиллах болно: үлдсэн хандаж болох хөтчүүдийн аль нэгнээс уншилт хэвийн үргэлжлэх боловч ажиллагаагүй болсон хөтчөөс хийгдэх уншилт бүх үлдсэн хөтчүүдийн харгалзах блокоос дахин тооцоологдох болно.

# 22.5. Vinum обьектууд

Эдгээр асуудлуудыг шийдэхийн тулд Vinum нь обьектийн дөрвөн түвшний шатлалыг шийддэг:

- Хамгийн харагддаг обьект нь *volume буюу эзлэхүүн* гэгддэг виртуал диск юм. Эзлэхүүнүүд нь гол нь UNIX® диск хөтчүүдийн нэгэн адил өмчүүдтэй боловч зарим нэг жижиг ялгаанууд байдаг. Тэдэнд хэмжээний хязгаарлалтууд байхгүй.

- Элэхүүнүүд нь тус бүр нь элэхүүний нийт хаягийн талбарыг илэрхийлдэг *plex-үүдээс* бүрддэг. Тиймээс шатлал дахь энэ түвшин нь нөөцийг хангадаг. Plex-үүдийг толин тусгал хийгдсэн массив дахь тус бүртээ ижил өгөгдлийг агуулсан дискнүүд гэж бодох хэрэгтэй.
- Vinum нь UNIX® диск хадгалалтын тогтолцоонд оршдог болохоор UNIX® хуваалтуудыг олон диск бүхий plex-үүдийн хувьд бүтээх блок болгон ашиглах боломжтой, гэхдээ жинхэнэдээ энэ нь хэтэрхий уян хатан биш болж болох юм: UNIX® дискнүүд нь зөвхөн хязгаарлагдмал тооны хуваалтуудтай байдаг. Ингэхийн оронд харин Vinum нь ганц UNIX® хуваалтыг (*хөтөч*) *subdisks буюу дэд дискнүүд* гэгддэг үргэлжлэн нийлсэн талбаруудад хуваадаг бөгөөд эдгээр дэд дискнүүдийг plex-үүдэд зориулж бүтээх блок болгон ашигладаг.
- Дэд дискнүүд нь Vinum *хөтчүүд* дээр, одоогоор UNIX® хуваалтууд дээр оршдог. Vinum хөтчүүд нь дурын тооны дэд дискнүүдийг агуулж болно. Хөтчийн эхлэл дэх тохиргоо болон төлвийн мэдээллийг хадгалах зориулалттай жижиг талбараас бусад хөтчийн бүх хэсэг өгөгдлийн хадгалалтанд зориулагдана.

Дараах хэсгүүд нь эдгээр обектуудын Vinum-ийн шаарддаг ажиллагааг хангах аргыг тайлбарлах болно.

### 22.5.1. Элэхүүний хэмжээний саналууд

Plex-үүд нь Vinum тохиргоонд бүх хөтчүүд дээгүүр тархсан олон дэд дискнүүдийг агуулж болно. Үүний үр дүнд хөтөч бүрийн хэмжээ нь plex-ийн хэмжээг хязгаарлахгүй, тиймээс элэхүүний хэмжээг бас хязгаарлахгүй юм.

### 22.5.2. Өгөгдлийн нөөц хадгалалт

Vinum нь олон plex-үүдийг элэхүүнд хавсран толин тусгал үүсгэлтийг хийж гүйцэтгэдэг. Plex бүр элэхүүн дэх өгөгдлийн дүрслэл юм. Элэхүүн нь нэгээс найман plex-ийг агуулж болно.

Plex нь элэхүүний бүх өгөгдлийг илэрхийлдэг боловч илэрхийллүүдийн хэсгүүд нь хийгдсэн аргаасаа (plex-ийн хэсгүүдийн хувьд дэд дискнүүдийг тодорхойлоогүйгээс) болоод эсвэл тохиолдлоор (хөтөч ажиллагаагүй болсноос болоод) бодитоор байхгүй байх боломжтой юм. Элэхүүний хаягийн бүрэн талбарын хүрээний хувьд хамгийн багаар бодоход нэг plex өгөгдлийг хангаж чаддаг байхад элэхүүн нь бүрэн ажиллагаатай байх юм.

### 22.5.3. Ажиллагааны асуудлууд

Vinum нь нийлүүлэлт болон судалчлалыг plex түвшинд хийж гүйцэтгэдэг:

- *Нийлүүлэгдсэн plex* нь харин дэд диск бүрийн хаягийн талбарыг ашигладаг.
- *Судалчлагдсан plex* нь дэд диск бүрийн дагуу өгөгдлийг судалчилдаг. Дэд диск бүр бүгд адил хэмжээтэй байх ёстой бөгөөд нийлүүлэгдсэн plex-ээс ялгахын тулд хамгийн багаар бодоход хоёр дэд диск байх шаардлагатай.

### 22.5.4. Аль Plex зохион байгуулалт вэ?

FreeBSD 12.0-тэй цуг ирдэг Vinum-ийн хувилбар нь хоёр төрлийн plex-ийг хийж гүйцэтгэдэг:

- Нийлүүлэгдсэн plex-үүд нь хамгийн уян хатан байдаг: тэдгээр нь ямар ч тооны дэд дискнүүдийг агуулж болох бөгөөд дэд дискнүүд нь өөр өөр урттай байж болно. Plex-ийг нэмэлт дэд дискнүүд нэмж өргөтгөж болох юм. Тэдгээр нь судалчлагдсан plex-үүдээс CPU-ийн хугацааны хувьд багыг шаарддаг, гэхдээ CPU дээрх нэмэлт ачааллын ялгаа нь хэмжигдэхээр тийм их биш юм. Нөгөө талаас авч үзэх юм бол тэдгээр нь ганц диск их идэвхитэй бөгөөд бусад нь сул зогсож байдаг халуун цэгүүдэд хамгийн мэдрэмтгий байдаг.
- Судалчлагдсан (RAID-0) plex-үүдийн хамгийн агуу давуу тал нь тэдгээр plex-үүд нь халуун цэгүүдийг багасгадаг явдал юм: хамгийн тохиромжтой (256 kB орчим) хэмжээтэйгээр судлыг сонгосноор та бүрэлдэхүүний хөтчүүд дээрх ачааллыг тэнцүүлж болох юм. Энэ хандлагын сул талууд нь (бутархайгаар хэмжигдэх) илүү төвөгтэй код болон дэд дискнүүд дээрх хязгаарлалтууд юм: тэдгээр нь бүгд ижил хэмжээтэй байх ёстой бөгөөд шинэ дэд дискнүүд нэмж plex-ийг өргөтгөх нь асар хэцүү учир Vinum нь үүнийг одоогоор хийж гүйцэтгэдэггүй юм. Vinum нь нэмэлт, амархан хязгаарлалтыг тохоож өгдөг: судалчлагдсан plex нь хамгийн багаар бодоход хоёр дэд дисктэй байх ёстой, тэгэхгүй бол үүнийг нийлүүлэгдсэн plex-ээс ялгах боломжгүй юм.

Vinum Plex зохион байгуулалтууд нь plex зохион байгуулалт бүрийн давуу болон сул талуудыг дүгнэн харуулж байна.

*Хүснэгт 8. Vinum Plex зохион байгуулалтууд*

| Plex төрөл | Хамгийн бага дэд дискнүүд | Дэд дискнүүд нэмж болох эсэх | Адил хэмжээтэй байх ёстой | Хэрэглээ |
|---|---|---|---|---|
| нийлүүлэгдсэн | 1 | тийм | үгүй | Байрлуулах хамгийн их уян хатан чанар ба боломжийн ажиллагаатай их хэмжээний өгөгдлийн хадгалалт |
| судалчлагдсан | 2 | үгүй | тийм | Маш өндөр зэрэгцээ хандалттай цуг өндөр хурдтай ажиллагаа |

## 22.6. Зарим жишээнүүд

Vinum нь хувь системүүдэд мэдэгддэг обьектуудыг тайлбарладаг *тохиргооны мэдээллийн баазын* ажиллагааг хангаж байдаг. Эхлээд хэрэглэгч нэг буюу эсвэл хэд хэдэн тохиргооны файлуудаас тохиргооны мэдээллийн баазыг diskinfo(8) хэрэгсэл програмын тусламжтайгаар

үүсгэдэг. Vinum нь өөрийн тохиргооны мэдээллийн баазын хуулбарыг диск зүсмэл (Vinum үүнийг *төхөөрөмж* гэдэг) бүр дээр өөрийн хяналтын доор хадгалдаг. Дахин эхлүүлэлт нь зөвөөр Vinum обьект бүрийн төлвийг сэргээдэг байлгахын тулд төлөв өөрчлөгдөх болгонд энэхүү мэдээллийн бааз нь шинэчлэгддэг.

### 22.6.1. Тохиргооны файл

Тохиргооны файл нь Vinum обьектуудыг тайлбарладаг. Энгийн эзлэхүүний тодорхойлолт ийм байж болох юм:

```
drive a device /dev/da3h
volume myvol
  plex org concat
    sd length 512m drive a
```

Энэ файл нь дөрвөн Vinum обьектийг тайлбарладаг:

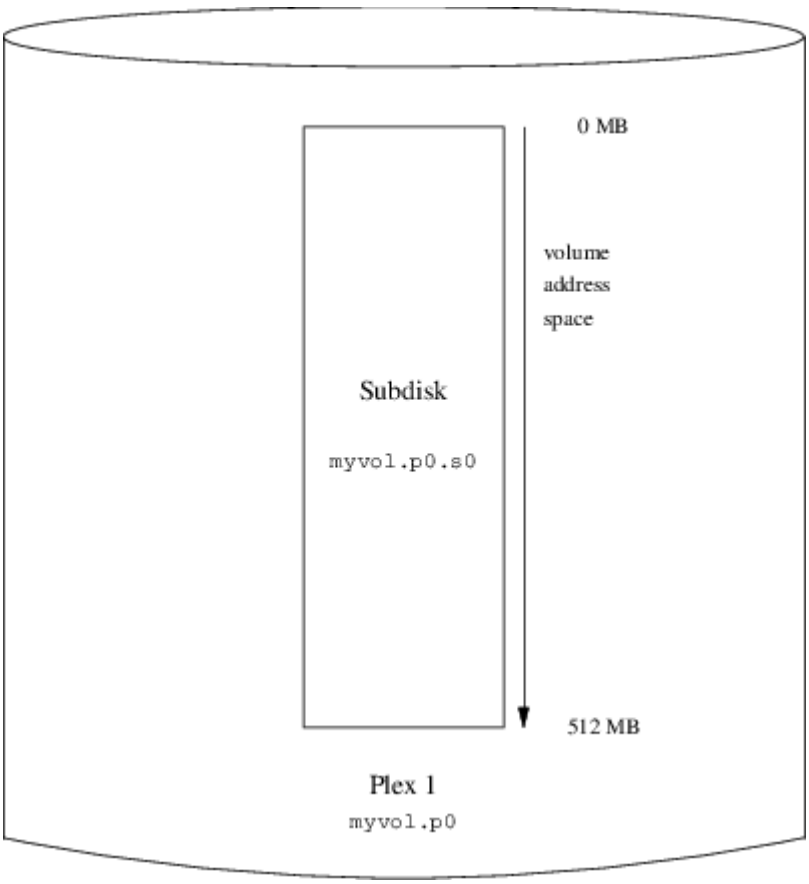
- *drive* мөр нь дискний хуваалтыг (*хөтөч*) тайлбарлах бөгөөд үүний байрлал нь түүн доор орших тоног төхөөрөмжтэй харьцангуй байна. Үүнд *a* гэж симбол нэр өгсөн байгаа. Ингэж симбол нэрсийг төхөөрөмжийн нэрсээс тусад нь салгах нь дискнүүдийг нэг байрлалаас нөгөө уруу эргэлзээгүйгээр шилжүүлэх боломжийг олгодог.
- *volume* мөр нь эзлэхүүнийг тайлбарлана. Цорын ганц шаардлагатай шинж чанар нь нэр бөгөөд энэ тохиолдолд *myvol* байна.
- *plex* мөр нь plex-ийг тодорхойлдог. Цорын ганц шаардлагатай параметр нь зохион байгуулалт бөгөөд энэ тохиолдолд *concat* болно. Ямар ч нэр хэрэггүй: систем автоматаар эзлэхүүний нэрэн дээр *.px* дагавар залгаж үүсгэдэг бөгөөд *x* нь эзлэхүүн дэх plex-ийн дугаар юм. Тиймээс энэ plex нь *myvol.p0* гэж нэрлэгдэнэ.
- *sd* мөр нь дэд дискийг тайлбарлана. Хамгийн бага тодорхойлолтууд нь үүнийг хадгалах хөтчийн нэр болон дэд дискний урт юм. Plex-үүдтэй адил нэр хэрэггүй: систем автоматаар plex-ийн нэрэн дээр *.sx* дагавар залгаж үүсгэдэг бөгөөд *x* нь plex дэх дэд дискний дугаар юм. Тиймээс Vinum нь энэ дэд дискэнд *myvol.p0.s0* гэж нэр өгдөг.

Энэ файлыг процесс хийсний дараа diskinfo(8) дараах үр дүнг гаргана:

```
      #  gvinum -> create config1
      Configuration summary
      Drives:         1 (4 configured)
      Volumes:        1 (4 configured)
      Plexes:         1 (8 configured)
      Subdisks:       1 (16 configured)

    D a                   State: up       Device /dev/da3h        Avail: 2061/2573
MB (80%)

    V myvol               State: up       Plexes:       1 Size:        512 MB

    P myvol.p0           C State: up       Subdisks:     1 Size:        512 MB
```

```
S myvol.p0.s0           State: up       PO:        0  B Size:        512 MB
```

Энэ гаралт нь diskinfo(8)-ийн хэлбэршүүлэлтийн товч жагсаалтыг харуулж байна. Энэ нь графикаар Энгийн Vinum эзлэхүүн зурагт дүрслэгдсэн байна.




*Зураг 105. Энгийн Vinum эзлэхүүн*

Энэ зураг болон дараагийн зураг нь дэд дискнүүд агуулсан plex-үүдийг агуулсан эзлэхүүнийг харуулж байна. Энэ хялбар жишээн дээр эзлэхүүн нь нэг plex-ийг агуулах бөгөөд plex нь нэг дэд дискийг агуулж байна.

Энэхүү эзлэхүүн нь хэвшмэл дискний хуваалттай харьцуулахад ямар нэг тухайлсан давуу чанаргүй юм. Энэ нь ганц plex-ийг агуулах учраас нөөц биш юм. Plex нь ганц дэд дискнээс тогтох учраас хэвшмэл дискний хуваалттай харьцуулахад хадгалалт хуваарилалтын хувьд ямар ч ялгаагүй юм. Дараах хэсгүүд нь төрөл бүрийн илүү сонирхолтой тохиргооны аргуудыг харуулах болно.

### 22.6.2. Нэмэгдсэн уян хатан чанар: Толин тусгал үүсгэх

Эзлэхүүний уян хатан чанарыг толин тусгал үүсгэж ихэсгэж болно. Толин тусгал хийсэн эзлэхүүнийг байрлуулахдаа хөтчийн сүйрэл plex бүрийг унагахгүй байлгахын тулд plex бүр дэх дэд дискнүүд нь өөр өөр хөтчүүд дээр байгаа эсэхийг шалгах нь чухал юм. Дараах тохиргоо нь эзлэхүүнийг толин тусгал хийж байна:

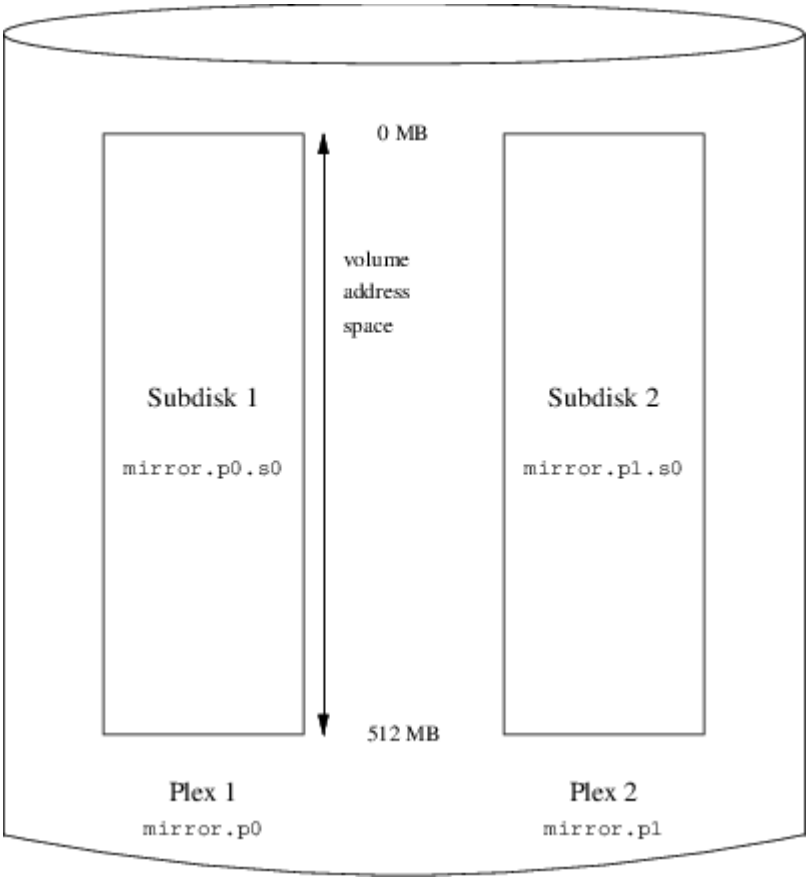
```
drive b device /dev/da4h
volume mirror
```

```
      plex org concat
        sd length 512m drive a
      plex org concat
        sd length 512m drive b
```

Vinum нь өөрийн тохиргооны мэдээллийн бааз дахь бүх обьектуудыг хянаж байдаг болохоор *a* хөтчийн тодорхойлолтыг энэ жишээн дээр дахин зааж өгөх шаардлагагүй юм. Энэ тодорхойлолтыг процесс хийсний дараа тохиргоо иймэрхүү харагдана:

```
    Drives:         2 (4 configured)
    Volumes:        2 (4 configured)
    Plexes:         3 (8 configured)
    Subdisks:       3 (16 configured)

    D a                     State: up       Device /dev/da3h        Avail: 1549/2573
MB (60%)
    D b                     State: up       Device /dev/da4h        Avail: 2061/2573
MB (80%)

    V myvol                 State: up       Plexes:       1 Size:        512 MB
    V mirror                State: up       Plexes:       2 Size:        512 MB

    P myvol.p0            C State: up       Subdisks:     1 Size:        512 MB
    P mirror.p0           C State: up       Subdisks:     1 Size:        512 MB
    P mirror.p1           C State: initializing     Subdisks:     1 Size:        512
MB

    S myvol.p0.s0           State: up       PO:        0  B Size:        512 MB
    S mirror.p0.s0          State: up       PO:        0  B Size:        512 MB
    S mirror.p1.s0          State: empty    PO:        0  B Size:        512 MB
```

Толин тусгал хийгдсэн Vinum эзлэхүүн зураг нь бүтцийг графикаар харуулж байна.




*Зураг 106. Толин тусгал хийгдсэн Vinum эзлэхүүн*

Энэ жишээн дээр plex бүр бүрэн 512 MB хаягийн талбарыг агуулж байна. Өмнөх жишээн дээрх шиг plex бүр зөвхөн ганц дэд дискийг агуулна.

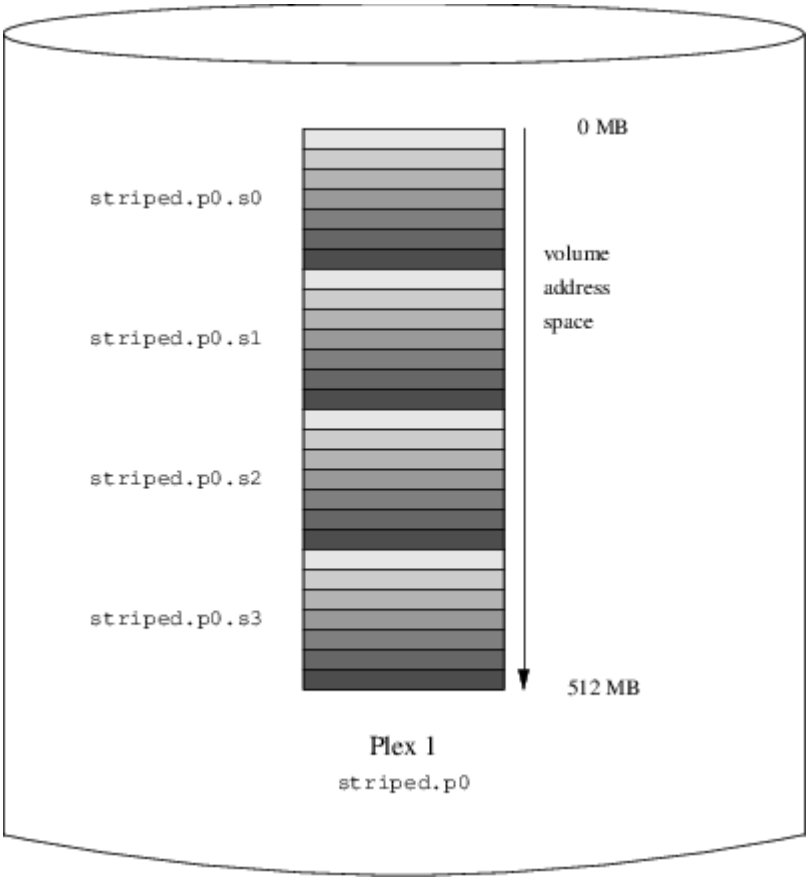
### 22.6.3. Ажиллагааг оновчтой болгох

Өмнөх жишээн дээрх толин тусгал хийгдсэн эзлэхүүн нь толин тусгал хийгдээгүй эзлэхүүнээс илүү сүйрэлд тэсвэртэй боловч ажиллагааны хувьд дутмаг байдаг: эзлэхүүн уруу хийгдэх бичилт бүр нь нийт дискний зурвасын өргөнөөс илүүтэйгээр ашиглан хоёр хөтөч хоёулан уруу нь бичилт хийхийг шаарддаг. Ажиллагааны тооцоонууд нь өөр хандлагыг шаарддаг: толин тусгал үүсгэхийн оронд өгөгдөл нь аль болох олон диск хөтчүүдийн дагуу судалчлагддаг. Дараах тохиргоо нь дөрвөн дискний хөтчүүдийн дагуу судалчлагдсан plex бүхий эзлэхүүнийг харуулж байна:

```
drive c device /dev/da5h
drive d device /dev/da6h
volume stripe
plex org striped 512k
  sd length 128m drive a
  sd length 128m drive b
  sd length 128m drive c
  sd length 128m drive d
```

Өмнөхийн адил Vinum-д мэдэгдэж байгаа хөтчүүдийг тодорхойлох хэрэггүй юм. Энэ тодорхойлолтыг процесс хийсний дараа тохиргоо иймэрхүү байна:

```
    Drives:         4 (4 configured)
    Volumes:        3 (4 configured)
    Plexes:         4 (8 configured)
    Subdisks:       7 (16 configured)

    D a                     State: up       Device /dev/da3h        Avail: 1421/2573
MB (55%)
    D b                     State: up       Device /dev/da4h        Avail: 1933/2573
MB (75%)
    D c                     State: up       Device /dev/da5h        Avail: 2445/2573
MB (95%)
    D d                     State: up       Device /dev/da6h        Avail: 2445/2573
MB (95%)

    V myvol                 State: up       Plexes:       1 Size:        512 MB
    V mirror                State: up       Plexes:       2 Size:        512 MB
    V striped               State: up       Plexes:       1 Size:        512 MB

    P myvol.p0            C State: up       Subdisks:     1 Size:        512 MB
    P mirror.p0           C State: up       Subdisks:     1 Size:        512 MB
    P mirror.p1           C State: initializing     Subdisks:     1 Size:        512
MB
    P striped.p1            State: up       Subdisks:     1 Size:        512 MB

    S myvol.p0.s0           State: up       PO:        0  B Size:        512 MB
    S mirror.p0.s0          State: up       PO:        0  B Size:        512 MB
    S mirror.p1.s0          State: empty    PO:        0  B Size:        512 MB
    S striped.p0.s0         State: up       PO:        0  B Size:        128 MB
    S striped.p0.s1         State: up       PO:      512 kB Size:        128 MB
    S striped.p0.s2         State: up       PO:     1024 kB Size:        128 MB
    S striped.p0.s3         State: up       PO:     1536 kB Size:        128 MB
```




*Зураг 107. Судалчлагдсан Vinum эзлэхүүн*

Энэ эзлэхүүн нь Судалчлагдсан Vinum эзлэхүүн зурагт үзүүлэгдсэн байгаа. Судлуудын харанхуй байдал plex-ийн хаягийн талбар дахь байрлалыг илэрхийлнэ: хамгийн цайвар судлууд нь эхлээд байх бөгөөд хамгийн харанхуйнууд нь сүүлд байна.

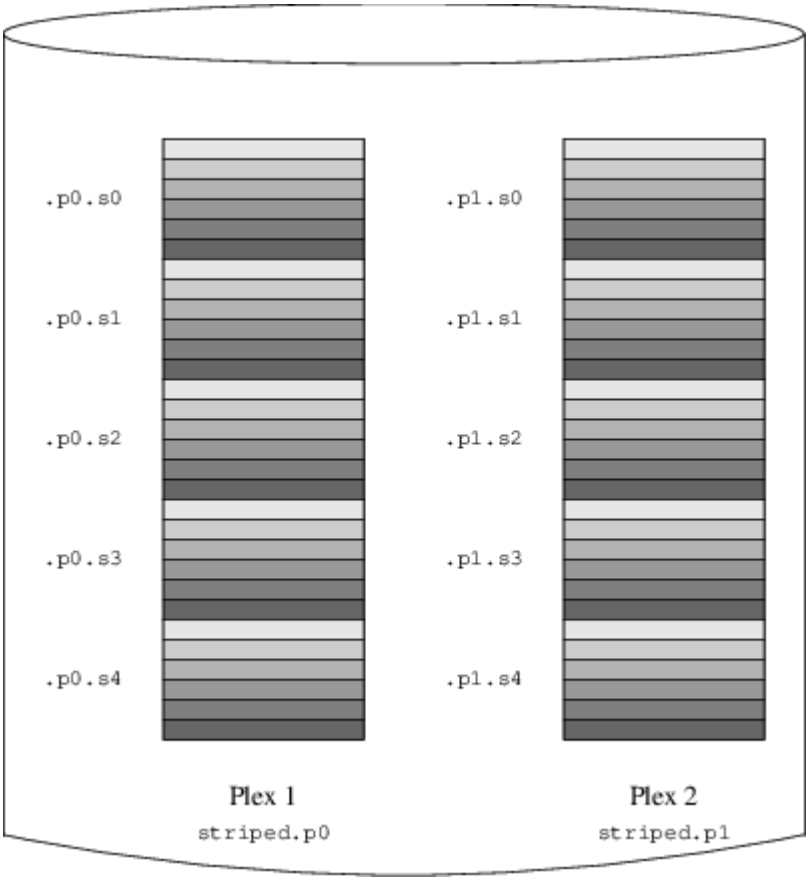
### 22.6.4. Уян хатан чанар ба ажиллагаа

Хангалттай тоног төхөөрөмжийн тусламжтайгаар стандарт UNIX® хуваалтуудтай харьцуулах юм бол нэмэгдсэн уян хатан чанар болон нэмэгдсэн ажиллагаа аль алиныг нь үзүүлдэг эзлэхүүнүүдийг бүтээх боломжтой юм. Нийтлэг тохиргооны файл иймэрхүү байж болох юм:

```
volume raid10
  plex org striped 512k
    sd length 102480k drive a
    sd length 102480k drive b
    sd length 102480k drive c
    sd length 102480k drive d
    sd length 102480k drive e
  plex org striped 512k
    sd length 102480k drive c
    sd length 102480k drive d
    sd length 102480k drive e
    sd length 102480k drive a
    sd length 102480k drive b
```

Хоёр дахь plex-ийн дэд дискнүүд нь эхний plex дэхээс хоёр хөтчөөр ялгаатай: дамжуулалт хоёр хөтчөөр дамжин хийгдсэн ч гэсэн бичилтүүд адил дэд дискнүүд уруугаа хийгдэхгүй байхыг энэ нь баталгаажуулж өгдөг.

Толин тусгал хийгдсэн, судалчлагдсан Vinum эзлэхүүн нь энэ эзлэхүүний бүтцийг илэрхийлж байна.




*Зураг 108. Толин тусгал хийгдсэн, судалчлагдсан Vinum эзлэхүүн*

# 22.7. Обьектийн нэрлэлт

Дээд дурдсаны дагуу Vinum нь анхдагч нэрсийг plex-үүд болон дэд дискнүүдэд өгдөг боловч тэдгээр нь дарагдан өөрчлөгдөж болох юм. Анхдагч нэрсийг дарж өөрчлөхийг зөвлөдөггүй: обьектуудыг дурын нэрээр солихыг зөвшөөрдөг VERITAS эзлэхүүн менежерээс харах юм бол энэ уян хатан чанар нь мэдэгдэхүйц давуу талыг авч ирдэггүй бөгөөд эндүүрэл үүсгэж болох юм.

Нэрс нь дурын хоосон биш тэмдэгтийг агуулж болох бөгөөд тэмдэгтүүдийг үсэг, тоо болон доогуур зураас байлгаж хязгаарлахыг зөвлөдөг. Эзлэхүүнүүд, plex-үүд болон дэд дискнүүд нь 64 тэмдэгт хүртэл урттай байж болох бөгөөд хөтчүүдийн нэрс 32 тэмдэгт хүртэл урттай байж болно.

Vinum обьектууд нь /dev/gvinum шатлалд төхөөрөмжийн цэгүүдийг өгдөг. Дээр үзүүлсэн тохиргоо нь Vinum-ийг дараах төхөөрөмжийн цэгүүдийг үүсгэхэд хүргэнэ:

- Эзлэхүүн бүрийн төхөөрөмжийн оруулгууд. Эдгээр нь Vinum-ийн ашигладаг гол төхөөрөмжүүд юм. Тэгэхээр дээрх тохиргоо нь /dev/gvinum/myvol, /dev/gvinum/mirror, /dev/gvinum/striped, /dev/gvinum/raid5 болон /dev/gvinum/raid10 төхөөрөмжүүдийг агуулах

болно.

- Бүх эзлэхүүнүүд нь /dev/gvinum/ дотор шууд оруулгуудыг авна.
- Plex болон дэд диск бүрийн төхөөрөмжийн цэгүүдийг агуулдаг харгалзах /dev/gvinum/plex болон /dev/gvinum/sd сангууд.

Жишээ нь дараах тохиргооны файл байлаа гэж үзье:

```
drive drive1 device /dev/sd1h
drive drive2 device /dev/sd2h
drive drive3 device /dev/sd3h
drive drive4 device /dev/sd4h
volume s64 setupstate
  plex org striped 64k
    sd length 100m drive drive1
    sd length 100m drive drive2
    sd length 100m drive drive3
    sd length 100m drive drive4
```

Энэ файлыг процесс хийсний дараа diskinfo(8) нь /dev/gvinum файлд дараах бүтцийг үүсгэдэг:

```
drwxr-xr-x  2 root  wheel        512 Apr 13 16:46 plex
crwxr-xr--  1 root  wheel   91,    2 Apr 13 16:46 s64
drwxr-xr-x  2 root  wheel        512 Apr 13 16:46 sd

/dev/vinum/plex:
total 0
crwxr-xr--  1 root  wheel   25, 0x10000002 Apr 13 16:46 s64.p0

/dev/vinum/sd:
total 0
crwxr-xr--  1 root  wheel   91, 0x20000002 Apr 13 16:46 s64.p0.s0
crwxr-xr--  1 root  wheel   91, 0x20100002 Apr 13 16:46 s64.p0.s1
crwxr-xr--  1 root  wheel   91, 0x20200002 Apr 13 16:46 s64.p0.s2
crwxr-xr--  1 root  wheel   91, 0x20300002 Apr 13 16:46 s64.p0.s3
```

Plex-үүд болон дэд дискнүүд нь тусгайлсан нэрсийг хуваарилах ёсгүй гэж зөвлөдөг боловч Vinum хөтчүүд нь нэрлэгдсэн байх ёстой. Энэ нь хөтчийг өөр байрлал уруу зөөж автоматаар таньсан хэвээр байх боломжийг бүрдүүлдэг. Хөтчийн нэрс 32 тэмдэгт хүртэл урттай байж болно.

### 22.7.1. Файлын системүүд үүсгэх нь

Эзлэхүүнүүд нь нэг зүйлээс бусдаараа дискнүүдэд ижил юм шиг системд харагддаг. UNIX® хөтчүүдээс ялгаатай нь Vinum эзлэхүүнүүдийг хуваадаггүй, тиймээс хуваалтын хүснэгтийг агуулдаггүй. Энэ нь зарим нэг дискний хэрэгслүүдэд ялангуяа newfs(8)-д өөрчлөлт хийхийг шаарддаг. Урьд нь newfs(8) Vinum-ийн эзлэхүүний нэрийн сүүлийн үсгийг хуваалт

таниулагч гэж тайлбарлахыг оролддог байсан байна. Жишээ нь дискний хөтөч /dev/ad0a эсвэл /dev/da2h гэсэн нэртэй байж болох юм. Эдгээр нэрс нь эхний (0) IDE диск дээрх эхний хуваалт (a) болон гурав дахь (3) SCSI диск дэх найм дахь хуваалтыг (h) илэрхийлдэг. Ялгаатай тал нь Vinum эзлэхүүн нь хуваалтын нэртэй ямар ч холбоогүй нэрээр магадгүй /dev/gvinum/concat гэж нэрлэгдэж болох юм.

Энэ эзлэхүүн дээр файлын систем үүсгэхийн тулд newfs(8)-г ашиглана:

```
# newfs /dev/gvinum/concat
```

# 22.8. Vinum тохируулах нь

GENERIC цөм Vinum-ийг агуулдаггүй. Vinum оруулсан тусгай цөм бүтээх боломжтой боловч ингэхийг зөвлөдөггүй. Vinum-ийг эхлүүлэх стандарт арга нь цөмийн модуль (kld) юм. Та бүр Vinum-д зориулж kldload(8)-г ч ашиглах хэрэггүй: diskinfo(8)-г эхлүүлэхэд энэ нь модуль ачаалагдсан эсэхийг шалгаад хэрэв үгүй бол түүнийг автоматаар ачаалдаг.

## 22.8.1. Эхлүүлэлт

Vinum нь тохиргооны мэдээллийг гол нь тохиргооны файлуудад байдаг яг тэр хэлбэрээр дискний зүсмэлүүд дээр хадгалдаг. Тохиргооны мэдээллийн баазаас унших үед Vinum тохиргооны файлуудад зөвшөөрөгдөөгүй хэд хэдэн түлхүүр үгнүүдийг таньдаг. Жишээ нь дискний тохиргоо дараах текстийг агуулж болох юм:

```
volume myvol state up
volume bigraid state down
plex name myvol.p0 state up org concat vol myvol
plex name myvol.p1 state up org concat vol myvol
plex name myvol.p2 state init org striped 512b vol myvol
plex name bigraid.p0 state initializing org raid5 512b vol bigraid
sd name myvol.p0.s0 drive a plex myvol.p0 state up len 1048576b driveoffset 265b
plexoffset 0b
sd name myvol.p0.s1 drive b plex myvol.p0 state up len 1048576b driveoffset 265b
plexoffset 1048576b
sd name myvol.p1.s0 drive c plex myvol.p1 state up len 1048576b driveoffset 265b
plexoffset 0b
sd name myvol.p1.s1 drive d plex myvol.p1 state up len 1048576b driveoffset 265b
plexoffset 1048576b
sd name myvol.p2.s0 drive a plex myvol.p2 state init len 524288b driveoffset 1048841b
plexoffset 0b
sd name myvol.p2.s1 drive b plex myvol.p2 state init len 524288b driveoffset 1048841b
plexoffset 524288b
sd name myvol.p2.s2 drive c plex myvol.p2 state init len 524288b driveoffset 1048841b
plexoffset 1048576b
sd name myvol.p2.s3 drive d plex myvol.p2 state init len 524288b driveoffset 1048841b
plexoffset 1572864b
sd name bigraid.p0.s0 drive a plex bigraid.p0 state initializing len 4194304b driveoff
set 1573129b plexoffset 0b
```

```
sd name bigraid.p0.s1 drive b plex bigraid.p0 state initializing len 4194304b driveoff
set 1573129b plexoffset 4194304b
sd name bigraid.p0.s2 drive c plex bigraid.p0 state initializing len 4194304b driveoff
set 1573129b plexoffset 8388608b
sd name bigraid.p0.s3 drive d plex bigraid.p0 state initializing len 4194304b driveoff
set 1573129b plexoffset 12582912b
sd name bigraid.p0.s4 drive e plex bigraid.p0 state initializing len 4194304b driveoff
set 1573129b plexoffset 16777216b
```

Эндэх мэдээжийн ялгаанууд нь байрлалын тодорхой мэдээлэл болон нэрлэлт (тэдгээрийн хоёул бас зөвшөөрөгдсөн боловч хэрэглэгчид ашиглахыг хориглосон) болон түвшингүүд дэх мэдээлэл (хэрэглэгчид байдаггүй) байгаа явдал юм. Vinum нь тохиргооны мэдээлэл дэх хөтчүүдийн тухай мэдээллийг хадгалдаггүй: энэ нь Vinum шошготой хуваалтуудын хувьд тохируулагдсан дискний хөтчүүдийг скан хийн хөтчүүдийг олдог. Энэ нь тэдгээр хөтчүүдэд өөр төрлийн UNIX® хөтчийн ID өгсөн байсан ч Vinum-д хөтчүүдийг зөвөөр таних боломжийг олгодог.

#### 22.8.1.1. Автомат эхлүүлэлт

*Gvinum* нь loader.conf(5)-ийн тусламжтай цөмийн модуль ачаалагдсаны дараа автоматаар эхлүүлэгддэг боломжийг үргэлж агуулсан байна. *Gvinum* модулийг ачаалалтын үед дуудахын тулд /boot/loader.conf файлд `geom_vinum_load="YES"` мөрийг нэмэх хэрэгтэй.

`gvinum start` тушаалаар Vinum-ийг эхлүүлэхэд Vinum нь аль нэг Vinum хөтчөөс тохиргооны мэдээллийн баазыг уншдаг. Ердийн тохиолдолд хөтөч бүр тохиргооны мэдээллийн баазын адил хуулбарыг агуулдаг учир аль хөтөч уншигдах нь хамаагүй юм. Гэхдээ сүйрлийн дараа Vinum нь аль хөтөч хамгийн сүүлд шинэчлэгдсэнийг тодорхойлж тэр хөтчөөс тохиргоог унших ёстой. Дараа нь тэр хэрэв шаардлагатай бол хуучин хөтчүүдээс тохиргоог шинэчилдэг.

## 22.9. Root файлын системийн хувьд Vinum-ийг ашиглах нь

Vinum ашиглан бүрэн толин тусгал хийгдсэн файлын системүүд бүхий машины хувьд root файлын системийг бас толин тусгал хийх нь зүйтэй. Ингэж тохируулах нь дурын файлын системийг толин тусгал хийхээс амаргүй юм, учир нь:

- root файлын систем нь ачаалах процессийн нэлээн эхэн үед бэлэн байж байх шаардлагатай учраас Vinum дэд бүтэц нь энэ үед аль хэдийн бэлэн болсон байх шаардлагатай.
- root файлын системийг агуулсан эзлэхүүн нь бас системийг эхлүүлэх код болон цөмийг агуулдаг бөгөөд цөм нь хост системийн эх хэрэгслүүд (өөрөөр хэлбэл PC ангиллын машинууд дээр BIOS) ашиглан уншигдах ёстой. Эдгээр хост системийн эх хэрэгслүүдэд Vinum-ийн талаар дэлгэрэнгүйг мэдүүлэх ихэвчлэн боломжгүй байдаг.

Дараах хэсгүүдэд "root эзлэхүүн" гэсэн ойлголт root файлын системийг агуулдаг Vinum эзлэхүүнийг тайлбарлахад голчлон ашиглагдана. Энэ эзлэхүүний хувьд `"root"` гэсэн нэрийг

ашиглах нь зүйтэй санаа боловч энэ нь аль ч талаараа заавал шаардлагатай биш юм. Дараах хэсгүүд дэх бүх тушаалын жишээнүүд нь энэ нэрийг ашиглах болно.

### 22.9.1. root файлын системийн хувьд Vinum-ийг аль болох эрт эхлүүлэх нь

Ингэхийн тулд хэд хэдэн арга хэмжээнүүдийг авах хэрэгтэй:

- Vinum нь ачаалах үед цөмд бэлэн байх ёстой. Тиймээс Автомат эхлүүлэлт-д тайлбарласан Vinum-ийг автоматаар эхлүүлэх арга нь энэ асуудлыг шийдэхэд хэрэглэх боломжгүй бөгөөд `start_vinum` параметрийг дараах тохиргоог зохион байгуулсан тохиолдолд тохируулах ёсгүй. Эхний тохируулга нь Vinum-ийг статикаар цөмд эмхэтгэх тохируулга байх юм. Тэгэхээр энэ боломж нь үргэлж бэлэн байх боловч ихэнх тохиолдолд хүсмээр зүйл биш байдаг. /boot/loader (Гуравдугаар шат /boot/loader)-аар цөмийг эхлэхээс өмнө vinum цөмийн модулийг эрт ачаалахаар тохируулах өөр бас нэг тохируулга байдаг. Ингэхийн тулд дараах мөрийг:

  ```
  geom_vinum_load="YES"
  ```

  /boot/loader.conf файл уруу оруулж хийж болно.

- *Gvinum*-ийн хувьд бүх эхлүүлэлт нь цөмийн модуль ачаалагдсаны дараа автоматаар хийгддэг учраас дээр тайлбарласан процедур байхад л хангалттай.

### 22.9.2. Vinum дээр тулгуурласан root эзлэхүүнийг эхлүүлэх кодонд хандах боломжтой болгох нь

Одоогийн FreeBSD эхлүүлэгч нь зөвхөн 7.5KB код байдаг бөгөөд энэ нь UFS файлын системээс файлуудыг (/boot/loader зэрэг) уншдагаар хийгдсэн байдаг болохоор Vinum-ийн тохиргооны мэдээллийг боловсруулж эхлэх эзлэхүүний өөрийнх нь элементүүдийг таньж Vinum-ийн дотоод бүтцүүдийг түүнд мэдүүлэх нь илт боломжгүй юм. Тиймээс root файлын системийг агуулдаг стандарт `"a"` хуваалтын хуурамч үзэгдлээр эхлүүлэгч кодыг хангах зарим нэгэн заль мэх шаардлагатай байдаг.

Ингэхийн тулд root эзлэхүүний хувьд дараах шаардлагуудыг хангасан байх ёстой:

- root эзлэхүүн нь судал хийгдээгүй эсвэл RAID-5 хийгдээгүй байх ёстой.
- root эзлэхүүн нь нэг plex-ийн хувьд нэгээс илүү нийлүүлэгдсэн дэд дисктэй байх ёсгүй.

Тус бүрдээ root файлын системийн нэг хуулбарыг агуулсан олон plex байлгах боломжтой бөгөөд ингэх сонирхолтой байдгийг санаарай. Гэхдээ эхлүүлэгч процесс нь эдгээр хуулбаруудын зөвхөн нэгийг эхлүүлэгч болон бүх файлуудыг олохын тулд цөм root файлын системийг өөрийг нь холбох хүртэл хэрэглэдэг. Дараа нь харгалзах төхөөрөмжийг эхлүүлэх боломжтой болгохын тулд эдгээр plex-үүдийн доторх ганц дэд диск бүрт өөрсдийн `"a"` хуваалтын хуурамч үзэгдэл хэрэгтэй болдог. Эдгээр хуурсан `"a"` хуваалт бүр өөрийн төхөөрөмжийн дотор адил офсетэд байрлах нь root эзлэхүүний plex-үүдийг агуулдаг бусад төхөөрөмжүүдтэй харьцуулах юм бол заавал хэрэгтэй байдаггүй. Гэхдээ Vinum эзлэхүүнүүдийг толин тусгал хийгдсэн төхөөрөмжүүд нь тэгш хэмт байхаар эндүүрлээс

хамгаалж үүсгэх нь магадгүй зүйтэй санаа байх юм.

Эдгээр "a" хуваалтуудыг root эзлэхүүний хэсгийг агуулсан төхөөрөмж бүрийн хувьд тохируулахын тулд доор дурдсаныг хийх хэрэгтэй:

1. Энэ төхөөрөмжийн root эзлэхүүний хэсэг болох дэд дискний байрлал (төхөөрөмжийн эхэн хэсгээс тодорхой офсетэд) болон хэмжээг дараах тушаалыг ашиглаж шалгах хэрэгтэй:

   ```
   # gvinum l -rv root
   ```

   Vinum офсетүүд болон хэмжээнүүд нь байтаар хэмжигддэг болохыг санаарай. bsdlabel тушаалд хэрэглэгддэг блок дугааруудыг олж авахын тулд тэдгээр нь 512-т хуваагдах ёстой.

2. Дараах тушаалыг:

   ```
   # bsdlabel -e devname
   ```

   гэж root эзлэхүүнд оролцож байгаа төхөөрөмж бүрийн хувьд ажиллуулна. *devname* нь зүсмэлийн хүснэгттүй дискнүүдийн дискний нэр (da0 гэх зэрэг) эсвэл зүсмэлийн нэр (ad0s1 гэх зэрэг) байх ёстой.

   Хэрэв төхөөрөмж дээр "a" хуваалт (Vinum-ийн өмнөх root файлын системийг агуулсан) байх юм бол энэ нь хандахад боломжтой хэвээр (зөвхөн энэ тохиолдолд) байхын тулд өөр нэрээр нэрлэгдэх шаардлагатай. Гэхдээ анхдагчаар системийг эхлүүлэхэд хэрэглэгдэхээ болих юм. Идэвхтэй хуваалтуудын (одоогоор холбогдсон root файлын систем зэрэг) нэрийг өөрчилж болохгүй, тийм болохоор үүнийг "Fixit" зөөвөрлөгчөөс эхлүүлэх үед эсвэл (толин тусгал хийгдсэн нөхцөлд) ачаалаагүй байгаа дискийг эхлээд удирдах хоёр алхамтай процессийн үед ажиллуулах ёстойг санаарай.

   Дараа нь энэ төхөөрөмж дээрх Vinum хуваалтын офсет нь энэ төхөөрөмжийн харгалзах root эзлэхүүний дэд дискний офсетэд нэмэгдэх ёстой. Үр дүнд нь гарах утга нь шинэ "a" хуваалтын хувьд "offset" буюу офсет утга болно. Энэ хуваалтын хувьд "size" буюу хэмжээ уттыг дээрх тооцооллоос үгчлэн авч болно. "fstype" нь 4.2BSD байх ёстой. "fsize", "bsize", болон "cpg" уттууд нь байгаа файлын системтэй таарахаар сонгогдсон байх хэрэгтэй, гэхдээ тэдгээр нь энэ сэдэвт бараг л чухал биш юм.

   Ийм замаар шинэ "a" хуваалт нь энэ төхөөрөмж дээрх Vinum хуваалтыг дарж байгуулагдана. Хэрэв Vinum хуваалт нь "vinum" fstype-г ашиглан зөвөөр тэмдэглэгдсэн бол bsdlabel тушаал нь зөвхөн энэ даралтыг зөвшөөрөх болно.

3. Ингээд боллоо! root эзлэхүүний нэг хуулбартай төхөөрөмж бүр дээр хуурсан "a" хуваалт одоо бий боллоо. Үр дүнг дараах тушаалтай төстэй тушаал ашиглан дахин шалгахыг зөвлөдөг:

```
# fsck -n /dev/devnamea
```

Шинэ Vinum root эзлэхүүнийг тохируулах үед идэвхтэй байгаа root файлын системтэй таардаггүй учир хяналтын мэдээлэл агуулсан бүх файлууд нь Vinum эзлэхүүн дэх root файлын системд харьцангуй байх ёстойг санах хэрэгтэй. Тэгэхээр ялангуяа /etc/fstab болон /boot/loader.conf файлуудыг бодолцох хэрэгтэй юм.

Дараагийн дахин ачаалалтын дараа эхлүүлэгч нь шинэ Vinum дээр тулгуурласан root файлын системээс тохирох хяналтын мэдээллийг олж түүний дагуу ажиллах ёстой. Цөмийн эхлүүлэх процессийн төгсгөлд бүх төхөөрөмжүүд зарлагдсаны дараа энэ тохиргоог амжилттай болсныг харуулсан алдартай мэдэгдэл иймэрхүү байдлаар харуулагдана:

```
Mounting root from ufs:/dev/gvinum/root
```

### 22.9.3. Vinum дээр тулгуурласан root тохиргооны жишээ

Vinum root эзлэхүүн тохируулагдсаны дараа `gvinum l -rv root` тушаалын үр дүн иймэрхүү харагдана:

```
...
Subdisk root.p0.s0:
        Size:        125829120 bytes (120 MB)
        State: up
        Plex root.p0 at offset 0 (0  B)
        Drive disk0 (/dev/da0h) at offset 135680 (132 kB)

Subdisk root.p1.s0:
        Size:        125829120 bytes (120 MB)
        State: up
        Plex root.p1 at offset 0 (0  B)
        Drive disk1 (/dev/da1h) at offset 135680 (132 kB)
```

Тэмдэглэх ёстой утгууд нь офсетийн (/dev/da0h хуваалтын хувьд харьцангуй байна) хувьд `135680` байна. Энэ нь `bsdlabel`-ийн ойлголтод 265 ширхэг 512 байт диск блок уруу хөрвүүлнэ. Үүнтэй адилаар энэ root эзлэхүүний хэмжээ нь 245760 512 байт блок юм. Энэ root эзлэхүүний хоёр дахь хуулбарыг агуулсан /dev/da1h нь тэгш хэмтэй тохиргоотой байна.

Эдгээр төхөөрөмжийн хувьд bsdlabel нь иймэрхүү харагдана:

```
...
8 partitions:
#        size   offset    fstype   [fsize bsize bps/cpg]
  a:   245760      281    4.2BSD     2048 16384     0   # (Cyl.    0*- 15*)
  c: 71771688        0    unused        0     0         # (Cyl.    0 - 4467*)
```

```
  h: 71771672       16     vinum                    # (Cyl.    0*- 4467*)
```

Хуурсан "a" хуваалтын "size" параметр нь дээр харуулсан утгатай таардаг бол харин "offset" параметр нь Vinum хуваалт "h" доторх офсет болон төхөөрөмж (эсвэл зүсмэл) дэх энэ хуваалтын офсетийн нийлбэр болохыг эндээс харж болно. Энэ нь Юу ч ачаалахгүй болж эхлүүлэгч сүйрэх-д тайлбарласан асуудлыг тойрон гарахад шаардлагатай ердийн тохиргоо юм. Мөн бүхэл "a" хуваалт нь энэ төхөөрөмжийн хувьд бүх л Vinum өгөгдлийг агуулан бүтнээрээ "h" дотор байхыг эндээс бас харж болно.

Дээрх жишээн дээр бүхэл төхөөрөмж нь Vinum-д зориулагдсан бөгөөд зөвхөн Vinum тохиргооны хэсэг болохоор зориулагдсан дискний шинэ тохиргоо учраас Vinum-аас өмнөх root хуваалтын үлдэгдэл байхгүй болохыг санах хэрэгтэй.

## 22.9.4. Алдааг олж засварлах нь

Хэрэв ямар нэгэн зүйл болохоо байвал энэ тохиолдолд сэргээх арга хэрэгтэй. Дараах жагсаалт нь тохиолддог цөөн хэдэн асуудлууд болон шийдлүүдийг харуулж байна.

### 22.9.4.1. Системийн эхлүүлэгч ачаалах боловч систем эхлэхгүй байх

Ямар нэг шалтгаанаар систем нь ачаалахгүй бол эхлүүлэгчийг 10 секундын анхааруулга дээр `space` товчлуур дарж тасалдуулж болно. Дуудагчийн хувьсагчуудыг (`vinum.autostart` зэрэг) `show` тушаал ашиглан шалгаж болох бөгөөд `set` эсвэл `unset` тушаалуудыг ашиглан тохируулж болно.

Хэрэв цөмийн Vinum модуль нь автоматаар дуудагддаг модулиудын жагсаалтад орж амжаагүй байх юм бол ердийн `load geom_vinum` тушаал туслана.

Бэлэн болсныхоо дараа ачаалах процессийг `boot -as` тушаалаар үргэлжлүүлж болно. `-as` тохируулгууд нь root файлын системийг холбохыг (`-a`) асуудаг бөгөөд ачаалах процессийг root файлын системийг зөвхөн унших эрхээр холбосон ганц-хэрэглэгчийн горимд (`-s`) зогсоохыг цөмөөс хүсдэг. Ийм замаар олон-plex эзлэхүүний зөвхөн нэг plex холбогдсон байхад plex-үүд хоорондын өгөгдлийн бүрэн бүтэн бус байдлын эрсдэл гардаггүй.

root файлын системийг холбохыг асуух мөрөн дээр зөв root файлын системтэй дурын төхөөрөмжийг оруулж болно. Хэрэв /etc/fstab зөв тохируулагдсан бол анхдагч нь `ufs:/dev/gvinum/root` мөртэй төсөөтэй байх ёстой. Ердийн өөр нэг сонголт нь `ufs:da0d` зэрэг Vinum-аас өмнөх root файлын системийг агуулдаг таамаглал хуваалт байж болох юм. Энд жинхэнэдээ толин тусгал үүсгэгдсэн тохируулгаас болоод Vinum root төхөөрөмжийн дэд дискнүүд уруу заасан аль нэг хуулбар "a" хуваалтыг оруулсан бол болгоомжтой байх хэрэгтэй бөгөөд энэ нь толин тусгал хийгдсэн root төхөөрөмжийн зөвхөн нэгийг холбох болно. Хэрэв энэ файлын систем нь дараа нь унших бичих эрхтэйгээр холбогдох бол Vinum root эзлэхүүний бусад plex-(үүд)ийг арилгах шаардлагатай. Учир нь эдгээр plex-үүд нь бүрэн бүтэн бус өгөгдлийг зөөвөрлөх болно.

### 22.9.4.2. Зөвхөн анхдагч эхлүүлэгч дуудах

Хэрэв /boot/loader дуудагдахгүй байгаа боловч анхдагч эхлүүлэгч дуудагдаж байвал (ачаалах процесс эхэлсний дараа дэлгэцийн зүүн багана дээр ганц зураас харагддаг) энэ үед анхдагч

эхлүүлэгчийг тасалдуулах оролдлогыг `space` товчлуур ашиглан хийж болно. Энэ нь эхлүүлэгчийг хоёрдугаар алхам дээр зогсоох бөгөөд Нэгдүгээр шат /boot/boot1 болон Хоёрдугаар шат /boot/boot2-с үзнэ үү. Энд `"a"`-с дээр шилжүүлэгдсэн өмнөх root файлын системийг агуулсан өөр хуваалтаас ачаалах оролдлого хийж болох юм.

#### 22.9.4.3. Юу ч ачаалахгүй болж эхлүүлэгч сүйрэх

Энэ тохиолдол нь хэрэв эхлүүлэгчийг Vinum суулгацаар устгасан бол тохиолддог. Харамсалтай нь одоогоор Vinum нь санамсаргүйгээр зөвхөн 4 KB хэмжээг чөлөөтэйгөөр өөрийн хуваалтын эхэнд өөрийн Vinum толгойн мэдээллийг бичиж эхлэхээсээ өмнө үлдээдэг. Гэхдээ нэгдүгээр алхам ба хоёрдугаар эхлүүлэгчид, тэдгээрийн хооронд суулгагдсан bsdlabel нь одоогоор 8 KB-г шаарддаг. Хэрэв Vinum хуваалт нь ачаалагдах боломжтой гэгдэж байгаа зүсмэл эсвэл диск дотор 0 офсетэд эхэлсэн бол Vinum нь эхлүүлэгчийг устгана.

Үүнтэй адилаар хэрэв дээрх тохиолдол сэргээгдсэн бол, жишээ нь "Fixit" зөөвөрлөгчөөс ачаалагдаж эхлүүлэгч нь Нэгдүгээр шат /boot/boot1 болон Хоёрдугаар шат /boot/boot2-д тайлбарласны дагуу дахин суулгагдсан бол эхлүүлэгч нь Vinum толгойг устгах бөгөөд Vinum нь өөрийн диск(үүд)ийг дахиж олохгүй байх болно. Яг жинхэнэдээ Vinum-ийн тохиргооны өгөгдөл эсвэл Vinum-ийн эзлэхүүнүүд дэх өгөгдөл нь энэ үед устгагддаггүй бөгөөд бүх өгөгдлийг яг адилхан Vinum тохиргооны өгөгдлийг дахин оруулснаар сэрээх боломжтой боловч ийм тохиолдлыг засварлах нь их хэцүү байдаг. Vinum толгой болон системийн эхлүүлэгчийг давхцуулахгүй байлгахын тулд бүхэл Vinum хуваалтыг хамгийн багаар бодоход 4 KB-аар шилжүүлэх хэрэгтэй болно.

# Chapter 23. Virtualisation

## 23.1. Ерөнхий агуулга

Виртуалчлалын програм хангамж нь олон үйлдлийн системийг нэг компьютер дээр ажиллуулах боломжийг олгодог. PC-үүдэд зориулсан тийм програм хангамжууд нь виртуалчлалын програм хангамжийг ажиллуулж зочин үйлдлийн систем хэд л бол хэдийг дэмжих эх үйлдлийн систем байдаг.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Хост буюу эх үйлдлийн систем болон зочин үйлдлийн системийн ялгаа.
- FreeBSD-г Intel® дээр суурилсан Apple® Macintosh® компьютер дээр хэрхэн суулгах талаар.
- FreeBSD-г Microsoft® Windows® дээр Virtual PC ашиглан хэрхэн суулгах талаар.
- Виртуалчлалын доор FreeBSD системийг хамгийн сайн ажиллагаатайгаар хэрхэн тааруулах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- UNIX® болон FreeBSD-ийн үндсүүдийг ойлгох.
- FreeBSD-г хэрхэн суулгах талаар мэдэх.
- Сүлжээний холболтоо хэрхэн тохируулах талаар мэдэх.
- Нэмэлт гуравдагч програм хангамжуудыг хэрхэн суулгах талаар мэдэх.

## 23.2. FreeBSD-г зочин OS маягаар ашиглах

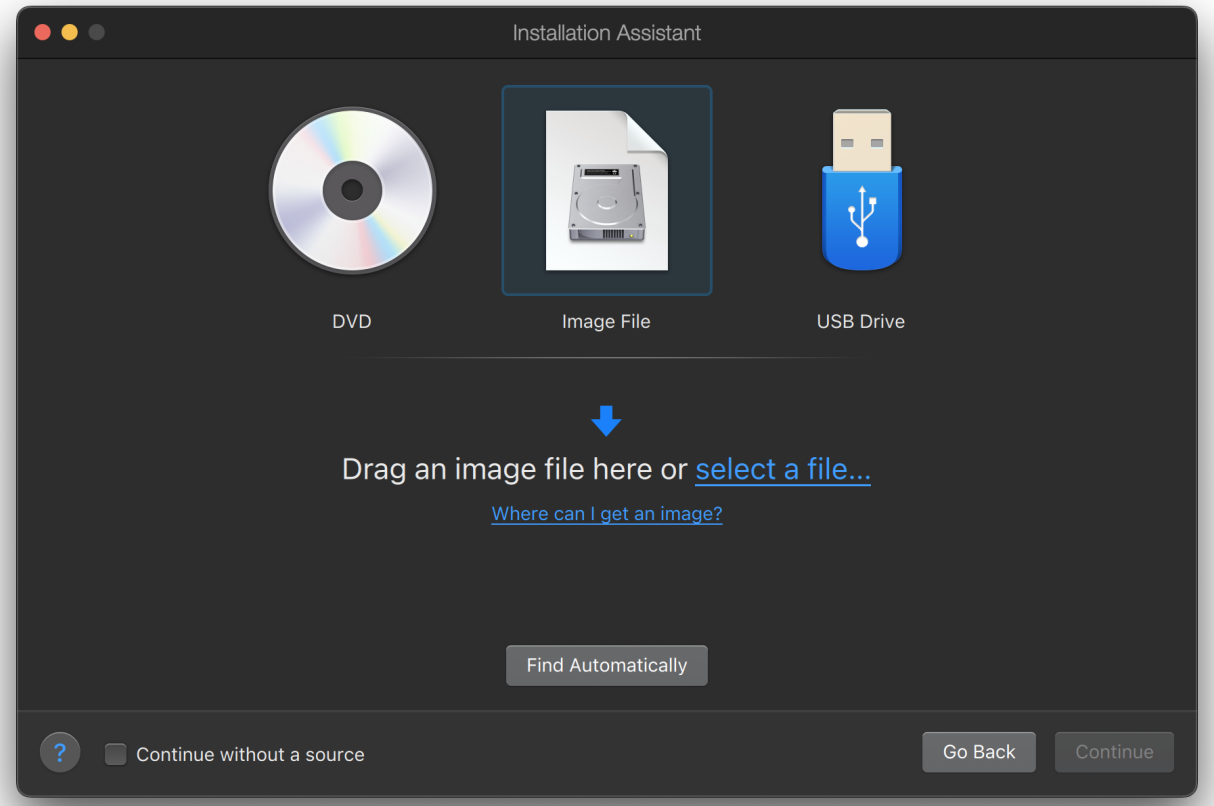
### 23.2.1. Mac OS® X дээрх Parallels

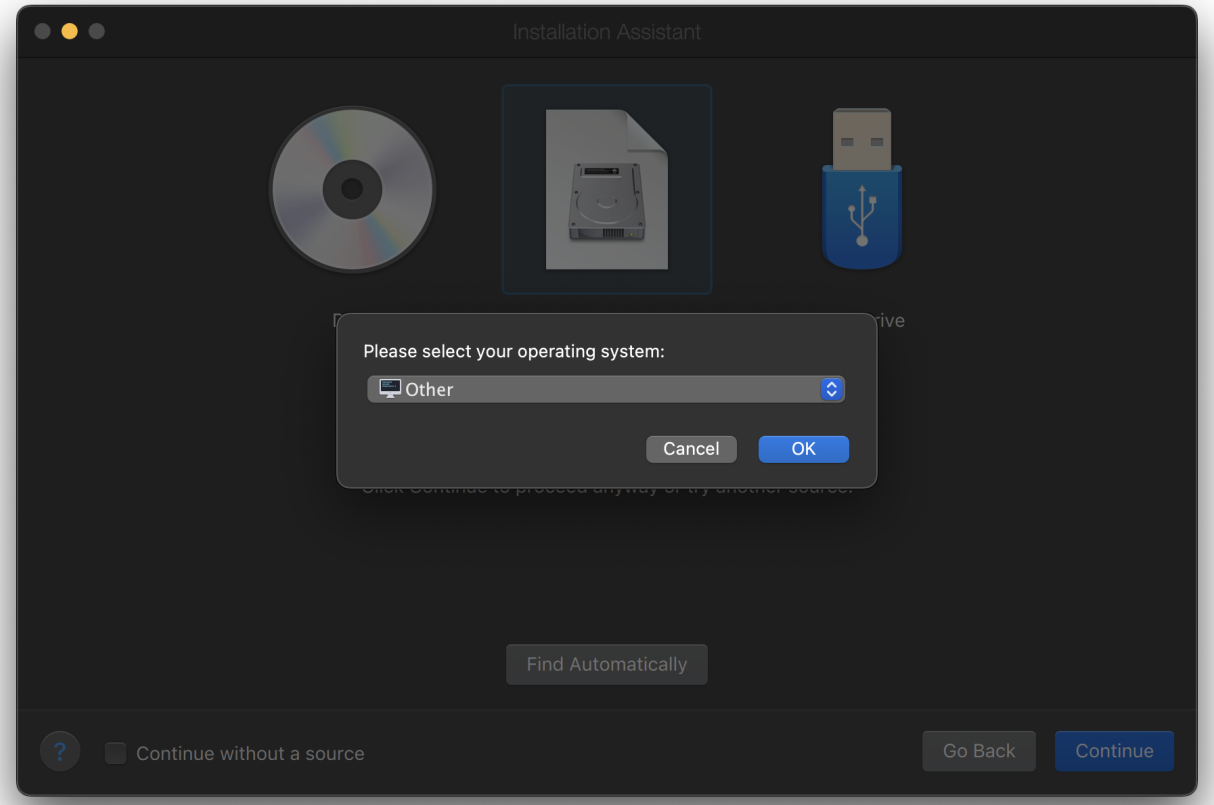
Mac®-д зориулсан Parallels Desktop нь Mac OS® 10.4.6 буюу түүнээс дээш хувилбарыг ашиглаж байгаа Intel® дээр суурилсан Apple® Mac® компьютеруудад зориулагдсан арилжааны програм хангамжийн бүтээгдэхүүн юм. FreeBSD нь бүрэн дэмжигдсэн зочин үйлдлийн систем юм. Mac OS® X дээр Parallels суулгагдсаны дараа хэрэглэгч виртуал машинаа тохируулж дараа нь хүсэж байгаа зочин үйлдлийн системээ суулгах ёстой.

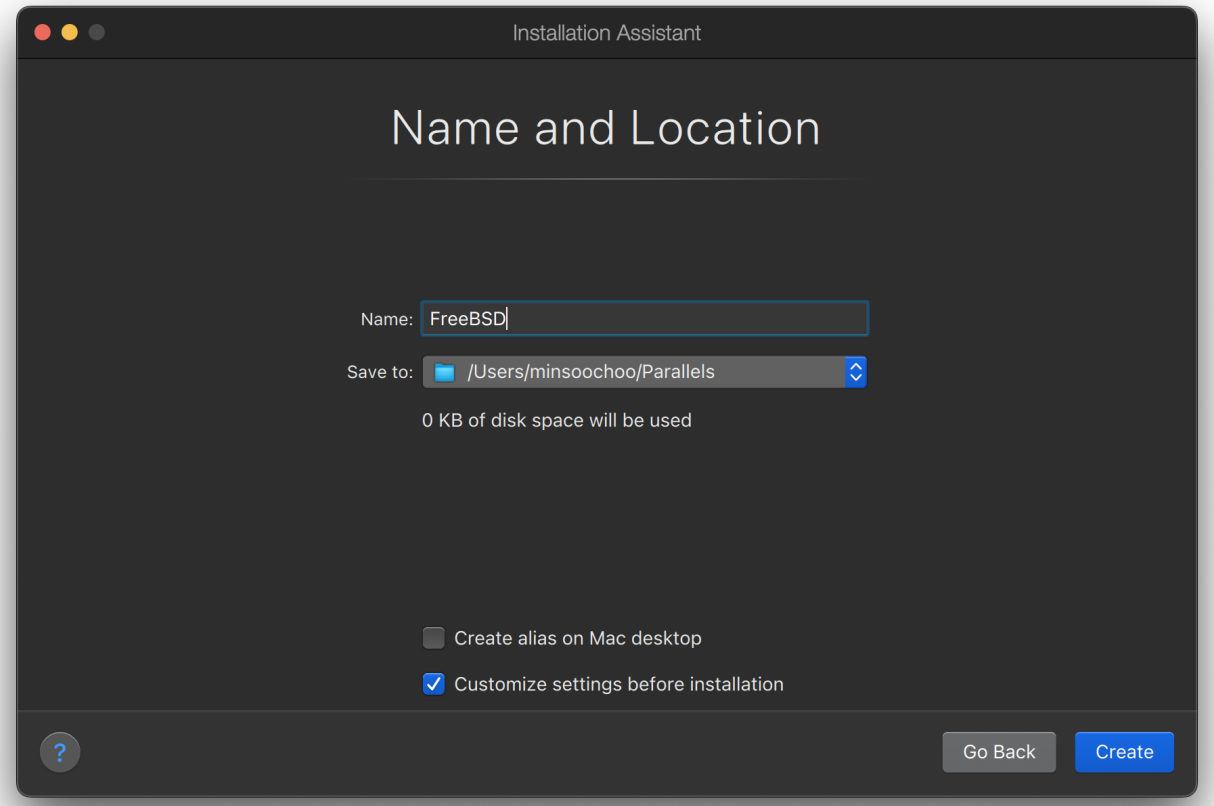
#### 23.2.1.1. Parallels/Mac OS® X дээр FreeBSD-г суулгах

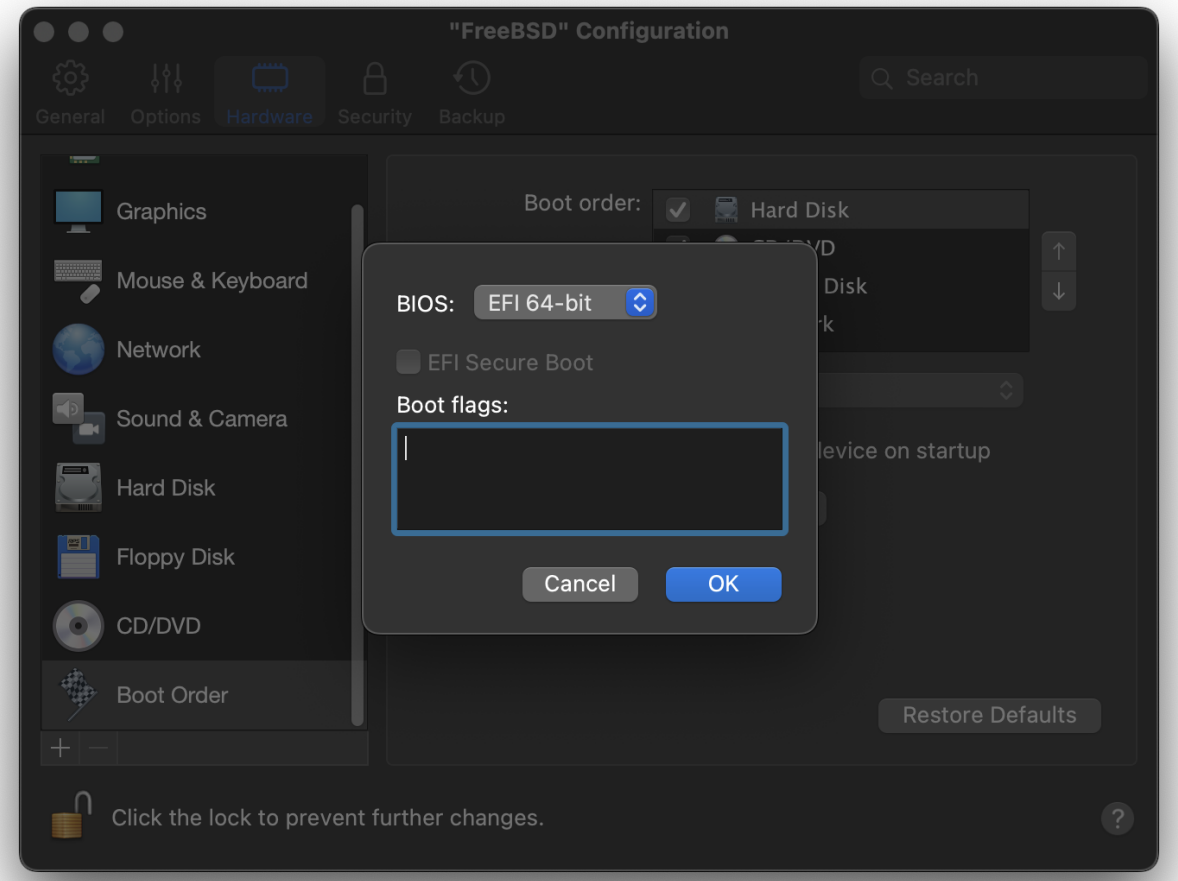
FreeBSD-г Parallels дээр суулгах эхний алхам нь FreeBSD-г суулгахад зориулж шинэ виртуал машиныг үүсгэх явдал юм. **Guest OS Type** буюу Зочин OS-ийн төрөл гэж асуухад нь FreeBSD гэж сонгох хэрэгтэй:

Тэгээд энэ виртуал FreeBSD машины хувьд өөрийн төлөвлөгөөнөөс хамаараад боломжийн хэмжээний диск болон санах ойг сонгох хэрэгтэй. Parallels дээр FreeBSD-г ашиглах ихэнх хэрэглээний хувьд 4GB дискний зай ба 512MB хэмжээний RAM сайн ажилладаг.

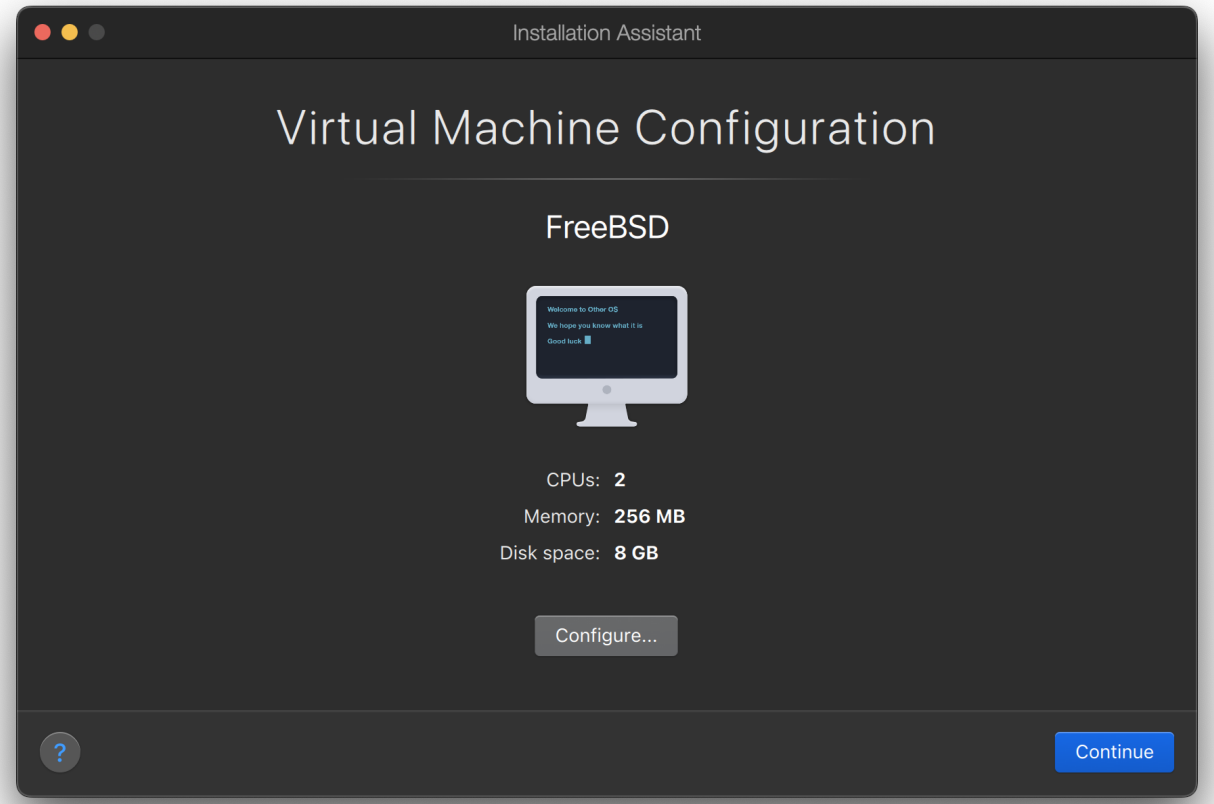

Installation Assistant
DVD
Image File
USB Drive
Drag an image file here or select a file...
Where can I get an image?
Find Automatically
Continue without a source
Go Back
Continue


Installation Assistant
Please select your operating system:
Other
Cancel
OK
Find Automatically
Continue without a source
Go Back
Continue

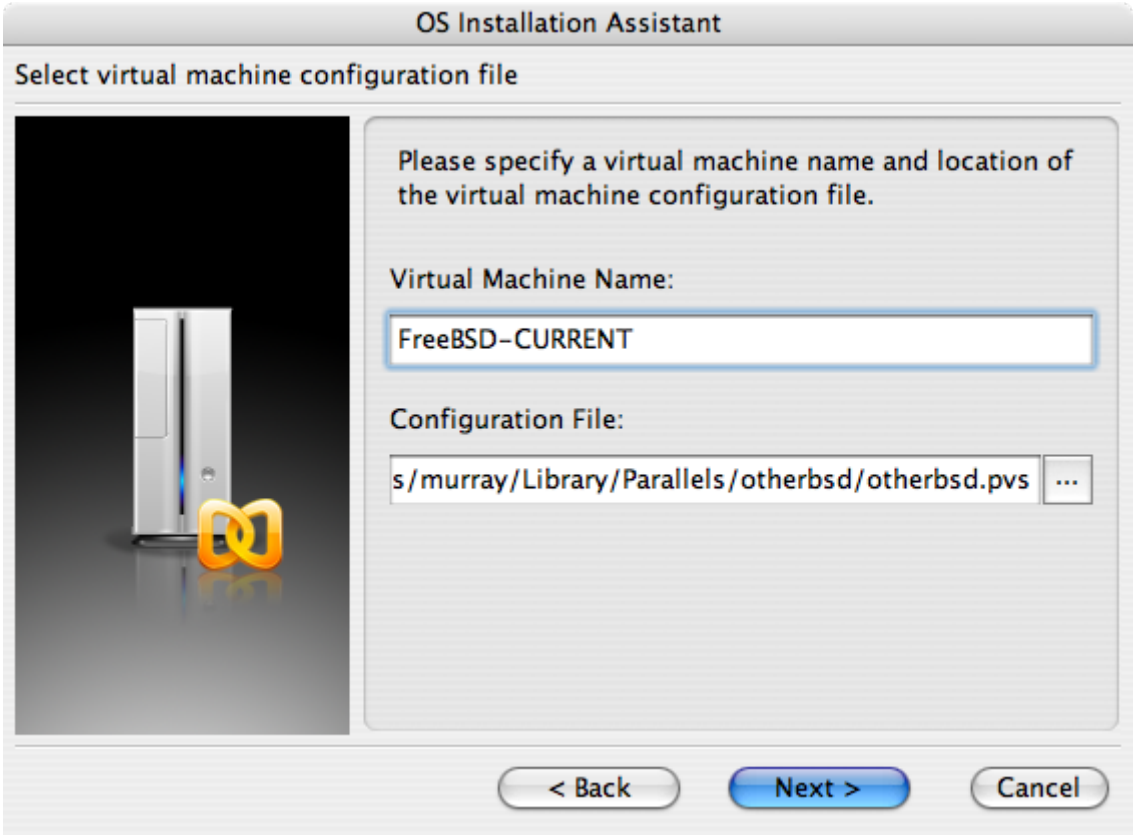

Installation Assistant
Name and Location
Name:
FreeBSD
Save to:
/Users/minsoochoo/Parallels
0 KB of disk space will be used
Create alias on Mac desktop
Customize settings before installation
Go Back
Create

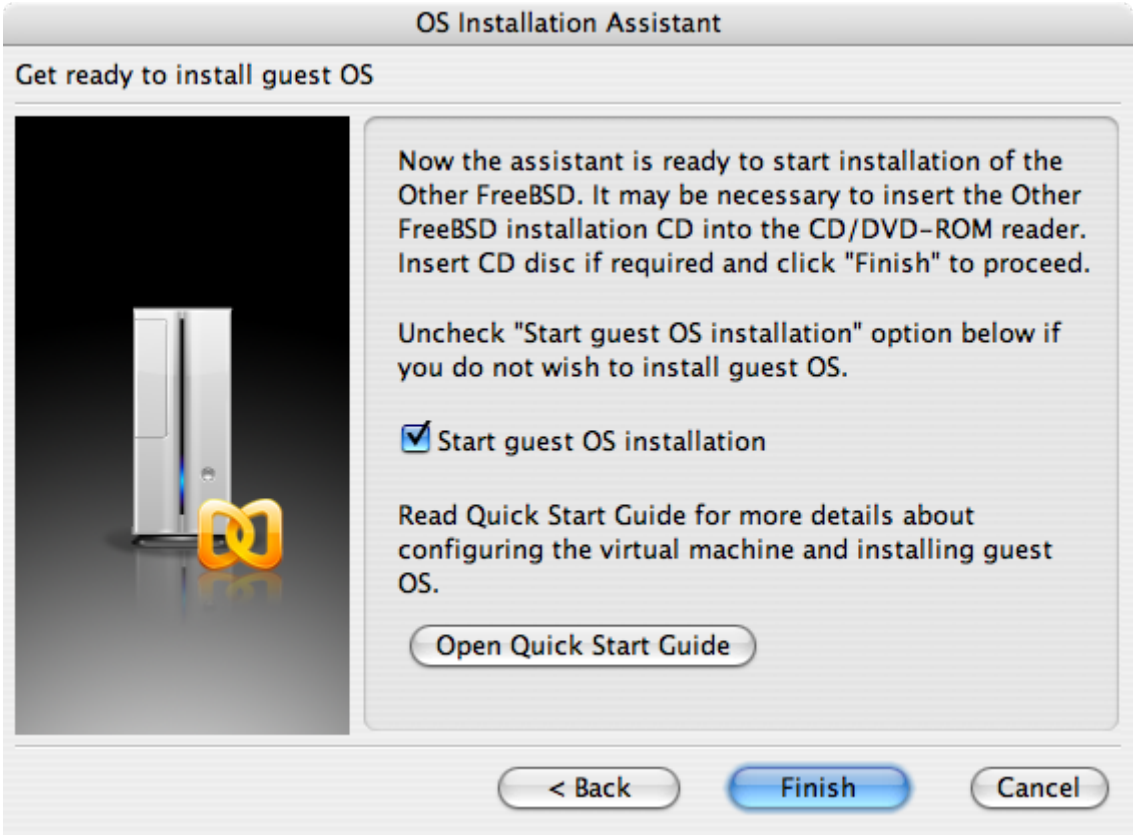



Сүлжээний төрөл болон сүлжээний интерфэйсийг сонгоно:

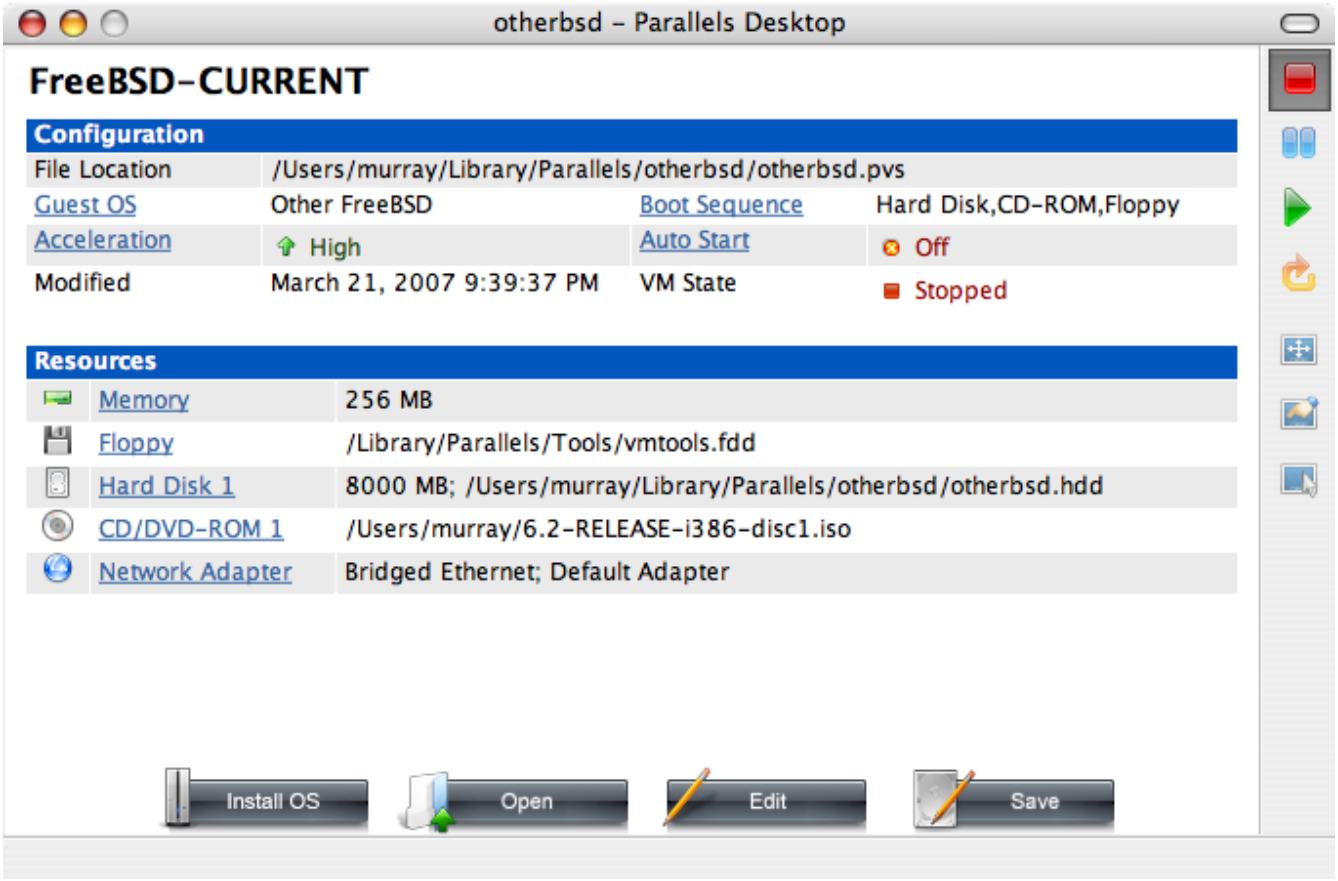

Installation Assistant
Virtual Machine Configuration
FreeBSD
Welcome to Other OS
We hope you know what it is
Good luck
CPUs: 2
Memory: 256 MB
Disk space: 8 GB
Configure...
Continue

Хадгалаад тохиргоог дуусгана:

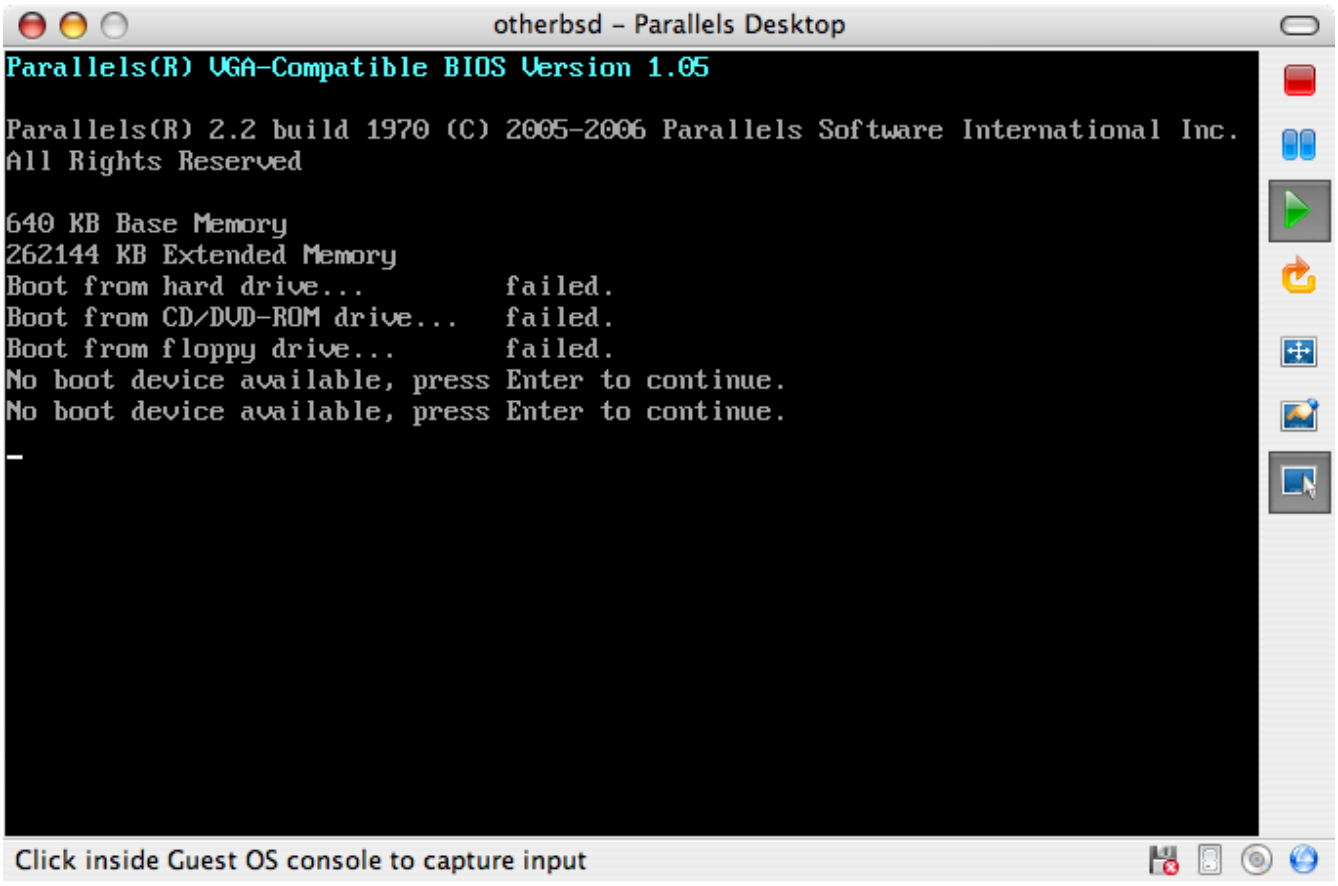





FreeBSD виртуал машин үүсгэгдсэний дараа FreeBSD-г түүн дээр суулгаж болно. Албан ёсны FreeBSD CDROM эсвэл албан ёсны FTP сайтаас татан авсан ISO дүрс ашиглан суулгах нь зүйтэй юм. Локал Mac® файлын систем дээрээ тохирох ISO дүрсийг хуулах юм уу эсвэл Mac®-ийн CD хөтөч дээр CDROM хийх хэрэгтэй. FreeBSD Parallels цонхны баруун доод талд байгаа дискний дүрсэн дээр дарах хэрэгтэй. Энэ нь виртуал машин дахь CDROM хөтчийг диск дээрх ISO файл эсвэл жинхэнэ CDROM хөтөчтэй холбох боломжийг олгох цонхыг харуулах болно.

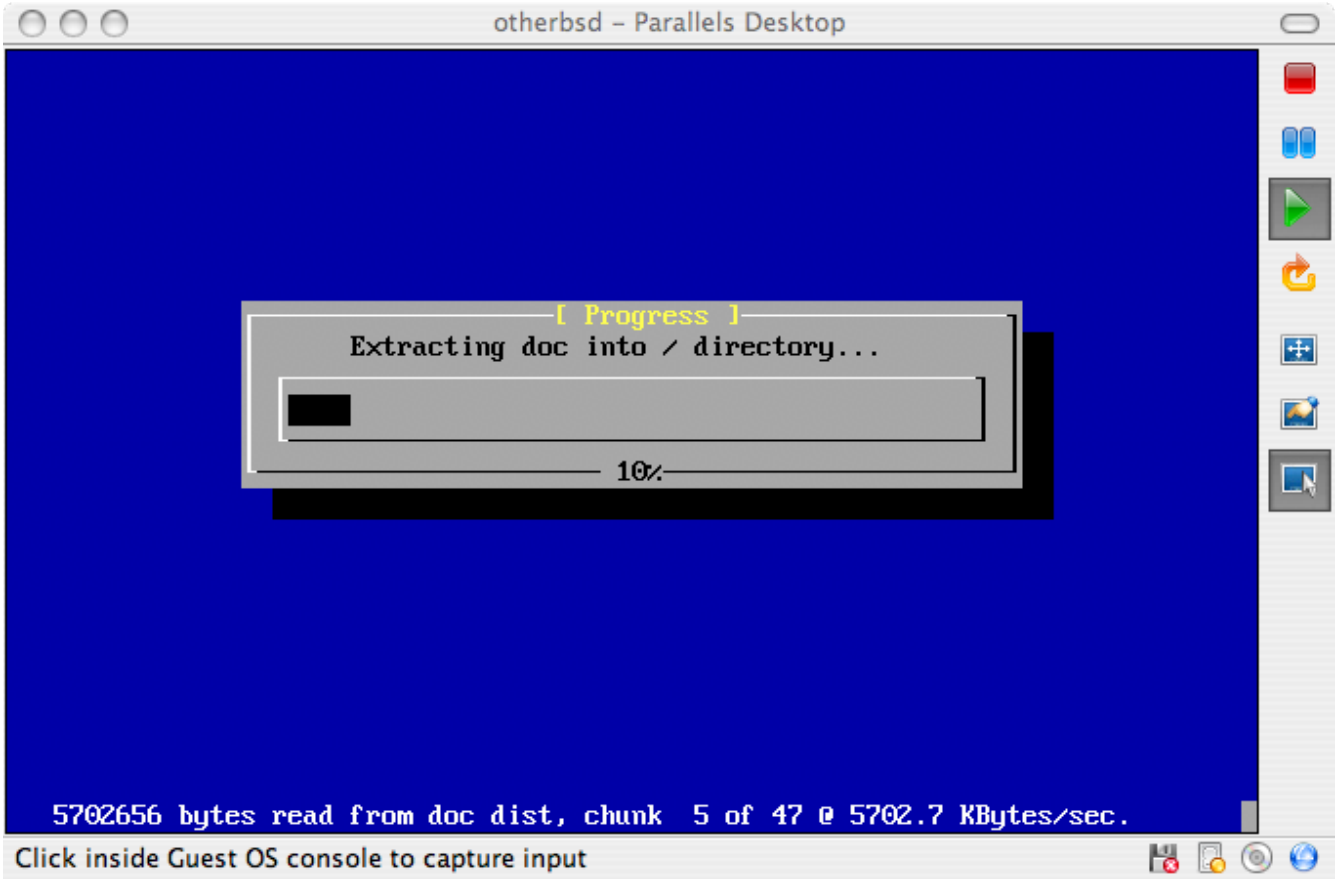



CDROM эхтэй энэ холболтыг хийснийхээ дараа дахин ачаалах дүрсийг дарж FreeBSD машиныг хэвийн үед дахин ачаалдаг шигээ дахин ачаална. Parallels нь танд CDROM байгаа эсэхийг шалгах тусгай BIOS-оор эхлэх болно.

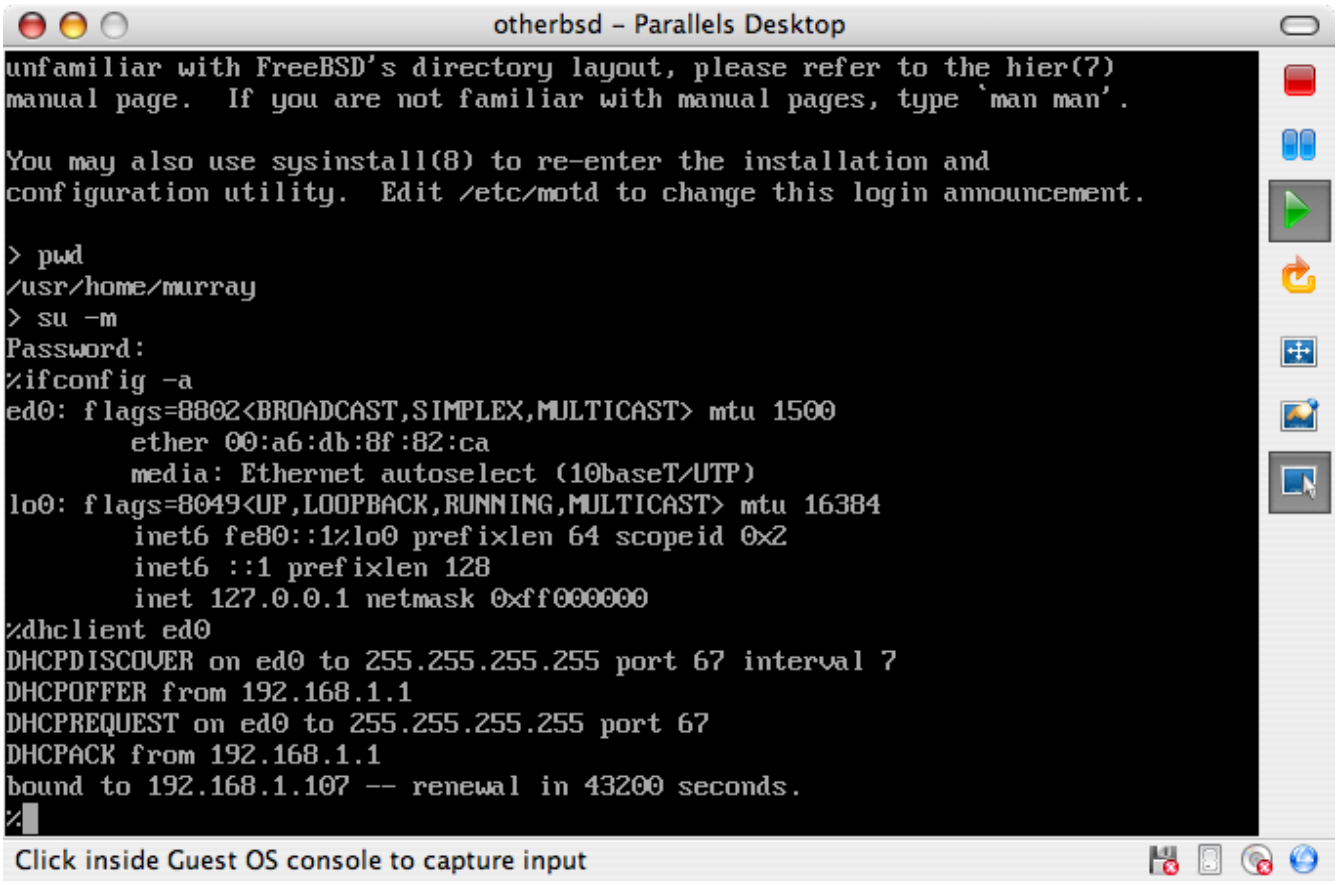



Энэ тохиолдолд энэ нь FreeBSD-ийн суулгах зөөвөрлөгчийг олж ердийн FreeBSD суулгалтыг

эхлүүлэх болно. Та Xorg суулгаж болох боловч энэ удаад тохируулахыг битгий оролдоорой.




Суулгалтыг дууссаныхаа дараа та өөрийн шинэ суулгасан FreeBSD виртуал машин руугаа дахин ачаалж ороорой.



#### 23.2.1.2. FreeBSD-г Parallels дээр тохируулах

FreeBSD-г Parallels-ийн тусламжтай Mac OS® X дээр амжилттайгаар суулгасны дараа виртуалчлагдсан ажиллагаанд зориулж системийг оновчтой болгохын тулд хэд хэдэн тохиргооны алхмуудыг хийх ёстой.

1. Ачаалагч дуудагчийн хувьсагчуудыг тохируулна

   Хамгийн чухал алхам бол Parallels-ийн орчинд FreeBSD-ийн CPU-н хэрэглээг багасгахын тулд `kern.hz`-г багасгах явдал юм. Дараах мөрийг /boot/loader.conf файлд нэмэн үүнийг хийж болно:

   ```
   kern.hz=100
   ```

   Энэ тохиргоогүйгээр, сул зогсож байгаа FreeBSD Parallels зочин нэг процессор бүхий iMac®-ийн CPU-ийн 15%-г ойролцоогоор ашиглах болно. Энэ өөрчлөлтийн дараа хэрэглээ нь 5% руу ойртох болно.

2. Шинэ цөмийн тохиргооны файлыг үүсгэнэ

   Бүх SCSI, FireWire, болон USB төхөөрөмжийн драйверуудыг цөмийн тохиргооны файлаас арилгаж болно. Parallels нь `ed` драйверийн ашигладаг виртуал сүлжээний адаптерийн боломжтой байдаг, тэгэхээр ed(4) болон miibus(4)-д зориулснаас бусад бүх сүлжээний төхөөрөмжүүдийг цөмөөс арилгаж болно.

3. Сүлжээг тохируулна

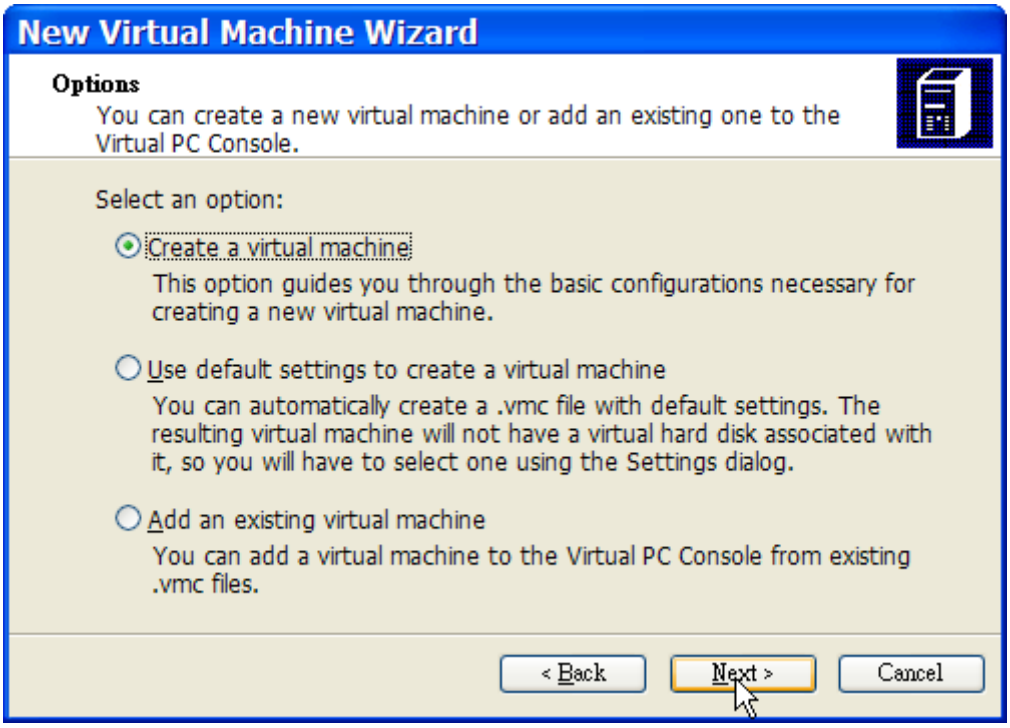
   Хамгийн хялбар сүлжээний тохиргоо бол хост Mac® шигээр ижил локал сүлжээнд виртуал машиныг холбохын тулд DHCP-г ашиглах явдал юм. Үүнийг /etc/rc.conf файлд `ifconfig_ed0="DHCP"` гэж нэмэн хийж болно. Сүлжээний илүү нарийн тохиргоонууд Сүлжээний нэмэлт ойлголтууд-д тайлбарлагдсан байгаа.

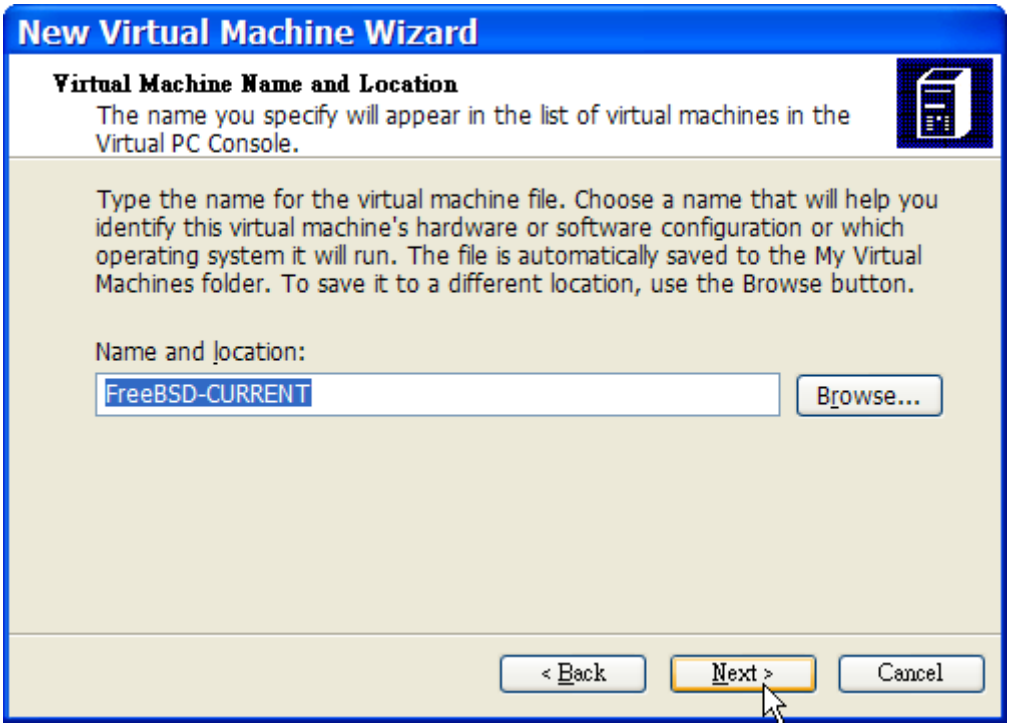
### 23.2.2. Windows® дээрх Virtual PC

Windows®-д зориулсан Virtual PC нь чөлөөтэй татаж авахаар байдаг, Microsoft®-ийн програм хангамж юм. Системийн шаардлагууд хуудсыг үзнэ үү. Microsoft® Windows® дээр Virtual PC суусны дараа хэрэглэгч виртуал машинаа тохируулж хүссэн зочин үйлдлийн системээ суулгах боломжтой.

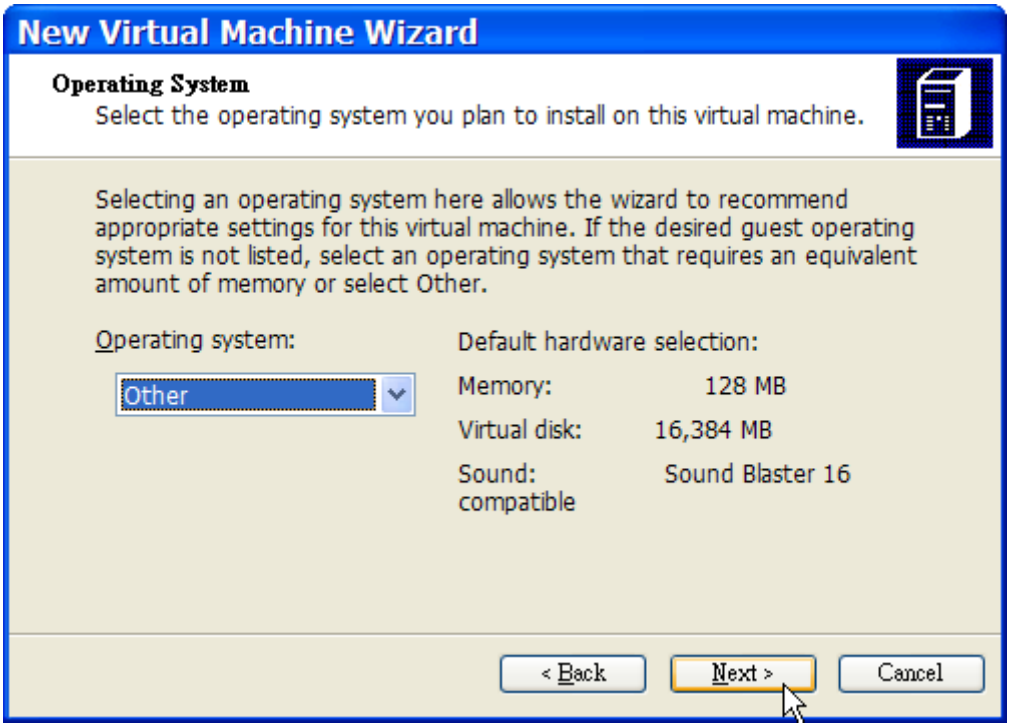
#### 23.2.2.1. Virtual PC дээр FreeBSD суулгах нь

Virtual PC дээр FreeBSD суулгах эхний алхам нь FreeBSD-г суулгах шинэ виртуал машин үүсгэх явдал юм. Асуухад нь Create a virtual machine гэсэн цэсийг сонгоно:

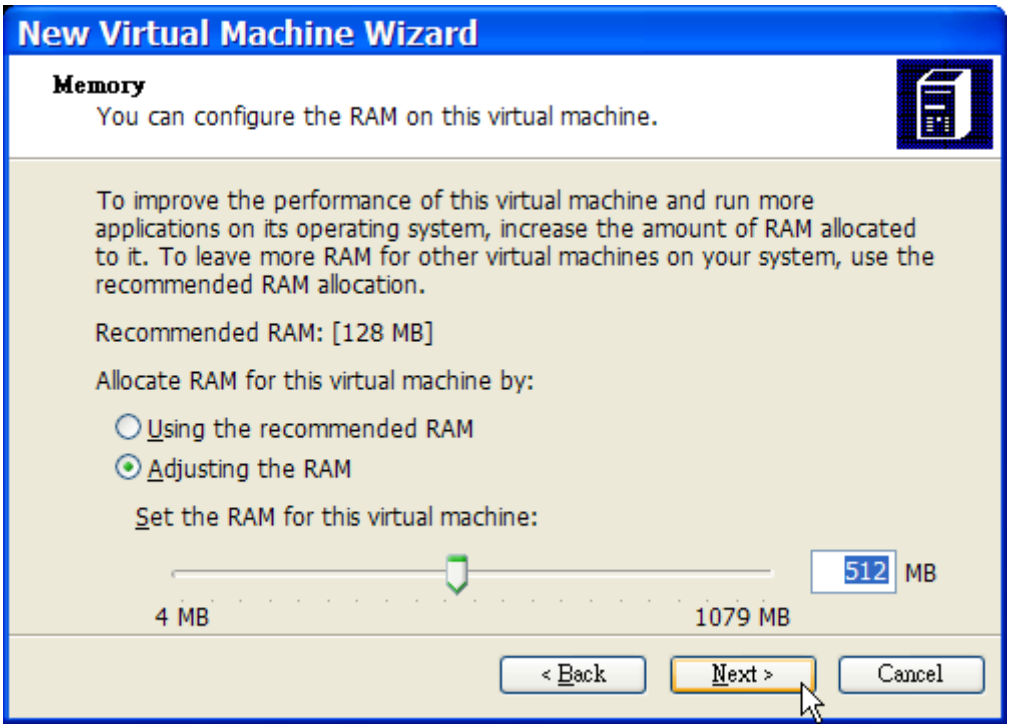


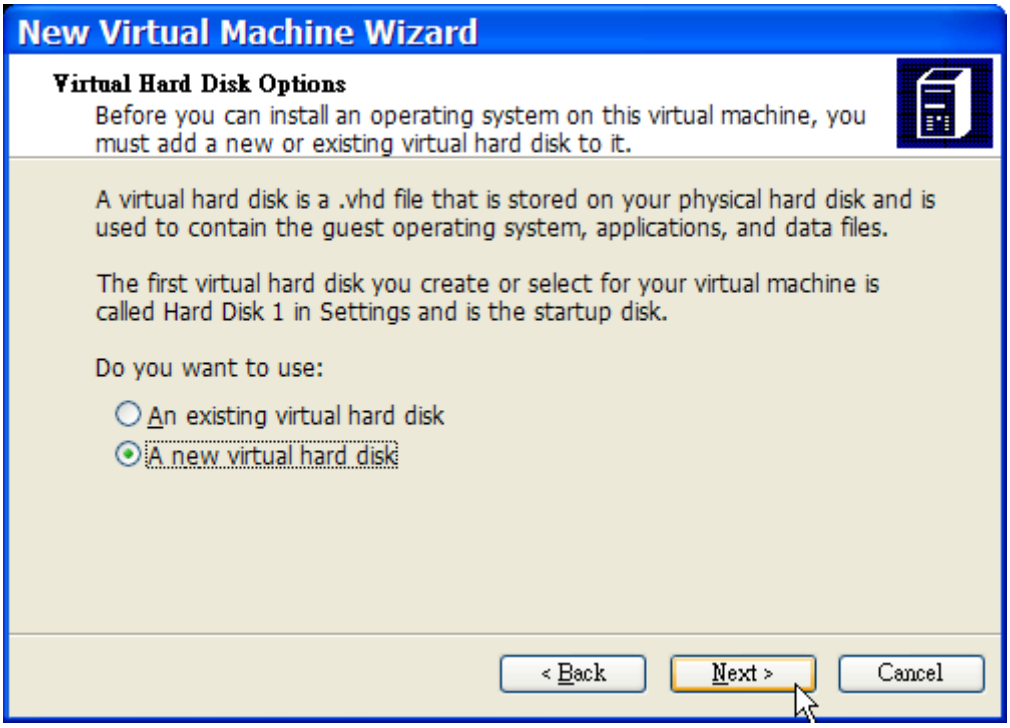




Тэгээд дараа нь Operating system гэдэг дээр Other гэдгийг сонгоно:

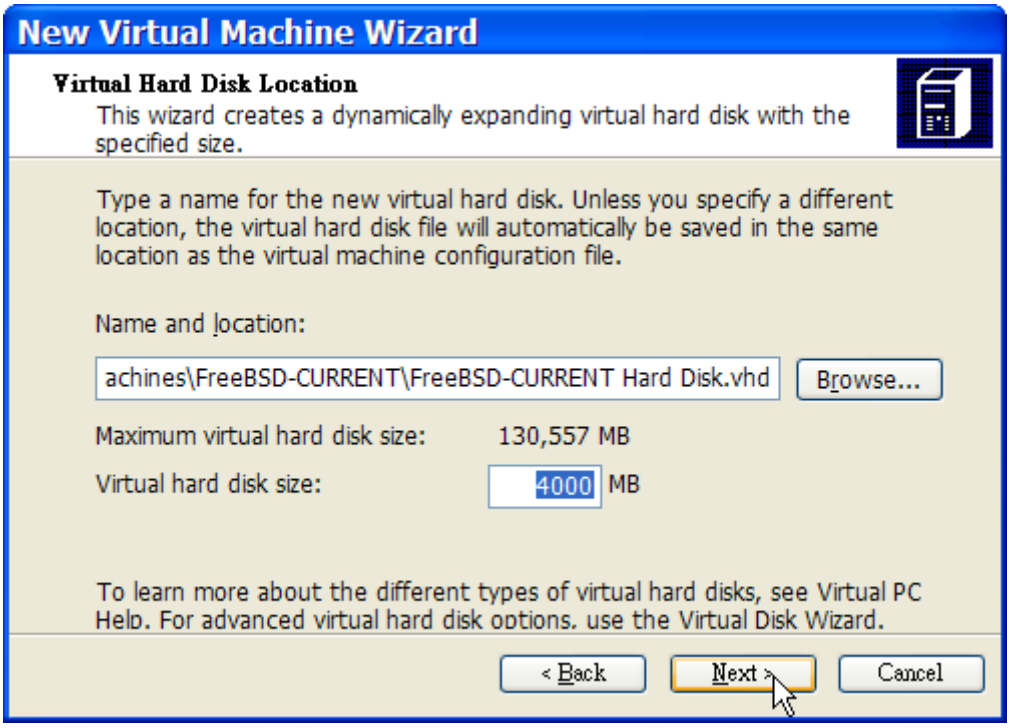



Дараа нь энэ виртуал FreeBSD-д зориулсан өөрийн төлөвлөгөөнөөсөө хамаараад хангалттай диск болон санах ойн хэмжээг сонгох хэрэгтэй. Virtual PC дээр FreeBSD-г ашиглахад ихэнхдээ 4GB дискний зай болон 512MB RAM хангалттай байдаг:

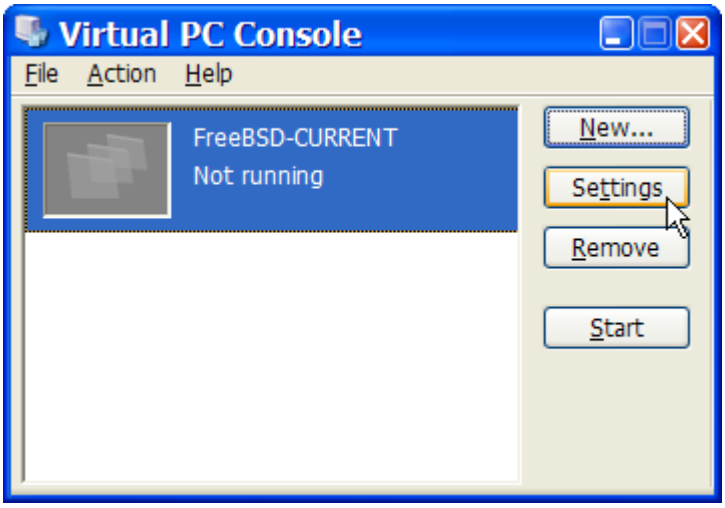


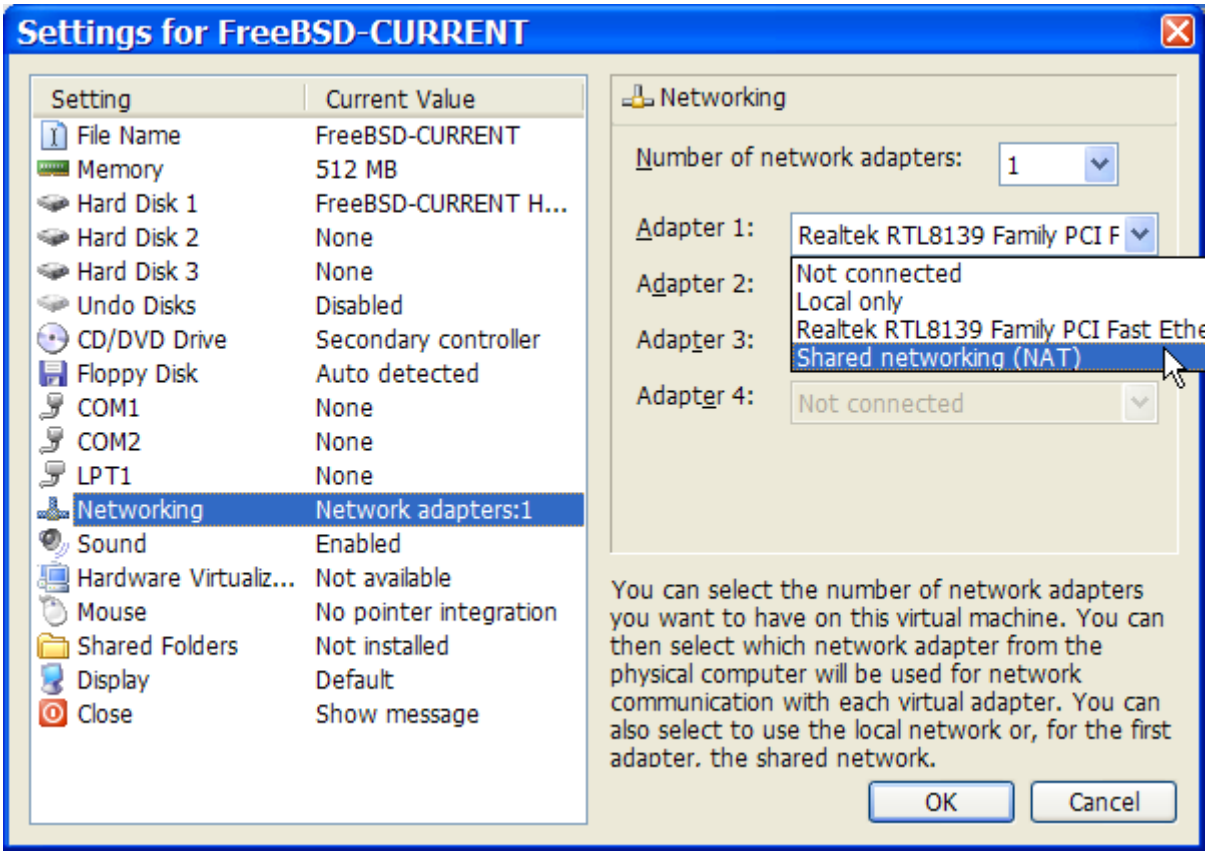



Хадгалаад тохиргоог дуусгана:

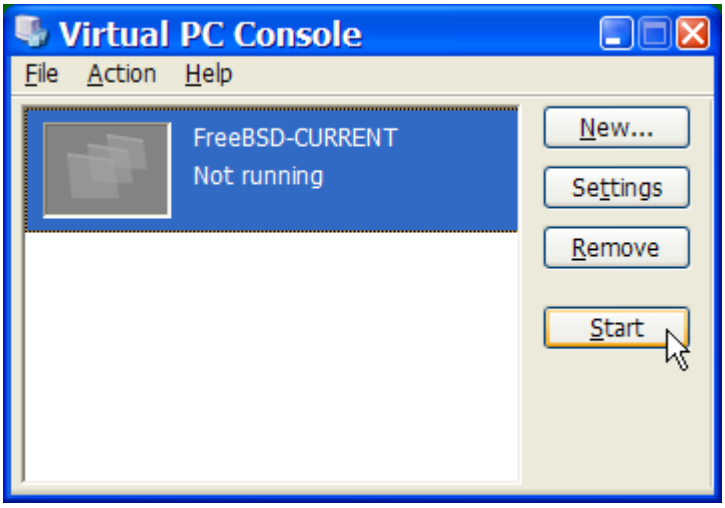


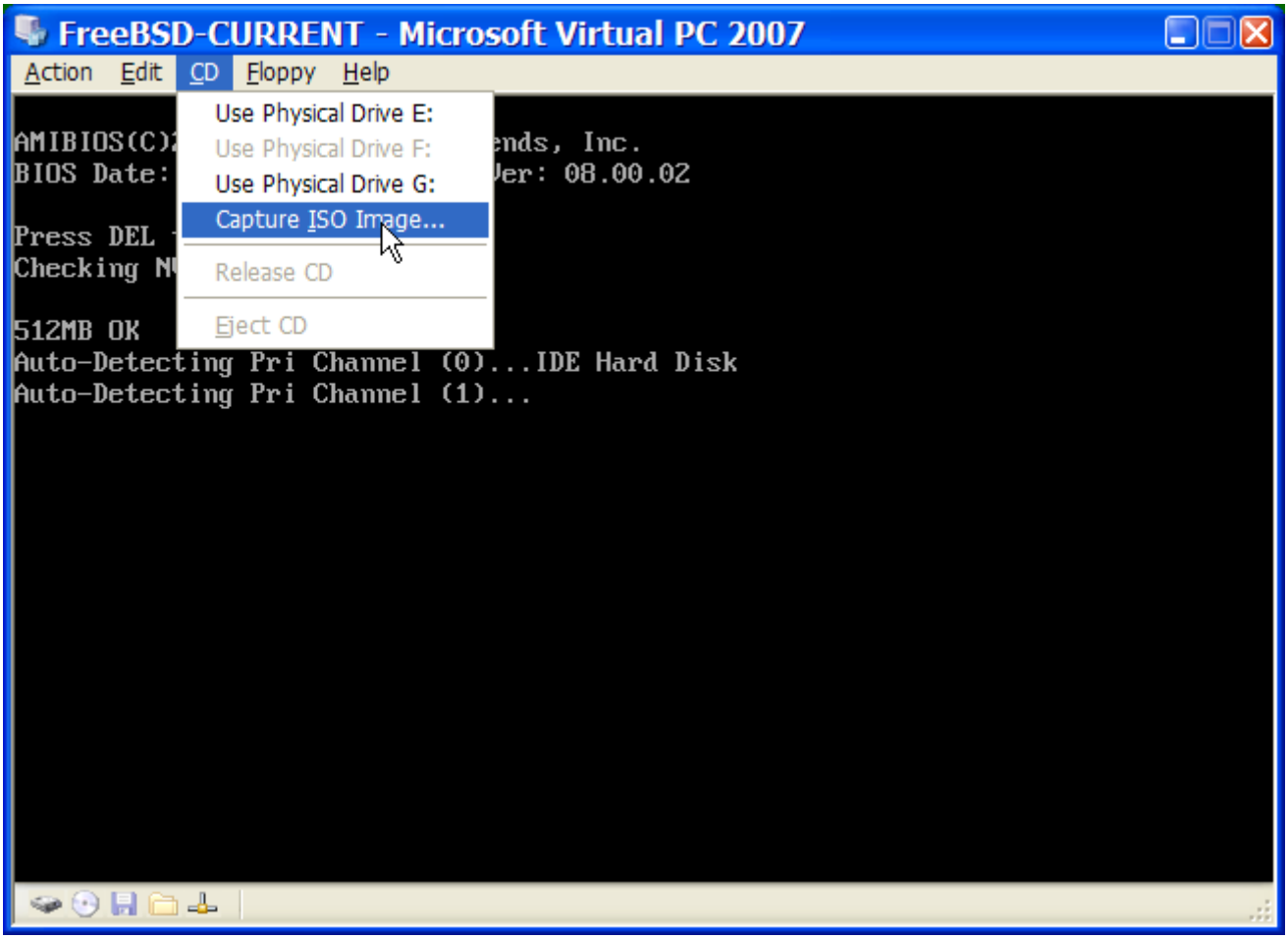

Өөрийн FreeBSD виртуал машинаа сонгоод **Settings** дээр дараад дараа нь сүлжээний төрөл болон сүлжээний интерфэйсийг тохируулна:



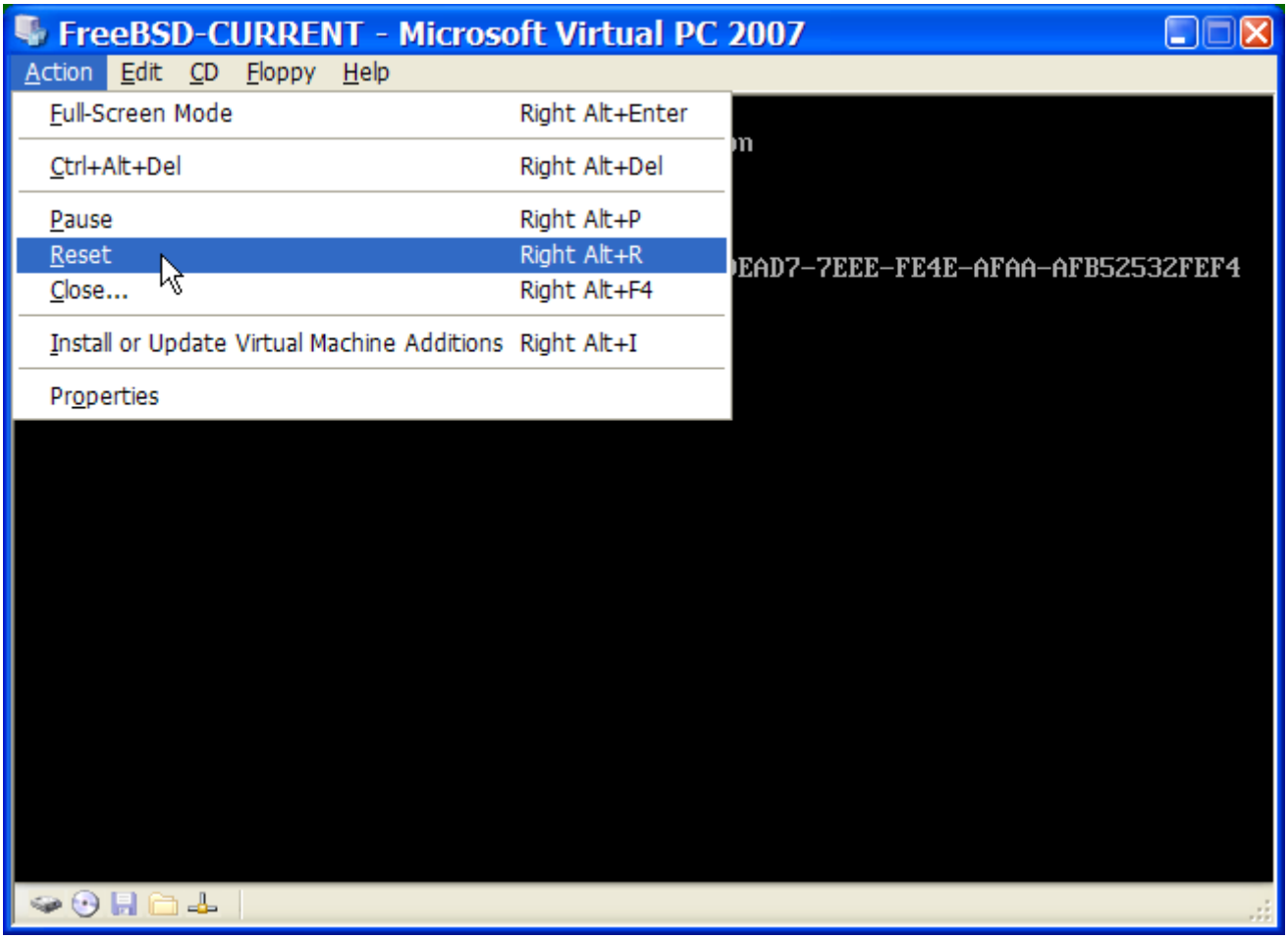




FreeBSD машин үүсгэгдсэний дараа түүн дээр FreeBSD суулгах боломжтой болно. Үүнийг албан ёсны FreeBSD CDROM эсвэл албан ёсны FTP хаягаас татаж авсан ISO дүрснээс суулгах нь зүйтэй. Windows® файлын систем дээр тохирсон ISO дүрс хуулах эсвэл CD хөтөчдөө CDROM хийснийхээ дараа өөрийн виртуал FreeBSD машин дээр хоёр дарж эхлүүлэх хэрэгтэй. Дараа нь Virtual PC цонхны **CD** цэсэн дээр дарж **Capture ISO Image...** цэсийг сонгоно. Ингэснээр виртуал машин дээр CDROM хөтчийг диск дээр байгаа ISO файлтай эсвэл жинхэнэ CDROM хөтөчтэй холбох боломжийг олгох цонхыг гаргаж ирэх юм.

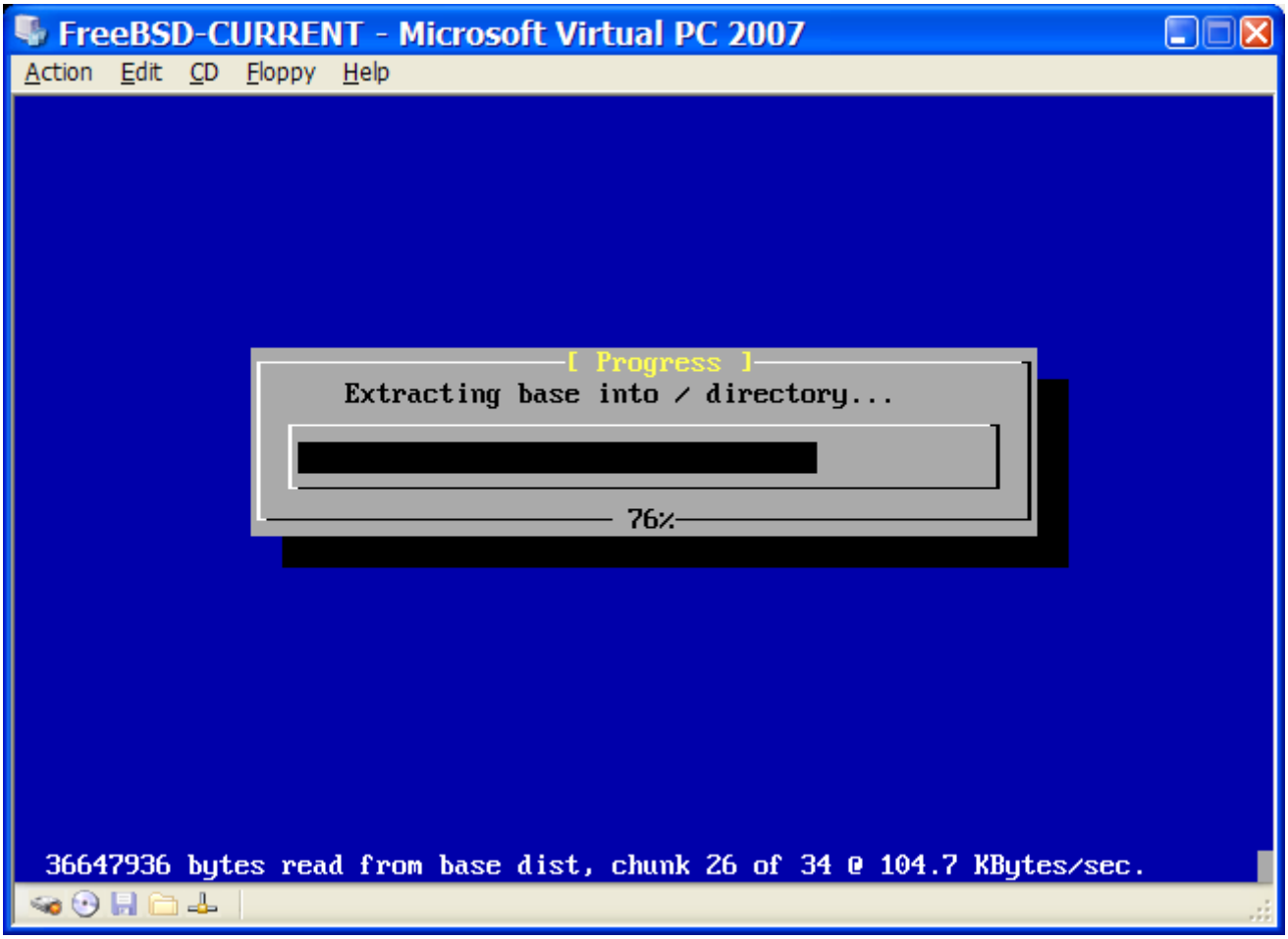


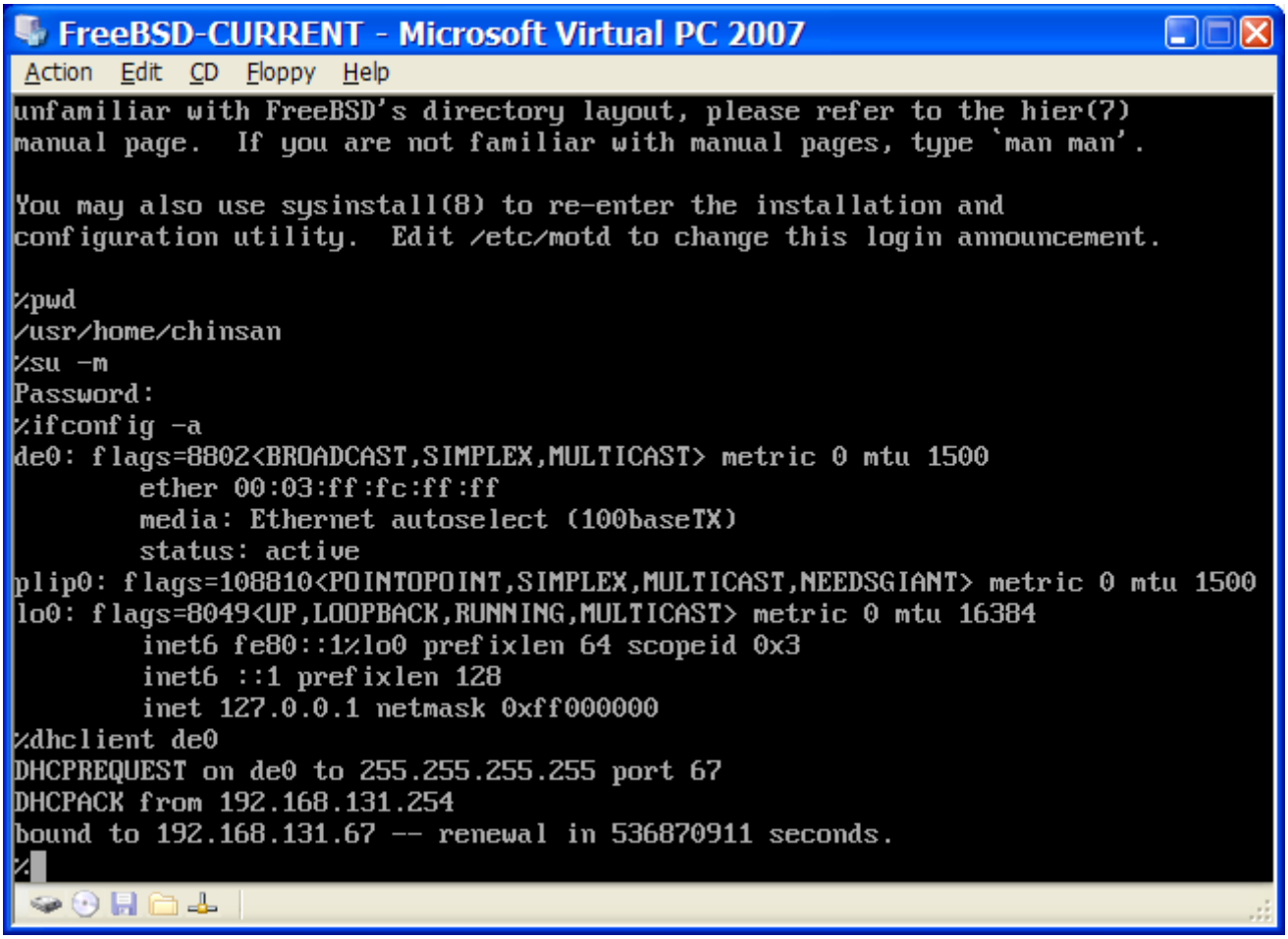

CDROM эхтэй энэ холболтыг хийсний дараа өөрийн FreeBSD виртуал машиныг **Action** болон **Reset** цэсүүдийг сонгон дахин ачаалах хэрэгтэй. Virtual PC нь CDROM байгаа эсэхийг эхлэн шалгах тусгай BIOS-оор дахин ачаалах болно.




Энэ тохиолдолд FreeBSD-ийн суулгац зөөвөрлөгчийг олж FreeBSD суулгалтыг эхлүүлнэ. Xorg тохируулахыг бүү оролдоорой.




Суулгаж дууссаныхаа дараа CDROM эсвэл хувилбарын ISO дүрсээ гаргаж авахаа мартуузай. Төгсгөлд нь өөрийн шинэхэн суулгасан FreeBSD виртуал машин уруугаа дахин ачаалж орох хэрэгтэй.




#### 23.2.2.2. Virtual PC дээр FreeBSD-г тохируулах нь

Microsoft® Windows® дээр Virtual PC ашиглан FreeBSD-г амжилттай суулгасны дараа виртуалчлагдсан үйлдэлд зориулж системийг оновчтой болгон хийж болох хэд хэдэн тохиргооны алхмууд байдаг.

1. Ачаалагч дуудагчийн хувьсагчуудыг тохируулна

   Хамгийн чухал алхам бол Virtual PC орчинд FreeBSD-ийн CPU-ийн хэрэглээг багасгахын тулд `kern.hz` тохиргоог багасгах явдал юм. Үүнийг хийхийн тулд доор дурдсан мөрийг /boot/loader.conf файлд нэмнэ:

   ```
   kern.hz=100
   ```

   Энэ тохиргоогүй бол сул зогсож байгаа FreeBSD Virtual PC зочин үйлдлийн систем нэг процессортой компьютерийн CPU-ний 40%-ийг ойролцоогоор ашиглах болно. Энэ тохиргооны дараа хэрэглээ дөнгөж 3% уруу дөхөж очих болно.

2. Цөмийн шинэ тохиргооны файл үүсгэнэ

   SCSI, FireWire болон USB төхөөрөмжийн бүх драйверуудыг цөмийн тохиргооны файлаас устгаж болно. Virtual PC нь de(4) драйверийн ашигладаг виртуал сүлжээний адаптерийн боломжтой бөгөөд тэгэхээр de(4) болон miibus(4)-с бусад бүх сүлжээний төхөөрөмжүүдийг цөмөөс арилгаж болно.

3. Сүлжээг тохируулна

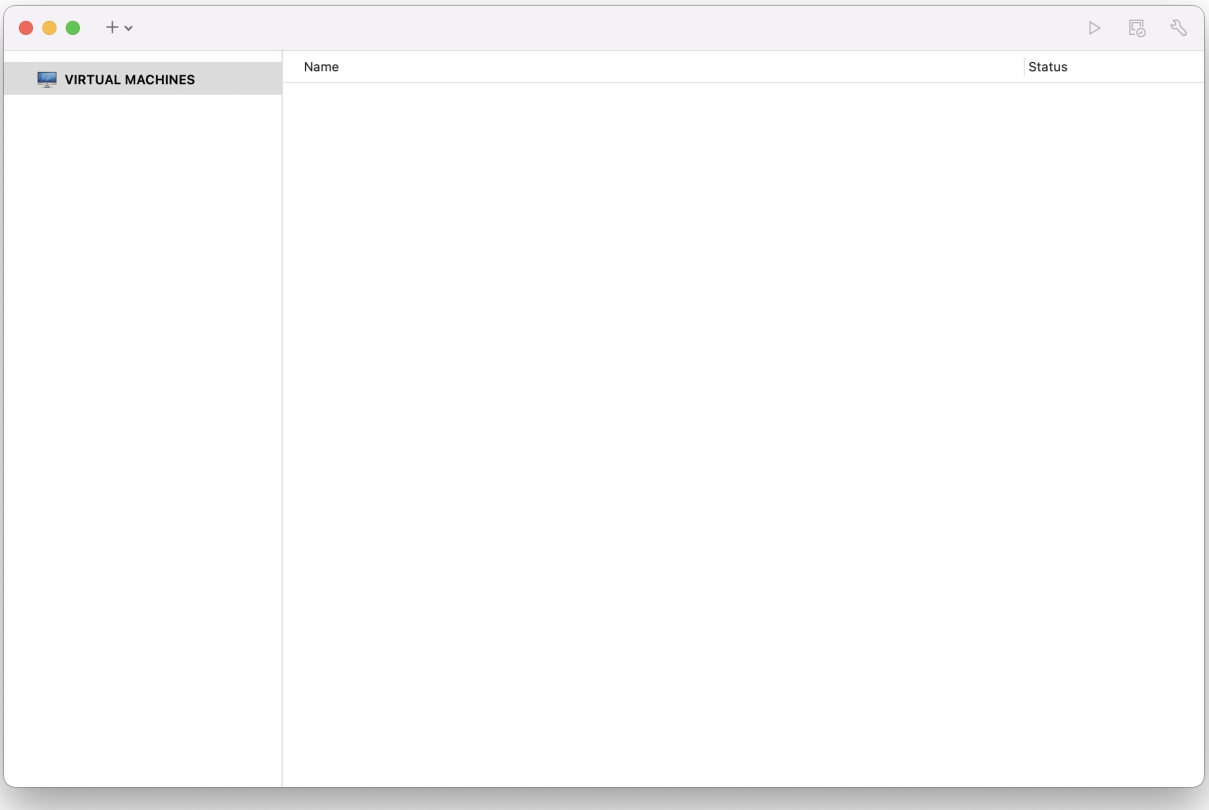
   Хамгийн энгийн сүлжээний тохиргоо нь Microsoft® Windows® холбогдсон тэр дотоод сүлжээндээ виртуал машиныг холбохдоо DHCP-ийг ашиглах явдал юм. `ifconfig_de0="DHCP"` мөрийг /etc/rc.conf файлд нэмж үүнийг хийнэ. Сүлжээний илүү нарийн тохиргоонуудын талаар Сүлжээний нэмэлт ойлголтууд дээр тайлбарласан байгаа болно.

## 23.2.3. Mac OS® дээрх VMware Fusion

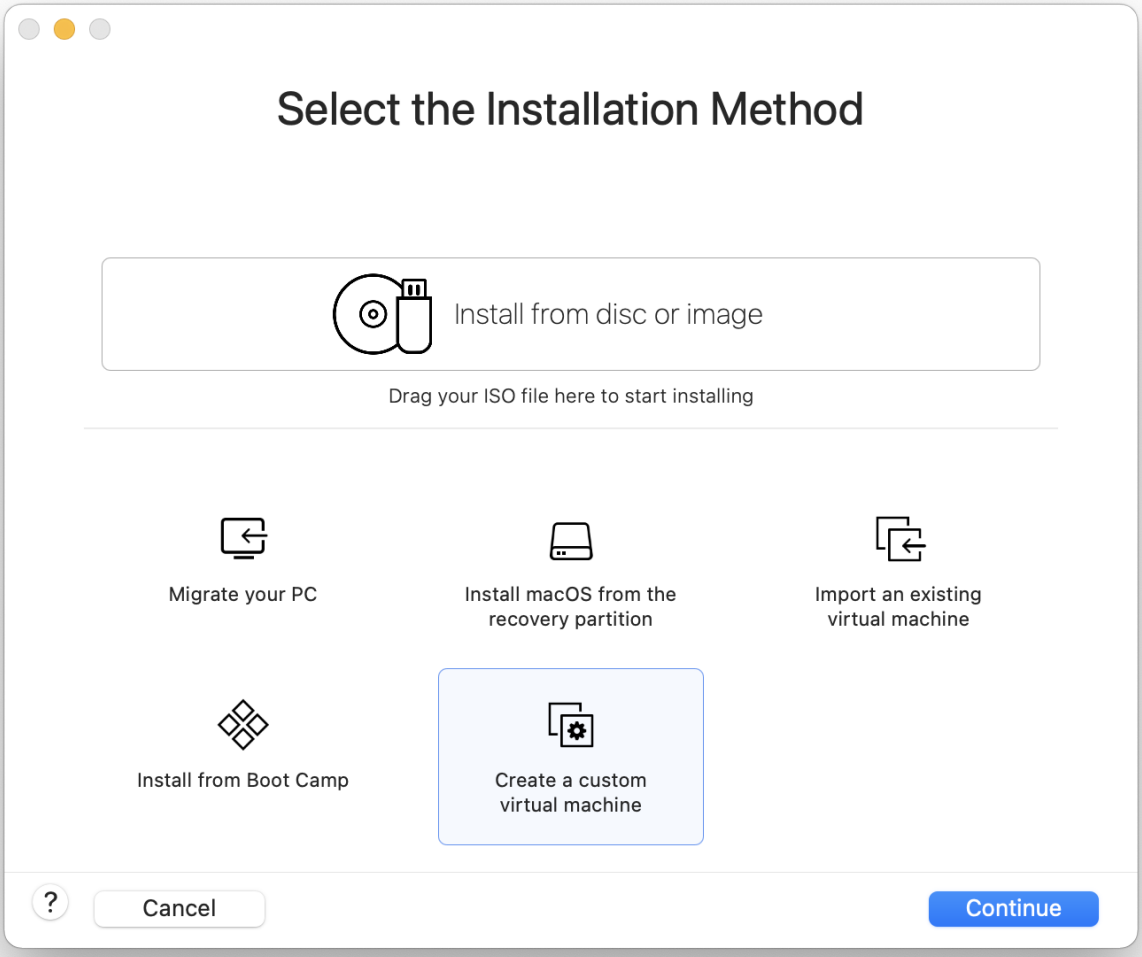
Mac®-д зориулсан VMware Fusion нь Mac OS® 10.4.9 болон түүнээс хойшх хувилбар бүхий Intel® дээр тулгуурласан Apple® Mac® компьютеруудад зориулагдсан арилжааны програм хангамж бүтээгдэхүүн юм. FreeBSD нь бүрэн дэмжигдсэн зочин үйлдлийн систем юм. Mac OS® X дээр VMware Fusion суулгасны дараа хэрэглэгч виртуал машинаа тохируулж хүссэн зочин үйлдлийн системээ суулгах боломжтой.

### 23.2.3.1. VMware Fusion дээр FreeBSD суулгах нь

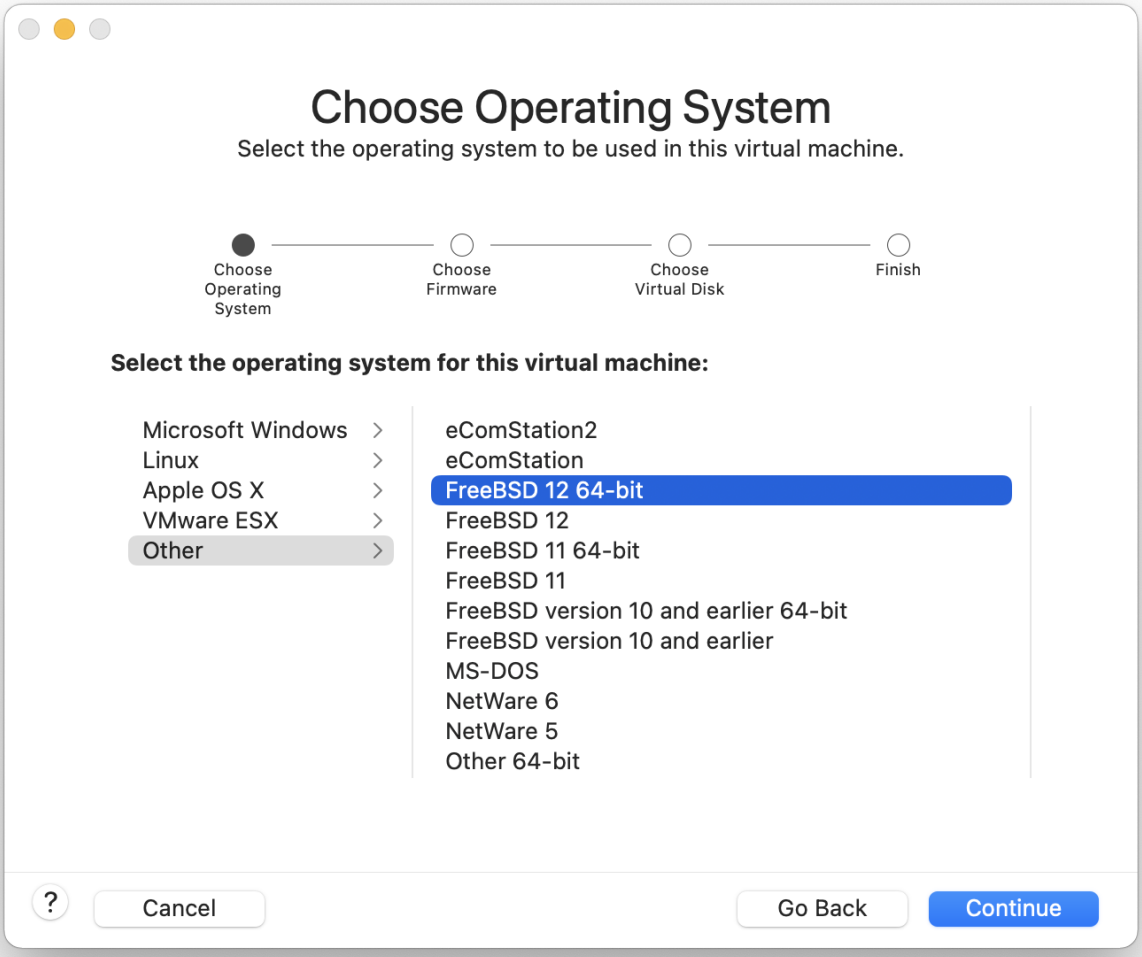
Эхний алхам нь VMware Fusion-г эхлүүлнэ. Ингэхэд Виртуал Машины Сан ачаалагдах болно. "New"-г дарж виртуал машин үүсгэнэ:



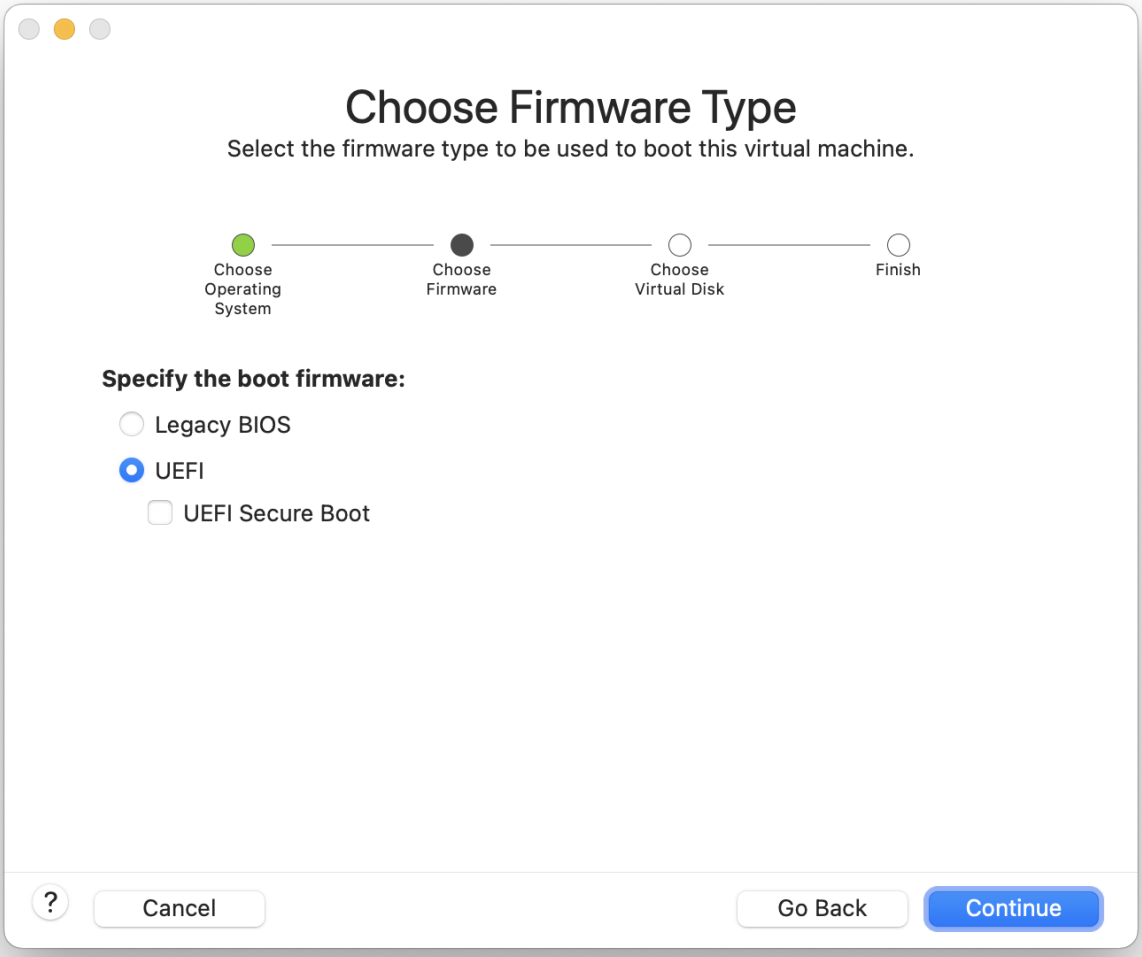

Энэ нь New Virtual Machine Assistant буюу Шинэ Виртуал Машины Туслагчийг ачаалах бөгөөд Continue дарж үргэлжлүүлнэ:



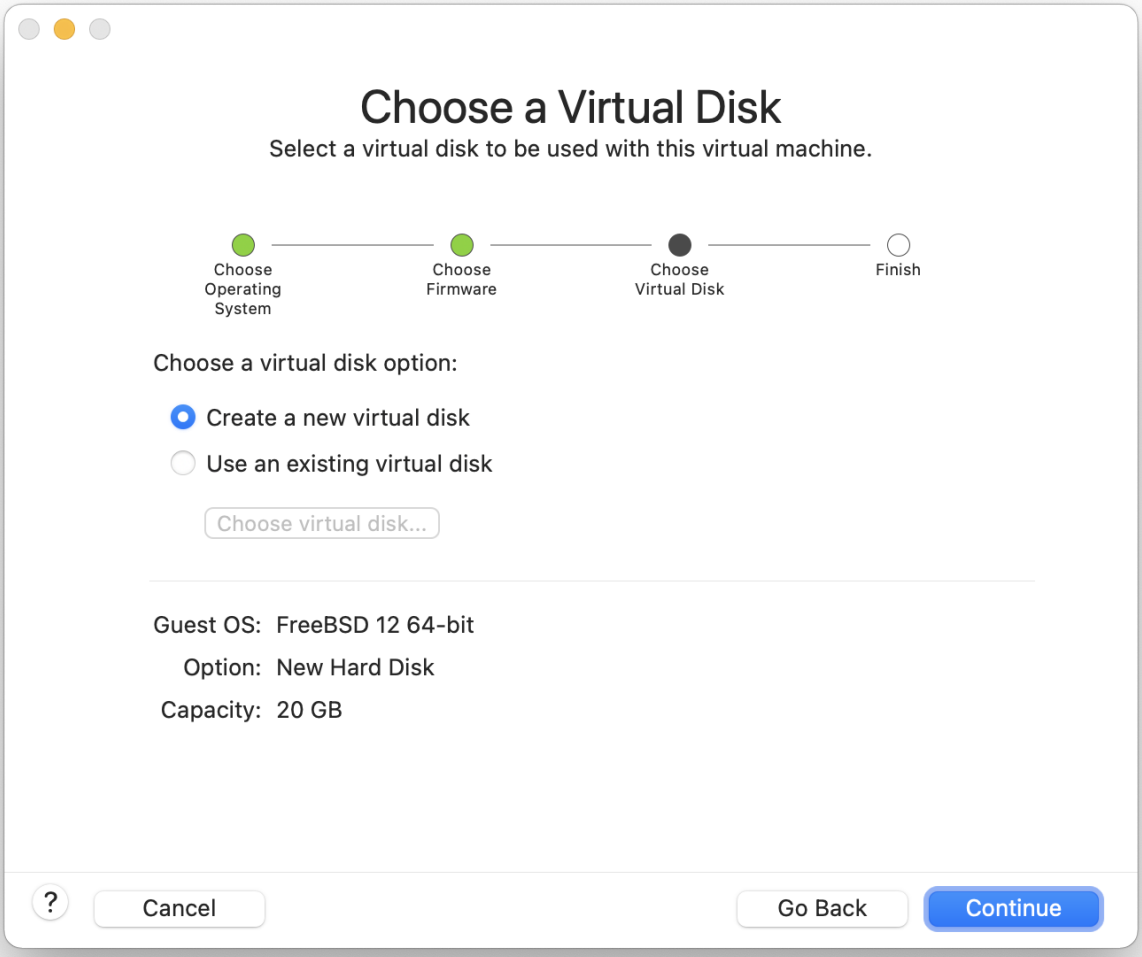

Operating System гэдэг дээр Other гэдгийг сонгоод FreeBSD юм уу эсвэл FreeBSD 64-bit гэдгийг **Version** гэдэг дээр сонгоорой:



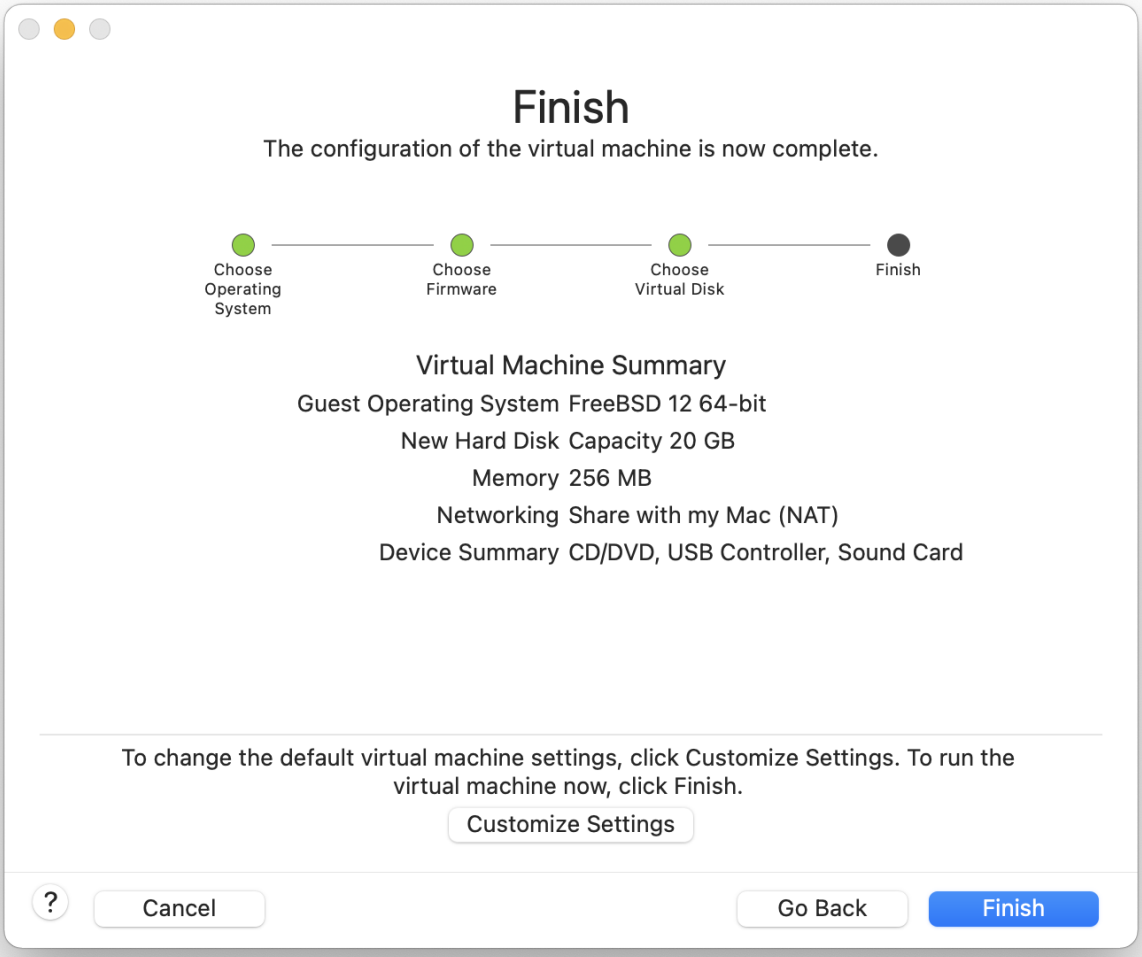

Виртуал машины нэр болон хадгалахыг хүсэж байгаа тэр сангийнхаа нэрийг сонгоно:



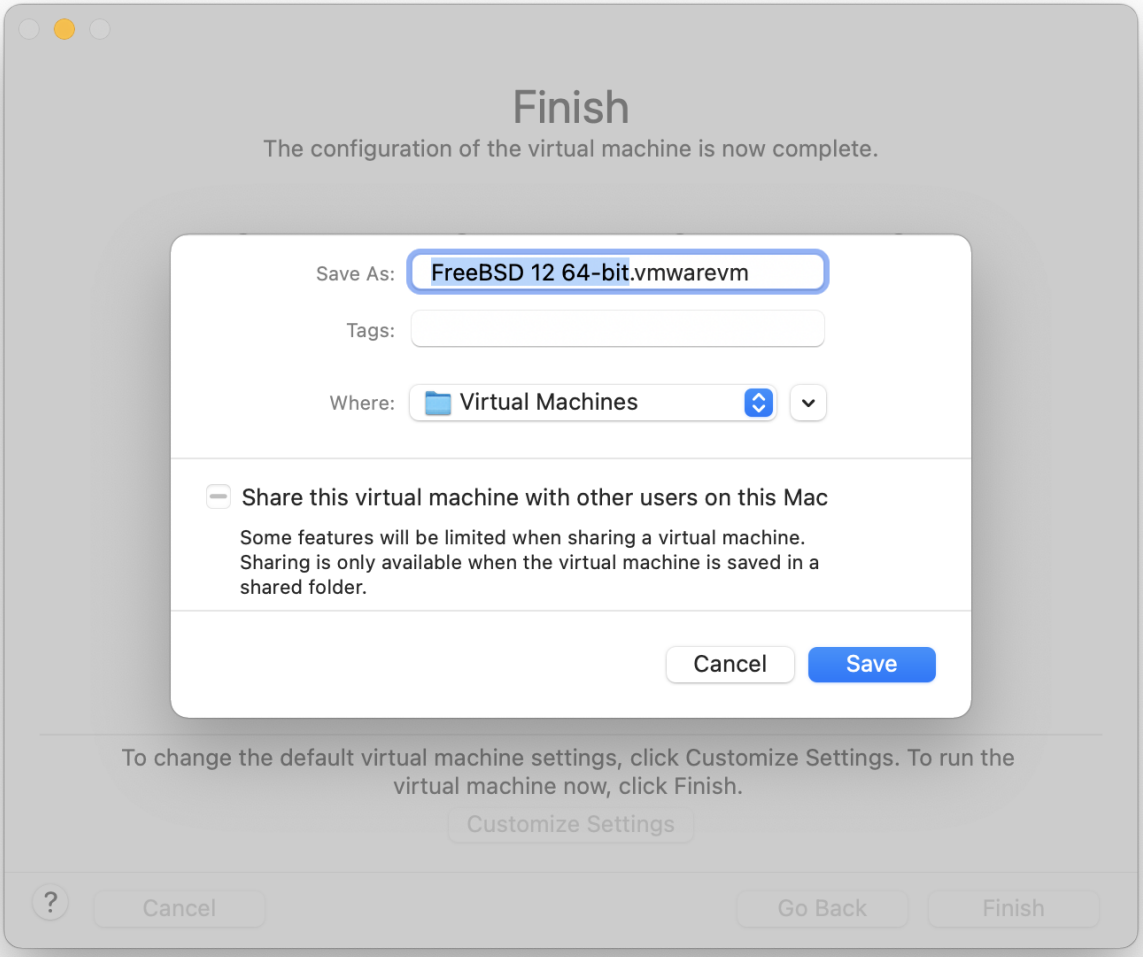

Виртуал машинд зориулсан Виртуал Хатуу Дискийнхээ хэмжээг сонгоно:




Виртуал машин суулгахдаа ISO дүрс эсвэл CD-ий аль нэгнээс суулгахыг сонгоно:



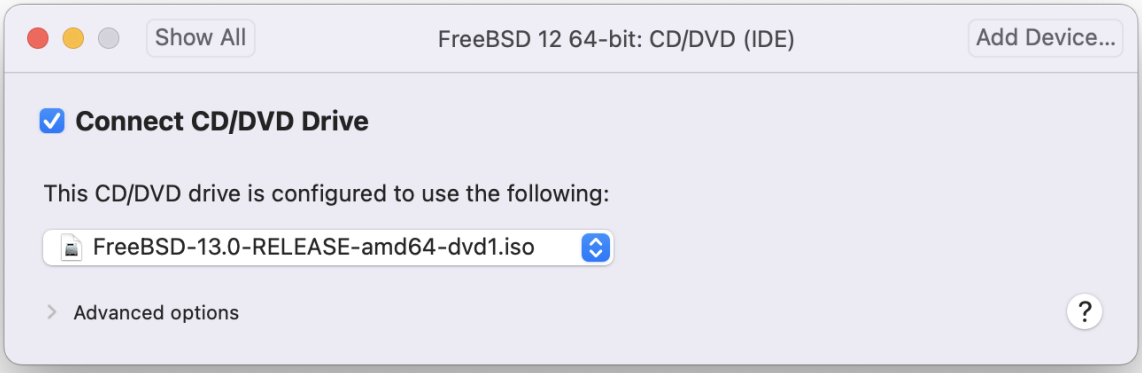

Finish-ийг дарсны дараа виртуал машин дахин ачаалах болно:




Ердийн суулгадагтайгаа адил FreeBSD-ийг суулгана:

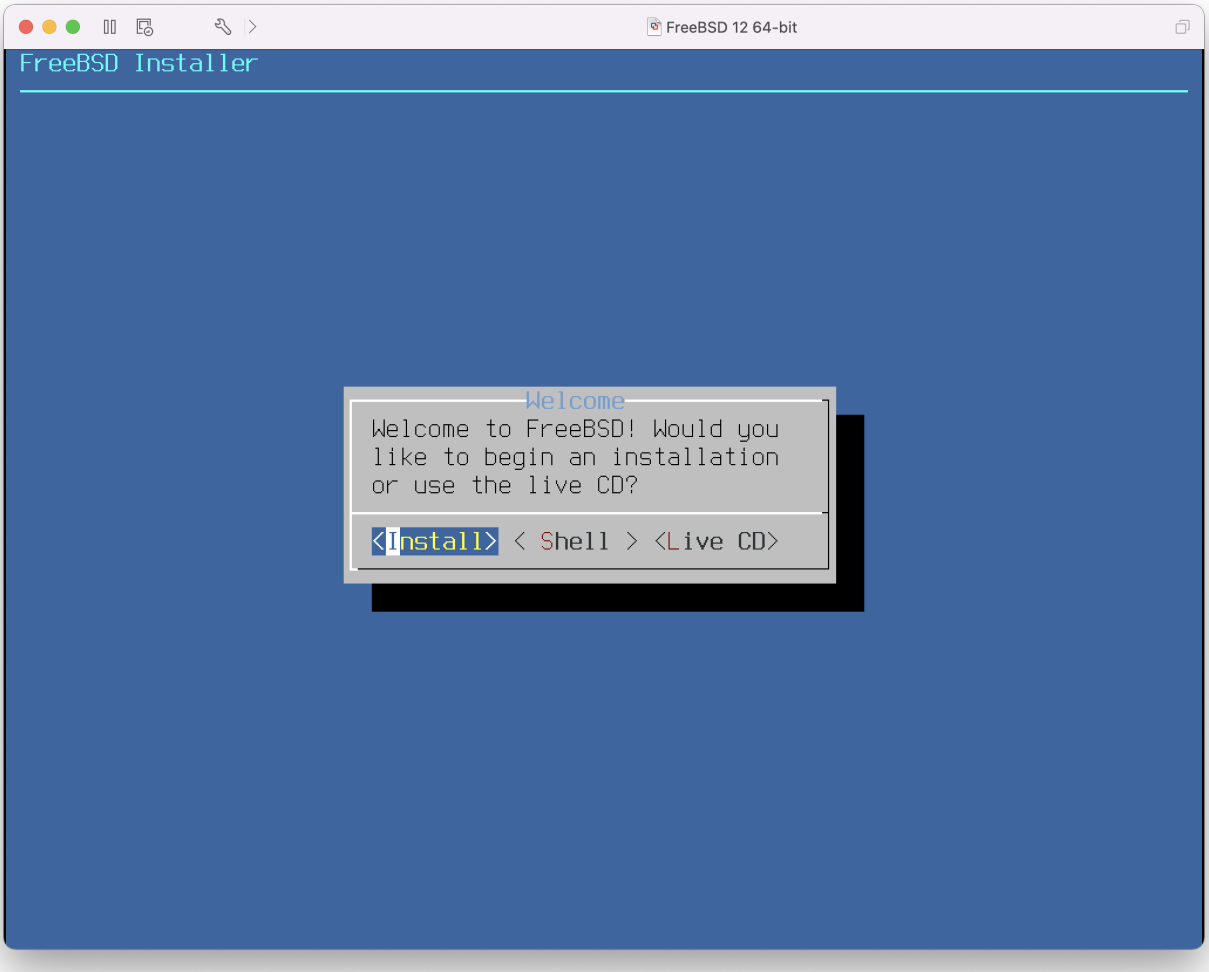
Суулгаж дууссаны дараа Санах ойн хэрэглээ зэрэг виртуал машины тохиргоонуудыг өөрчилж болно:

Виртуал машин ажиллаж байхад виртуал машины системийн тоног төхөөрөмжийн тохиргоонуудыг өөрчилж болдоггүй.

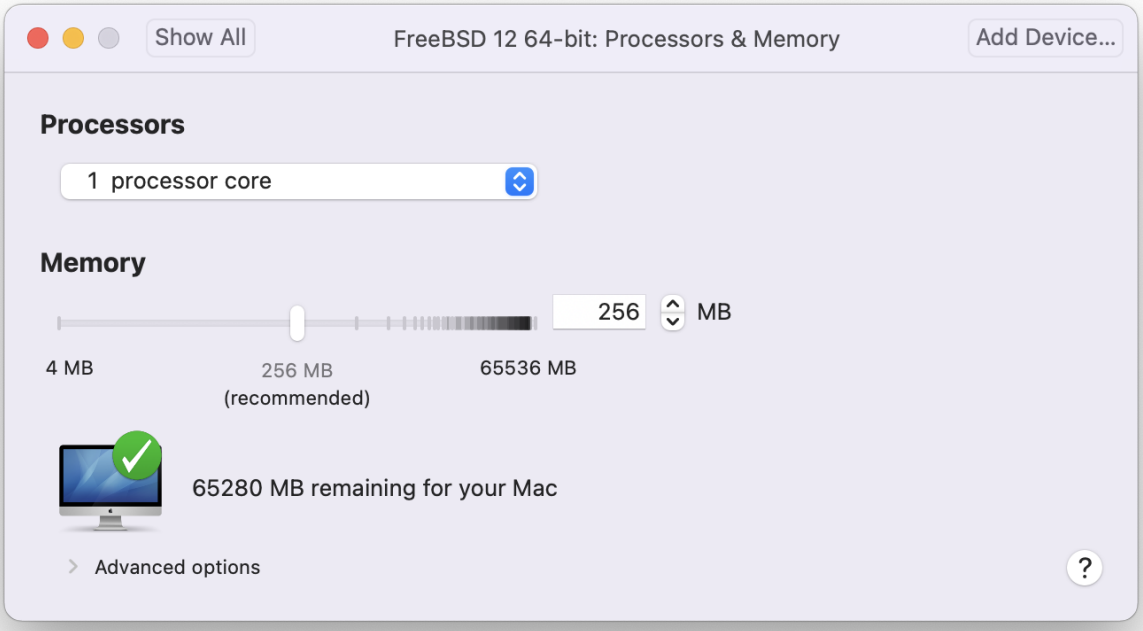



Виртуал машины хандаж болох CPU-ийн тоо:

CDROM төхөөрөмжийн төлөв. CD/DVD/ISO дахиж хэрэг болохгүй бол ерөнхийдөө виртуал машинаас салгаж болно.




Сүүлийн өөрчлөх юм нь сүлжээнд виртуал машин хэрхэн холбогдохыг заах юм. Эх хостоос нь гадна бусад машинуудаас виртуал машин рүү холболт хийхийн тулд Connect directly to the physical network (Bridged) буюу Физик сүлжээнд шууд холбогдох (Гүүр хийгдсэн) гэдгийг сонгоорой. Хэрэв тэгж хүсэхгүй байгаа бол виртуал машиныг Интернэт рүү хандах боломжтой болгодог боловч сүлжээнээс виртуал машин руу хандах боломжгүй Share the host's internet connection (NAT) буюу хостын интернэтийн холболтыг (NAT) хуваалцах гэсэн сонголтыг сонгохыг зөвлөдөг.




Тохиргоонуудыг өөрчилж дууссаныхаа дараа шинээр суулгасан FreeBSD виртуал машинаа ачаалах хэрэгтэй.

#### 23.2.3.2. FreeBSD-г VMware Fusion дээр тохируулах нь

Mac OS® X дээр VMware Fusion-ийн тусламжтайгаар FreeBSD амжилттай суугдсаны дараа виртуалчлагдсан үйлдэлд зориулж системийг оновчтой болгохын тулд хийгдэж болох хэд хэдэн тохиргооны алхмууд байдаг.

1. Ачаалагч дуудагчийн хувьсагчуудыг тохируулна

   VMware Fusion орчин доор FreeBSD-ийн CPU-ийн хэрэглээг багасгахын тулд `kern.hz` тохируулгыг багасгах нь хамгийн чухал алхам юм. Дараах мөрийг /boot/loader.conf-д нэмснээр үүнийг хийж болно:

   ```
   kern.hz=100
   ```

   Энэ тохиргоогүйгээр сул зогсож байгаа FreeBSD VMware Fusion зочин OS нь ганц процессор бүхий iMac®-ийн CPU-ний бараг 15%-ийг ашиглах болно. Энэ өөрчлөлтийн дараа ашиглалт дөнгөж 5% хүрэх хэмжээний байх болно.

2. Цөмийн шинэ тохиргооны файл үүсгэнэ

   FireWire, болон USB төхөөрөмжийн бүх л драйверуудыг цөмийн тохиргооны файлаас хасаж болно. VMware Fusion нь em(4) драйверийн ашигладаг виртуал сүлжээний адаптераар хангадаг учир em(4)-д зориулснаас бусад сүлжээний бүх төхөөрөмжүүдийг цөмөөс хасаж болно.

3. Сүлжээг тохируулна

   Хамгийн хялбар сүлжээний тохиргоо нь хост Mac® холбогдсон байгаа дотоод сүлжээндээ DHCP ашиглан виртуал машинаа холбох явдал юм. /etc/rc.conf файлд `ifconfig_em0="DHCP"` мөрийг нэмснээр үүнийг хийж болно. Сүлжээний илүү дэлгэрэнгүй тохиргоонуудын талаар Сүлжээний нэмэлт ойлголтууд-д тайлбарласан байгаа.

### 23.2.4. FreeBSD зочин дээрх VirtualBox™ зочин нэмэгдлүүд

VirtualBox™ зочин нэмэгдлүүд нь дараах боломжуудыг олгодог:

- Завсрын самбар (Clipboard) хуваалцах.
- Хулганы заагчийн нэгттэл.
- Хостын цагийн синхрончлол.
- Цонхны хэмжээ өөрчлөх боломж.
- Хялбар (Seamless) горим.

FreeBSD зочин дээр дараах тушаалуудыг ажиллуулна.

Эхлээд emulators/virtualbox-ose-additions багц эсвэл портыг FreeBSD зочин дээр суулгана. Энэ нь портыг суулгах болно:

```
# cd /usr/ports/emulators/virtualbox-ose-additions && make install clean
```

Дараах мөрүүдийг /etc/rc.conf-д нэмнэ:

```
vboxguest_enable="YES"
vboxservice_enable="YES"
```

Хэрэв ntpd(8) эсвэл ntpdate(8) ашиглагдах бол хостын цагийн синхрончлолыг болиулах ёстой:

```
vboxservice_flags="--disable-timesync"
```

`vboxvideo` драйверыг `Xorg -configure` автоматаар таних ёстой. Хэрэв үгүй бол VirtualBox™ видео картын хувьд /etc/X11/xorg.conf-г өөрчлөөрөй:

```
Section "Device"
    ### Available Driver options are:-
    ### Values: <i>: integer, <f>: float, <bool>: "True"/"False",
    ### <string>: "String", <freq>: "<f> Hz/kHz/MHz"
    ### [arg]: arg optional
```

```
    Identifier "Card0"
    Driver "vboxvideo"
    VendorName "InnoTek Systemberatung GmbH"
    BoardName "VirtualBox Graphics Adapter"
    BusID "PCI:0:2:0"
EndSection
```

vboxmouse драйверыг ашиглахын тулд /etc/X11/xorg.conf дахь хулгахын хэсгийг тохируулаарай:

```
Section "InputDevice"
    Identifier "Mouse0"
    Driver "vboxmouse"
EndSection
```

HAL хэрэглэгчид дараах /usr/local/etc/hal/fdi/policy/90-vboxguest.fdi файлыг үүсгэх юм уу эсвэл /usr/local/shared/hal/fdi/policy/10osvendor/90-vboxguest.fdi файлаас хуулж үүсгэх шаардлагатай:

```
<?xml version="1.0" encoding="utf-8"?>
<!--
# Sun VirtualBox
# Hal driver description for the vboxmouse driver
# $Id: chapter.xml,v 1.33 2012-03-17 04:53:52 eadler Exp $

    Copyright (C) 2008-2009 Sun Microsystems, Inc.

    This file is part of VirtualBox Open Source Edition (OSE, as
    available from http://www.virtualbox.org. This file is free software;
    you can redistribute it and/or modify it under the terms of the GNU
    General Public License (GPL) as published by the Free Software
    Foundation, in version 2 as it comes in the "COPYING" file of the
    VirtualBox OSE distribution. VirtualBox OSE is distributed in the
    hope that it will be useful, but WITHOUT ANY WARRANTY of any kind.

    Please contact Sun Microsystems, Inc., 4150 Network Circle, Santa
    Clara, CA 95054 USA or visit http://www.sun.com if you need
    additional information or have any questions.
-->
<deviceinfo version="0.2">
  <device>
    <match key="info.subsystem" string="pci">
      <match key="info.product" string="VirtualBox guest Service">
        <append key="info.capabilities" type="strlist">input</append>
    <append key="info.capabilities" type="strlist">input.mouse</append>
        <merge key="input.x11_driver" type="string">vboxmouse</merge>
    <merge key="input.device" type="string">/dev/vboxguest</merge>
      </match>
```

```
    </match>
  </device>
</deviceinfo>
```

## 23.3. FreeBSD-г хост буюу эх маягаар ашиглах

VirtualBox™ нь идэвхтэйгээр хөгжүүлэгдэж байгаа бүрэн хэмжээний виртуалчлалын багц бөгөөд Windows®, Mac OS®, Linux® ба FreeBSD зэрэг ихэнх үйлдлийн системүүдийн хувьд байдаг. Энэ нь Windows® эсвэл UNIX® төст зочин үйлдлийн системүүдийг адил хэмжээгээр ажиллуулах чадвартай юм. Энэ нь нээлттэй эхийн тоног төхөөрөмж хэлбэрээр байдаг боловч хаалттай хэсгүүд нь нэмэлт өргөтгөл хэлбэрээр байдаг. Эдгээр хэсгүүдэд USB 2.0 төхөөрөмжүүдийн дэмжлэг байдаг. Дэлгэрэнгүй мэдээллийг VirtualBox™-ийн викигийн "Downloads" хуудас http://www.virtualbox.org/wiki/Downloads-с олж болно. Одоогоор эдгээр өргөтгөлүүд FreeBSD-н хувьд байхгүй.

### 23.3.1. VirtualBox™ суулгах нь

VirtualBox™ нь FreeBSD-ийн багц эсвэл порт хэлбэрээр emulators/virtualbox-ose санд байрладаг. Дараах тушаалыг ашиглан портыг суулгаж болно:

```
# cd /usr/ports/emulators/virtualbox-ose
# make install clean
```

Тохиргооны цэс дээрх сонголтуудаас ашигтай нэг нь `GuestAdditions` буюу зочин нэмэлтүүд програмууд юм. Эдгээр нь хулганын заагчийн нэгдэл (хулганыг эх болон зочин үйлдлийн системийн хооронд гарын тусгай хослол дарж шилжих шаардлагагүйгээр хуваалцан хэрэглэх боломжийг бүрдүүлдэг) болон ялангуяа Windows® зочдын хувьд илүү хурдан видео харуулах зэрэг зочин үйлдлийн системүүдийг хэд хэдэн ашигтай боломжуудаар хангадаг. Зочны нэмэлтүүд нь зочинг суулгаж дууссаны дараа **Devices** цэсэнд байх болно.

VirtualBox™-ийг анх удаа эхлүүлэхээсээ өмнө хэдэн тохиргооны өөрчлөлтүүдийг хийх ёстой. Порт цөмийн модулийг /boot/modules санд суулгах бөгөөд ажиллаж байгаа цөмд дараах тушаалаар дуудах ёстой:

```
# kldload vboxdrv
```

Модулийг дахин ачаалсны дараа үргэлж дуудагддаг байхаар тохируулахын тулд дараах мөрийг /boot/loader.conf файлд нэмнэ:

```
vboxdrv_load="YES"
```

Гүүр болон зөвхөн хостын сүлжээний боломжийг олгох цөмийн модулийг ашиглахын тулд дараах мөрийг /etc/rc.conf файлд нэмж компьютерийг дахин ачаалах хэрэгтэй:

```
vboxnet_enable="YES"
```

`vboxusers` бүлэг VirtualBox™-г суулгах явцад үүсгэгддэг. VirtualBox™-д хандах шаардлагатай бүх хэрэглэгчид энэ бүлгийн гишүүн болж нэмэгдсэн байх ёстой. `pw` тушаалыг шинэ гишүүдийг нэмэхэд хэрэглэж болно:

```
# pw groupmod vboxusers -m yourusername
```

/dev/vboxnetctl-н анхдагч зөвшөөрлүүд нь хязгаарлагдмал байдаг бөгөөд гүүр сүлжээний хувьд өөрчлөх шаардлагатай байдаг:

```
# chown root:vboxusers /dev/vboxnetctl
# chmod 0660 /dev/vboxnetctl
```

Зөвшөөрлүүдийн өөрчлөлтийг байнгын болгохын тулд эдгээр мөрүүдийг /etc/devfs.conf файлд нэмэх хэрэгтэй:

```
own     vboxnetctl root:vboxusers
perm    vboxnetctl 0660
```

VirtualBox™-ийг ажиллуулахын тулд Xorg дээрээс дараахийг бичиж ажиллуулах хэрэгтэй:

```
% VirtualBox
```

VirtualBox™-ийг тохируулж ашиглах талаар дэлгэрэнгүй мэдээллийг http://www.virtualbox.org албан ёсны вэб сайтад хандаж үзнэ үү. FreeBSD-тэй холбоотой мэдээлэл болон алдаог олж засварлах зааврууды FreeBSD-ийн вики http://wiki.FreeBSD.org/VirtualBox дэх тохирох хуудсанд хандаж үзнэ үү.

### 23.3.2. VirtualBox™ USB дэмжлэг

USB төхөөрөмжөөс унших болон түүн рүү бичихийн тулд хэрэглэгчид `operator` бүлгийн гишүүн байх ёстой:

```
# pw groupmod operator -m jerry
```

Дараа нь дараахийг /etc/devfs.rules файлд нэмэх юм уу эсвэл байхгүй бол үүсгээрэй:

```
[system=10]
add path 'usb/*' mode 0660 group operator
```

Эдгээр шинэ дүрмүүдийг ачаалахын тулд дараах мөрийг /etc/rc.conf файлд нэмнэ:

```
devfs_system_ruleset="system"
```

Дараа нь devfs-г дахин ачаална:

```
# service devfs restart
```

USB-г одоо зочин үйлдлийн систем дээр идэвхжүүлж болно. USB төхөөрөмжүүд нь VirtualBox™-н тохиргоон дээр харагдаж байх ёстой.

### 23.3.3. VirtualBox™ хостын DVD/CD хандалт

Хостын DVD/CD хөтчүүдэд зочин системээс хандахдаа физик хөтчийг хуваалцан хэрэглэх замаар ханддаг. VirtualBox™ дотор үүнийг виртуал машины Settings дэх Storage цонхноос тохируулдаг. Шаардлагатай бол эхлээд хоосон IDE CD/DVD төхөөрөмж үүсгэнэ. Дараа нь виртуал CD/DVD хөтөч сонгохын тулд гарч ирэх цэснээс хостын хөтчийг сонгоно. `Passthrough` гэж хаяглагдсан сонгох боломжтой хайрцаг гарч ирнэ. Энэ нь виртуал машиныг тоног төхөөрөмж шууд ашиглах боломжтой болгодог. Жишээ нь дууны CD эсвэл хэрэв сонгогдсон бол зөвхөн шарах боломж гэх мэтийг дурдаж болно.

VirtualBox™ DVD/CD-г ажиллуухын тулд HAL-г ажиллуулах ёстой бөгөөд ингэхийн тулд /etc/rc.conf-д түүнийг идэвхжүүлж хэрэв ажиллаагүй байгаа бол эхлүүлэх хэрэгтэй:

```
hald_enable="YES"
```

```
# service hald start
```

VirtualBox™ DVD/CD-г хэрэглэгч ашиглахын тулд тэдэнд /dev/xpt0, /dev/cdN, болон /dev/passN файлуудад хандах зөвшөөрөл хэрэгтэй. Үүнийг хийхдээ хэрэглэгчийг `operator` бүлэг рүү нэмж шийднэ. Эдгээр төхөөрөмжүүдийн зөвшөөрлүүдийг засахдаа дараах мөрүүдийг /etc/devfs.conf файлд нэмж хийж өгнө:

```
perm cd* 0600
perm xpt0 0660
perm pass* 0660
```

```
# service devfs restart
```

# Chapter 24. Локалчлал - I18N/L10N-ийн хэрэглээ болон тохируулга

## 24.1. Ерөнхий агуулга

FreeBSD нь дэлхий даяар байрласан хэрэглэгчид болон хувь нэмэр болгон оруулагчидтай маш тархмал төсөл юм. Энэ бүлэг Англиар ярьдаггүй хэрэглэгчдэд жинхэнэ ажлаа хийх боломж олгох FreeBSD-ийн интернационалчлал болон локалчлалын боломжуудыг хэлэлцэх болно. Систем болон програмын түвшингүүдийн аль алинд нь i18n шийдлийн олон үзэл бодлууд байдаг бөгөөд бид шаардлагатай тохиолдолд илүү тусгайлсан баримтын эхүүдийг хэрэглэгчдэд зааж өгөх болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Орчин үеийн үйлдлийн системүүд дээр хэлнүүд болон локалууд (locales) нь хэрхэн кодчилогддог талаар.
- Өөрийн нэвтрэх бүрхүүлийн хувьд локалыг хэрхэн тохируулах талаар.
- Англи биш хэлнүүдийн хувьд өөрийн консолыг хэрхэн тохируулах талаар.
- X Цонхны Системийг өөр хэлнүүдтэй хэрхэн үр дүнтэйгээр ашиглах талаар.
- i18n-нийцтэй програмуудыг бичих талаар илүү мэдээллийг хаанаас олох талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- Нэмэлт гуравдагч этгээдийн програмуудыг (Програм суулгах. Багцууд болон портууд) хэрхэн суулгах талаар.

## 24.2. Үндсүүд

### 24.2.1. I18N/L10N гэж юу вэ?

Хөгжүүлэгчид интернационалчлалыг I18N гэж internationalization гэсэн үгийн эхний болон эцсийн үсгийн хоорондох үсгийн тоог оруулан богиносгосон юм. L10N нь бас адил журмаар "localization"-с улбаалан нэрлэгдсэн юм. Хоёулаа нэгдсэн I18N/L10N аргууд, протоколууд болон програмууд нь хэрэглэгчдэд өөрсдийнх нь сонгосон хэлнүүдийг ашиглах боломжийг олгодог.

I18N програмууд нь сангуудын доорх I18N хэрэгслүүдийг ашиглан програмчлагдсан байдаг. Энэ нь энгийн файлыг бичих болон харуулагдсан цэснүүд, текстүүдийг хэл бүр уруу орчуулах боломжийг хөгжүүлэгчдэд олгоно. Бид энэ дадлыг мөрдөхийг програм зохиогчдоос шаргуу хүсдэг.

### 24.2.2. I18N/L10N-ийг би яагаад ашиглах ёстой гэж?

I18N/L10N нь Англи хэлнээс өөр хэл дээр өгөгдлийг харах, оруулах, эсвэл боловсруулахыг таныг хүсэх үед хэрэглэгддэг.

### 24.2.3. I18N чармайлтад ямар хэлнүүд дэмжигдсэн байдаг вэ?

I18N болон L10N нь зөвхөн FreeBSD зориулагдаагүй. Одоогоор хэрэглэгч дэлхийн гол гол хэлнүүдийн ихэнхийг сонгож болох бөгөөд гэхдээ энэ нь энд байгаа жагсаалтаар хязгаарлагдахгүй: Хятад, Герман, Япон, Солонгос, Франц, Орос, Вьетнам болон бусад хэлнүүдийг дурдаж болно.

## 24.3. Локалчлалыг ашиглах нь

Өөрийн бүх сүр жавхлангаараа I18N нь зөвхөн FreeBSD-д зориулагдаагүй бөгөөд энэ нь ёс заншил болсон байдаг. Энэ ёс заншлыг дагаж FreeBSD-д туслахыг бид танаас хүсдэг.

Локалчлалын тохируулгууд нь гурван гол ойлголт дээр тулгуурладаг: Хэлний код, Улсын код ба Кодчилол. Локалын нэрс эдгээр хэсгүүдээс дараах маягаар бүрэлдэн тогтоно:

```
LanguageCode_CountryCode.Encoding
```

### 24.3.1. Хэл болон Улсын кодууд

FreeBSD системийг тухайн хэл уруу (эсвэл I18N дэмждэг бусад UNIX® төст системүүд дээр) локалчлахын тулд хэрэглэгч тухайн улс болон хэлийг (улсын код нь програмд өгөгдсөн хэлний аль хувилбарыг ашиглахыг хэлж өгдөг) заах кодуудыг олж мэдэх хэрэгтэй. Мөн вэб хөтчүүд, SMTP/POP серверүүд зэрэг нь тэдгээр дээр тулгуурлан шийдэл гаргадаг. Дараах нь хэл/улсын кодны жишээнүүд юм:

| Хэл/Улсын код | Тайлбар |
|---|---|
| en_US | English буюу Англи - Нэгдсэн Улс |
| ru_RU | Орос улсад зориулсан Russian буюу Орос |
| zh_TW | Тайваньд зориулсан Уламжлалт Хятад хэл |

Байгаа локалчлалын жагсаалтыг дараах тушаалаар авч болно:

```
% locale -a
```

### 24.3.2. Кодчилолууд

Зарим хэлнүүд 8-бит, өргөн эсвэл олон байт тэмдэгтүүд зэрэг ASCII биш кодчилолыг ашигладаг. Олон байт тэмдэгтүүдийн талаар илүү дэлгэрэнгүйг multibyte(3)-ээс үзнэ үү. Хуучин програмууд тэдгээрийг танидаггүй бөгөөд тэдгээрийг хянах тэмдэгтүүд гэж алддаг. Шинэ програмууд ихэвчлэн 8-бит тэмдэгтүүдийг танидаг. Шийдлээс хамаараад хэрэглэгчид програмыг өргөн эсвэл олон байт тэмдэгтийн дэмжлэгтэйгээр эмхэтгэх эсвэл зөвөөр тохируулах шаардлагатай байж болох юм. Өргөн эсвэл олон тэмдэгтүүдийг оруулж процесс хийж чадахын тулд FreeBSD портын цуглуулга хэл бүрийг өөр өөр програмуудтайгаар хангадаг. FreeBSD порт дахь харгалзах I18N баримтжуулалтаас лавлана уу.

Ялангуяа програмыг хэрхэн зөв тохируулах эсвэл зөв уттуудыг configure/Makefile/эмхэтгэгчид дамжуулахыг шийдэхдээ програмын баримтжуулалтаас харах хэрэгтэй болно.

Санаж байх ёстой зарим нэг зүйлүүд нь:

- Тухайн хэлний ганц C chars тэмдэгтийн олонлогууд (multibyte(3)-г үзнэ үү), өөрөөр хэлбэл ISO8859-1, ISO8859-15, KOI8-R, CP437.
- Өргөн эсвэл олон байт кодчилол, өөрөөр хэлбэл EUC, Big5.

Та тэмдэгтийн олонлогуудын идэвхтэй жагсаалтыг IANA Registry-с шалгаж болно.

FreeBSD нь X11-нийцтэй локалын кодчилолуудыг харин ашигладаг.

### 24.3.3. I18N програмууд

FreeBSD-ийн портууд болон багцын системд I18N програмууд нь амархан танигдахын тулд нэрэндээ `I18N`-тайгаар нэрлэгддэг. Гэхдээ тэдгээр нь хэрэгтэй хэлийг үргэлж дэмжсэн байдаггүй.

### 24.3.4. Локалыг тохируулах

Нэвтрэх бүрхүүл дээр локалын нэрний утгыг `LANG` уруу экспорт хийхэд ихэвчлэн хангалттай байдаг. Үүнийг хэрэглэгчийн ~/.login_conf файл эсвэл хэрэглэгчийн бүрхүүлийн эхлүүлэх файлд (~/.profile, ~/.bashrc, ~/.cshrc) зааж өгч болно. `LC_CTYPE`, `LC_CTIME` зэрэг локал дэд олонлогуудыг тохируулах хэрэггүй. Дэлгэрэнгүй мэдээллийн талаар тухайн хэлний FreeBSD-ийн баримтаас лавлана уу.

Та өөрийн тохиргооны файлуудаа дараах хоёр орчны хувьсагчийг тохируулах хэрэгтэй:

- POSIX® setlocale(3) төрлийн функцуудад зориулсан `LANG`
- Програмуудын MIME тэмдэгтийн олонлогт зориулсан `MM_CHARSET`

Энэ нь хэрэглэгчийн бүрхүүлийн тохиргоо, тухайн програмын тохиргоо болон X11-ийн тохиргоог агуулдаг.

#### 24.3.4.1. Локал аргуудыг тохируулах

Локалыг тохируулах хоёр арга байдаг бөгөөд хоёуланг доор тайлбарласан байгаа. Эхнийхийг (бидний зөвлөж байгааг) нэвтрэх ангилалд орчны хувьсагчуудыг зааж хоёр дахийг системийн бүрхүүлийн эхлүүлэх файлд орчны хувьсагчийн заалтуудыг нэмснээр хийдэг.

##### 24.3.4.1.1. Нэвтрэх ангилалуудын арга

Энэ арга нь бүрхүүл бүрийн эхлүүлэх файлд тухайн бүрхүүлийн заалтуудыг нэмэхийн оронд локалын нэр болон MIME тэмдэгтийн олонлогуудад хэрэгтэй орчны хувьсагчуудыг боломжит бүрхүүл бүрийн хувьд нэг л удаа заах боломжийг олгодог. Хэрэглэгчийн түвшний тохируулгыг хэрэглэгч өөрөө хийж болох бөгөөд Администраторын түвшний тохируулга нь супер хэрэглэгчийн зөвшөөрлүүдийг шаарддаг.

#### 24.3.4.1.1.1. Хэрэглэгчийн түвшний тохируулга

Энд хэрэглэгчийн гэр сан дотор байрлах Latin-1 кодчилолын хувьд хоёр хувьсагчийг нь тохируулж өгсөн .login_conf файлын хамгийн бага жишээ байна:

```
me:\
    :charset=ISO-8859-1:\
    :lang=de_DE.ISO8859-1:
```

Энд BIG-5 кодчилолд Уламжлалт Хятад хэлд зориулан хувьсагчуудыг тохируулж байгаа .login_conf файлын жишээ байна. Зарим програм хангамжууд нь Хятад, Япон болон Солонгос хэлний хувьд локалын хувьсагчуудыг зөвөөр хүндэлдэггүй учраас илүү олон хувьсагчуудыг тохируулсныг эндээс харж болно.

```
#Users who do not wish to use monetary units or time formats
#of Taiwan can manually change each variable
me:\
    :lang=zh_TW.Big5:\
    :setenv=LC_ALL=zh_TW.Big5:\
    :setenv=LC_COLLATE=zh_TW.Big5:\
    :setenv=LC_CTYPE=zh_TW.Big5:\
    :setenv=LC_MESSAGES=zh_TW.Big5:\
    :setenv=LC_MONETARY=zh_TW.Big5:\
    :setenv=LC_NUMERIC=zh_TW.Big5:\
    :setenv=LC_TIME=zh_TW.Big5:\
    :charset=big5:\
    :xmodifiers="@im=gcin": #Set gcin as the XIM Input Server
```

Илүү дэлгэрэнгүйг Администраторын түвшний тохируулга болон login.conf(5)-с үзнэ үү.

#### 24.3.4.1.2. Администраторын түвшний тохируулга

/etc/login.conf дахь хэрэглэгчийн нэвтрэх ангилал зөв хэл тохируулсныг шалгана. Эдгээр тохируулгууд /etc/login.conf-д байгаа эсэхийг шалгаарай:

```
language_name:Account Type Description:\
    :charset=MIME_charset:\
    :lang=locale_name:\
    :tc=default:
```

Бидний урдны жишээний адил Latin-1-г сонговол энэ нь иймэрхүү харагдана:

```
german|German Users Accounts:\
    :charset=ISO-8859-1:\
    :lang=de_DE.ISO8859-1:\
    :tc=default:
```

Хэрэглэгчийн нэвтрэх ангиллуудыг өөрчлөхөөсөө өмнө дараах тушаалыг ажиллуулж

```
# cap_mkdb /etc/login.conf
```

шинэ тохиргоог системд харагдуулахаар /etc/login.conf-д хийнэ.

#### 24.3.4.1.3. Нэвтрэх ангиллуудыг vipw(8) ашиглан солих

Шинэ хэрэглэгчид нэмэхийн тулд `vipw` тушаал ашиглан оруулгыг иймэрхүү болгоно:

```
user:password:1111:11:language:0:0:User Name:/home/user:/bin/sh
```

#### 24.3.4.1.4. Нэвтрэх ангиллуудыг adduser(8) ашиглан солих

Шинэ хэрэглэгчид нэмэхийн тулд `adduser` тушаал ашиглан доор дурдсаныг хийнэ:

- /etc/adduser.conf файлд `defaultclass = language`-г тохируулна. Та энэ тохиолдолд бусад хэлнүүдийн бүх хэрэглэгчдэд зориулан `анхдагч` ангилалыг оруулах хэрэгтэйг санах хэрэгтэй.
- adduser(8)-с гарч ирэх асуулт бүрт

  ```
  Enter login class: default []:
  ```

  асуултад тухайн хэлийг зааж өгөх бас нэг хувилбар байж болно.

- Өөр нэг арга нь нэмэхээр хүсэж байгаа өөр хэлний хэрэглэгч бүрийн хувьд доор дурдсаныг ашиглах явдал юм:

  ```
  # adduser -class language
  ```

#### 24.3.4.1.5. Нэвтрэх ангиллуудыг pw(8) ашиглан солих

Хэрэв та pw(8)-г шинэ хэрэглэгч нэмэхийн тулд ашиглаж байгаа бол үүнийг иймэрхүү маягаар дуудаарай:

```
# pw useradd user_name -L language
```

#### 24.3.4.1.6. Бүрхүүлийн эхлүүлэх файл арга

Энэ аргыг зөвлөдөггүй, учир нь сонгосон боломжит бүрхүүл програм бүрийн хувьд өөр тохируулгыг шаарддаг. Оронд нь Нэвтрэх ангиллын аргыг ашиглаарай.

Локалын нэр болон MIME тэмдэгтийн олонлогийг нэмэхийн тулд доор үзүүлсэн хоёр орчны

хувьсагчийг /etc/profile-д болон/эсвэл бүрхүүлийн эхлүүлэх файл /etc/csh.login-д зааж өгнө. Бид Герман хэлийг доор жишээ болгон ашиглах болно:

/etc/profile файлд:

```
LANG=de_DE.ISO8859-1; export LANG
MM_CHARSET=ISO-8859-1; export MM_CHARSET
```

Эсвэл /etc/csh.login файлд:

```
setenv LANG de_DE.ISO8859-1
setenv MM_CHARSET ISO-8859-1
```

Өөрөөр та дээрх зааврууд ыг /usr/shared/skel/dot.profile файлд (дээрх /etc/profile-д ашигласантай адил) эсвэл /usr/shared/skel/dot.login файлд (дээрх /etc/csh.login-д ашигласантай адил) нэмж болно.

X11-ийн хувьд:

$HOME/.xinitrc файлд:

```
LANG=de_DE.ISO8859-1; export LANG
```

Эсвэл:

```
setenv LANG de_DE.ISO8859-1
```

Таны бүрхүүлээс хамаараад (дээр дурдсаныг үзнэ үү).

### 24.3.5. Консол тохируулах

Бүх ганц C chars тэмдэгтийн олонлогуудын хувьд зөв консолын фонтуудыг /etc/rc.conf-д асуултанд байгаа хэлний хувьд тохируулна:

```
font8x16=font_name
font8x14=font_name
font8x8=font_name
```

Эндэх *font_name*-ийг /usr/shared/syscons/fonts сангаас .fnt төгсгөлгүйгээр авсан.

Хэрэв шаардлагатай бол keymap болон screenmap-ийг өөрийн ганц C chars тэмдэгтийн олонлогийн хувьд `sysinstall` ашиглан зааж өгнө. sysinstall дотор байхдаа Configure-г сонгоод дараа нь Console-г сонгоно. Өөрөөр та доор дурдсаныг /etc/rc.conf-д нэмж болно:

```
scrnmap=screenmap_name
keymap=keymap_name
keychange="fkey_number sequence"
```

Эндэх *screenmap_name*-ийг /usr/shared/syscons/scrnmaps сангаас .scm төгсгөлгүйгээр авсан. Зохих оноогдсон фонттой screenmap нь псевдографик талбарт VGA адаптерийн фонтын тэмдэгтийн матриц дээр бит 8-ийг бит 9 уруу өргөтгөхөд ихэвчлэн тойрон гарах арга зам болгон хэрэглэгддэг, өөрөөр хэлбэл хэрэв дэлгэцийн фонт бит 8 багана ашиглаж байвал тэр талбараас үсэгнүүдийг шилжүүлэн гаргахын тулд хэрэглэгддэг.

Хэрэв та /etc/rc.conf-д moused дэмонг идэвхжүүлэхийг дараах байдлаар тохируулсан бол:

```
moused_enable="YES"
```

хулганы заагчийн мэдээллийг дараагийн хэсгээс шалгаарай.

Анхдагчаар syscons(4) драйверийн хулганы заагч нь тэмдэгтийн олонлогийн 0xd0-0xd3 хэсгийг эзэлдэг. Хэрэв таны хэл энэ хэсгийг ашигладаг бол заагчийн хэсгийг үүнээс гадна шилжүүлэх хэрэгтэй. FreeBSD-ийн хувьд тойрон гарах арга замыг идэвхжүүлэхийн тулд дараах мөрийг /etc/rc.conf-д нэмнэ:

```
mousechar_start=3
```

Эндэх *keymap_name*-ийг /usr/shared/syscons/keymaps сангаас .kbd төгсгөлгүй авсан. Хэрэв та аль keymap-ийг ашиглах эсэхдээ эргэлзэж байвал kbdmap(1)-г ашиглаж дахин ачаалалгүйгээр keymap-уудыг тест хийж болно.

`keychange` нь функцын товчлууруудыг сонгосон терминалын төрөлтэй тааруулахын тулд програмчлахад ихэвчлэн хэрэгтэй байдаг, учир нь функцын товчлуурын дарааллуудыг товчлууруудын оноолтод тодорхойлж болдоггүй.

Бас консолын терминалын зөв төрлийг бүх `ttyv*` оруулгуудын хувьд /etc/ttys файлд тохируулсан эсэхээ шалгаарай. Одоогийн урьдчилан тодорхойлсон зохицлууд нь:

| **Тэмдэгтийн Олонлог** | **Терминалын Төрөл** |
|---|---|
| ISO8859-1 эсвэл ISO8859-15 | `cons25l1` |
| ISO8859-2 | `cons25l2` |
| ISO8859-7 | `cons25l7` |
| KOI8-R | `cons25r` |
| KOI8-U | `cons25u` |
| CP437 (VGA default) | `cons25` |
| US-ASCII | `cons25w` |

Өргөн эсвэл олон байт тэмдэгтүүд бүхий хэлнүүдийн хувьд өөрийн /usr/ports/language сангаас зөв FreeBSD портоо ашиглаарай. Зарим портууд консол маягаар байхад түүнийг систем сериал vtty-үүд шигээр хардаг, ийм учраас та X11 болон псевдо-сериал консолын хувьд хангалттай vtty-үүдийг хадгалж байх шаардлагатай. Өөр хэлүүдийг консолд ашиглахын тулд энд програмуудын хэсэгчилсэн жагсаалт байна:

| Хэл | Байрлал |
| --- | --- |
| Уламжлалт Хятад хэл (BIG-5) | chinese/big5con |
| Япон хэл | japanese/kon2-16dot эсвэл japanese/mule-freewnn |
| Солонгос хэл | korean/han |

## 24.3.6. X11 тохируулах

X11 нь FreeBSD Төслийн нэг хэсэг биш боловч бид энд FreeBSD хэрэглэгчдэд зориулж зарим мэдээлэл орууллаа. Илүү дэлгэрэнгүйг Xorg вэб хаягаас эсвэл аль X11 сервер ашиглаж байгаа түүнийхээ вэб хаягаас лавлана уу.

~/.Xresources файлд та програмуудад зориулсан I18N тохируулгуудыг (өөрөөр хэлбэл фонтууд, цэснүүд, гэх мэт.) тааруулж өгч болно.

### 24.3.6.1. Фонтуудыг үзүүлэх

Xorg сервер (x11-servers/xorg-server) суулгаад дараа нь хэлний TrueType® фонтуудыг суулгана. Зөв локалыг тохируулах нь танд цэснүүдийг сонгосон хэлээрээ харах зэрэг боломжийг олгоно.

### 24.3.6.2. Англи биш тэмдэгтүүдийг оруулах нь

X11 оруулах арга (XIM) нь бүх X11 клиентүүдэд зориулсан шинэ стандарт юм. Бүх X11 програмууд нь XIM оруулах серверүүдээс ролт авдаг XIM клиентүүд маягаар бичигдэх ёстой. Өөр өөр хэлнүүдэд зориулагдсан хэд хэдэн XIM серверүүд байдаг.

## 24.3.7. Хэвлэгч тохируулах

Зарим ганц C chars тэмдэгтийн олонлогууд нь ихэвчлэн хэвлэгчүүд дотор хатуугаар бичигдсэн байдаг. Өргөн эсвэл олон байт тэмдэгтийн олонлогууд нь тусгай тохируулга шаарддаг бөгөөд бид apsfilter програмыг хэрэглэхийг зөвлөдөг. Та бас PostScript® эсвэл PDF хэлбэрүүд уруу тухайн хэлний хөрвүүлэгчдийг ашиглан баримтыг хөрвүүлж болох юм.

## 24.3.8. Цөм ба файлын системүүд

FreeBSD-ийн хурдан файлын систем (FFS) нь 8-бит цэвэр учир үүнийг ямар ч ганц C chars тэмдэгтийн олонлогтой цуг ашиглаж болно (multibyte(3)-г үзнэ үү), гэхдээ тэмдэгтийн олонлогийн нэр файлын системд хадгалагддаггүй; өөрөөр хэлбэл энэ нь түүхий 8-бит бөгөөд кодчилолын дарааллын талаар юу ч мэддэггүй. Албан ёсоор FFS нь өргөн эсвэл олон байт тэмдэгтийн олонлогуудын аль ч хэлбэрийг дэмждэггүй. Гэхдээ зарим өргөн эсвэл олон байт тэмдэгтийн олонлогууд нь FFS-д ийм дэмжлэгийг идэвхжүүлэхэд зориулсан бие биеэсээ

ангид засваруудтай байдаг. Тэдгээр нь цорын ганц ийш тийш хөрвүүлэх боломжгүй шийдлүүд буюу hack бөгөөд бид тэдгээрийг эх модонд оруулахгүй гэж шийдэцгээсэн юм. Тохирох хэлнүүдийн вэб хуудаснуудаас илүү дэлгэрэнгүй мэдээлэл болон засвар файлуудын талаар лавлана уу.

FreeBSD MS-DOS® файлын систем нь MS-DOS®, Юникод тэмдэгтийн олонлогууд болон FreeBSD-ийн сонгосон файлын системийн тэмдэгтийн олонлогуудын хооронд хөрвүүлэх тохируулж болох боломж бүхий байдаг. Дэлгэрэнгүйг mount_msdosfs(8)-с үзнэ үү.

# 24.4. I18N програмуудыг эмхэтгэх

FreeBSD-ийн олон портууд I18N дэмжлэгтэйгээр хөрвүүлэгдсэн байдаг. Тэдгээрийн зарим нь портын нэрэндээ -I18N гэж тэмдэглэгдсэн байдаг. Эдгээр болон бусад олон програмууд I18N-д зориулагдсан дэмжлэгтэйгээр бүтээгдсэн байдаг бөгөөд тусгай хэлэлцүүлэг шаардлагагүй юм.

Гэхдээ MySQL зэрэг зарим програмууд тусгайлсан charset бүхий байхаар тохируулагдсан Makefile файлтай байх шаардлагатай. Үүнийг ихэвчлэн Makefile-д эсвэл эхэд байх configure програмд утга дамжуулан хийдэг.

# 24.5. FreeBSD-г өөр хэлнүүд уруу локалчлах

## 24.5.1. Орос хэл (KOI8-R Кодчилол)

KOI8-R кодчилолын талаар дэлгэрэнгүй мэдээллийг KOI8-R Лавлахууд (Russian Net Character Set)-с үзнэ үү.

### 24.5.1.1. Локал Тохируулах

Дараах мөрүүдийг өөрийн ~/.login_conf файлд нэмнэ:

```
me:My Account:\
    :charset=KOI8-R:\
    :lang=ru_RU.KOI8-R:
```

Энэ бүлгийн өмнөхөөс локал тохируулах жишээнүүдийг үзнэ үү.

### 24.5.1.2. Консол Тохируулга

- Дараах мөрийг өөрийн /etc/rc.conf файлд нэмнэ:

  ```
  mousechar_start=3
  ```

- Мөн дараах тохируулгуудыг /etc/rc.conf-д ашиглана:

  ```
  keymap="ru.utf-8"
  ```

```
scrnmap="utf-82cp866"
font8x16="cp866b-8x16"
font8x14="cp866-8x14"
font8x8="cp866-8x8"
```

- /etc/ttys файл дахь `ttyv*` оруулга болгоны хувьд `cons25r`-ийг терминалын төрөлд ашиглана.

Энэ бүлгийн өмнөхөөс консол тохируулах жишээнүүдийг үзнэ үү.

#### 24.5.1.3. Хэвлэгчийн Тохируулга

Орос тэмдэгтүүдтэй ихэнх хэвлэгчид CP866 гэсэн тоног төхөөрөмжийн код хуудастай ирдэг бөгөөд KOI8-R-с CP866 уруу хөрвүүлэхийн тулд тусгай гаралтын шүүгч хэрэгтэй болдог. Ийм шүүгч анхдагчаар /usr/libexec/lpr/ru/koi2alt гэж суугддаг. Орос хэвлэгчийн /etc/printcap оруулга иймэрхүү харагдах ёстой:

```
lp|Russian local line printer:\
    :sh:of=/usr/libexec/lpr/ru/koi2alt:\
    :lp=/dev/lpt0:sd=/var/spool/output/lpd:lf=/var/log/lpd-errs:
```

Дэлгэрэнгүй тайлбарын талаар printcap(5)-с үзнэ үү.

#### 24.5.1.4. MS-DOS® FS болон Орос файлын нэрс

Дараах жишээ fstab(5) оруулга нь холбогдсон MS-DOS® файлын системүүд дээр Орос файлын нэрийн дэмжлэгийг идэвхжүүлдэг:

```
/dev/ad0s2      /dos/c  msdos   rw,-Wkoi2dos,-Lru_RU.KOI8-R 0 0
```

`-L` тохируулга ашиглагдсан локалын нэрийг сонгодог бөгөөд `-W` нь тэмдэгтийн хөрвүүлэлтийн хүснэгтийг заадаг. `-W` тохируулгыг ашиглахын тулд /usr-г MS-DOS® хуваалтаас өмнө холбох хэрэгтэй, учир нь хөрвүүлэлтийн хүснэгтүүд /usr/libdata/msdosfs-д байрладаг. Дэлгэрэнгүй мэдээллийг mount_msdosfs(8) гарын авлагын хуудаснаас үзнэ үү.

#### 24.5.1.5. X11 тохируулга

1. Эхлээд X биш локалын тохируулгыг тайлбарласнаар хийнэ.
2. Хэрэв та Xorg-г ашиглаж байгаа бол x11-fonts/xorg-fonts-cyrillic багцыг суулгана.

   Өөрийн /etc/X11/xorg.conf файлын `"Files"` хэсгийг шалгаарай. Дараах мөр аль ч `FontPath` оруулгуудаас *өмнө* нэмэгдсэн байх шаардлагатай:

   ```
   FontPath   "/usr/local/lib/X11/fonts/cyrillic"
   ```

   Өөр илүү олон кирилл фонтуудыг авахыг хүсвэл портуудаас үзнэ үү.

3. Орос гарыг идэвхжүүлэхийн тулд өөрийн xorg.conf файлын "Keyboard" хэсэгт доор дурдсаныг нэмээрэй:

```
Option "XkbLayout"   "us,ru"
Option "XkbOptions"  "grp:toggle"
```

Мөн XkbDisable-г хаасан (тайлбар болгосон) эсэхийг тэндээс шалгаарай.

grp:toggle-н хувьд RUS/LAT шилжүүлэгч Right Alt болох бөгөөд grp:ctrl_shift_toggle шилжүүлэгчийн хувьд Ctrl + Shift болно. grp:caps_toggle-н хувьд RUS/LAT шилжүүлэгч CapsLock байна. Хуучин CapsLock функцын хувьд Shift + CapsLock (зөвхөн LAT горимд) байсаар байна. grp:caps_toggle нь Xorg дээр тодорхойгүй шалтгааны улмаас ажилладаггүй.

Хэрэв та өөрийн гар дээрээ "Windows®" товчлууруудтай бол, зарим үсгэн товчлуурууд нь RUS горимд буруугаар тааруулагддагийг анзаарсан байх, өөрийн xorg.conf файлдаа дараах мөрийг нэмээрэй:

```
Option "XkbVariant" ",winkeys"
```

Орос XKB гар нь локалчлал хийгдээгүй програмуудтай ажиллахгүй байж болох юм.

Хамгийн багаар локалчлагдсан програмууд XtSetLanguageProc (NULL, NULL, NULL); функцыг програмын эхэнд дуудах ёстой.

X11 програмуудыг локалчлах талаар дэлгэрэнгүй зааврыг X Цонхны KOI8-R хаягаас үзнэ үү.

### 24.5.2. Тайваньд зориулсан уламжлалт Хятад хэлний локалчлал

FreeBSD-Тайвань Төсөл олон Хятад портуудыг ашиглан FreeBSD-д зориулсан Хятад HOWTO-г http://netlab.cse.yzu.edu.tw/~statue/freebsd/zh-tut/ дээр байрлуулсан байдаг. FreeBSD Хятад HOWTO -н одоогийн засварлагч нь Чуан-Шинг Шен statue@freebsd.sinica.edu.tw юм.

Чуан-Шинг Шен statue@freebsd.sinica.edu.tw нь FreeBSD-Тайваний zh-L10N-tut ашиглан Хятад FreeBSD Цуглуулга (CFC) үүсгэсэн байгаа. Багцууд болон скрипт файлууд ftp://freebsd.csie.nctu.edu.tw/pub/taiwan/CFC/ дээр байгаа болно.

### 24.5.3. Герман хэлний локалчлал(бүх ISO 8859-1 хэлнүүдэд зориулсан)

Славен Резич eserte@cs.tu-berlin.de нь FreeBSD машин дээр umlaut буюу Герман хэлний авиа өөрчлөгддөгийг тэмдэглэдэг хоёр цэгийг хэрхэн ашиглах талаар заавар бичсэн байгаа. Энэ заавар нь Герман хэл дээр бичигдсэн бөгөөд http://user.cs.tu-berlin.de/~eserte/FreeBSD/doc/umlaute/umlaute.html дээр байгаа болно.

### 24.5.4. Грек хэлний локалчлал

Никос Кокалис nickkokkalis@gmail.com нь FreeBSD дээрх Грек хэлний дэмжлэгийн талаар бүрэн нийтлэл бичсэн байгаа. Энэ нь FreeBSD-ийн албан ёсны Грек баримтжуулалтын хэсэг болон http://www.freebsd.org/doc/el/articles/greek-language-support/ хаягт орсон байдаг. Энэ нь *зөвхөн* Грек хэл дээр юм.

### 24.5.5. Япон болон Солонгос хэлний локалчлал

Япон хэлний хувьд http://www.jp.FreeBSD.org/-д хандана уу, Солонгос хэлний хувьд http://www.kr.FreeBSD.org/-д хандана уу.

### 24.5.6. Англи биш FreeBSD баримтжуулалт

FreeBSD-ийн зарим хувь нэмэр оруулагчид FreeBSD-ийн баримтжуулалтын зарим хэсгийг орчуулсан байгаа. Тэдгээрийг эх сайт дээрх холбоосуудаас эсвэл /usr/shared/doc сангаас үзэж болох юм.

# Chapter 25. FreeBSD-г шинэчилж сайжруулах нь

## 25.1. Ерөнхий агуулга

FreeBSD нь өөрийн хувилбаруудын хооронд байнгын хөгжүүлэлтийн доор оршин тогтнож байдаг. Зарим хүмүүс албан ёсоор гаргасан хувилбаруудыг ашиглах хүсэлтэй байдаг бол зарим хүмүүс хамгийн сүүлийн үеийн хөгжүүлэлтийг дагах сонирхолтой байдаг. Гэхдээ албан ёсны хувилбарууд хүртэл аюулгүй байдлын болоод бусад чухал засваруудаар шинэчлэгдэж байдаг. Ямар хувилбар ашиглаж байгаагаас үл хамааражаад FreeBSD нь таны системийг шинэ байлгахад шаардлагатай бүх л хэрэгслүүд болон хувилбар хооронд хялбараар шинэчлэх боломжоор хангадаг. Энэхүү бүлэг нь хөгжүүлэлтийн системийг дагахыг хүсэх эсвэл гаргасан хувилбартай үлдэх эсэхийг шийдэхэд танд туслах болно. Таны системийг шинэчлэхэд зориулсан үндсэн хэрэгслүүдийг бас харуулах болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Систем болон портын цуглуулгыг ямар хэрэгслүүд ашиглан шинэчилж болох талаар.
- freebsd-update, Subversion, CVSup, CVS, эсвэл CTM програмуудын тусламжтай өөрийн системийг хэрхэн хамгийн сүүлийн хэлбэрт авчрах талаар.
- Суулгагдсан системийн төлвийг мэдэгдэж байгаа сайн хуулбартай хэрхэн харьцуулах талаар.
- Subversion эсвэл баримтжуулах порт ашиглан өөрийн баримтуудыг хэрхэн сүүлийн хувилбарт байлгаж байх талаар.
- FreeBSD-STABLE болон FreeBSD-CURRENT хөгжүүлэлтийн салбаруудын ялгаа.
- Бүх үндсэн системийг `make buildworld` (гэх мэт) ашиглан хэрхэн дахин бүтээж суулгах талаар.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдэх шаардлагатай:

- Өөрийн сүлжээний холболтыг зөв тохируулах (Сүлжээний нэмэлт ойлголтууд).
- Нэмэлт гуравдагч програм хангамжуудыг хэрхэн суулгахыг мэдэх (Програм суулгах. Багцууд болон портууд).

Энэ бүлэгт FreeBSD-ийн эхийг авч шинэчлэхийн тулд `svn` тушаалыг ашиглагдсан. Үүнийг хэрэглэхийн тулд devel/subversion гэсэн порт буюу багцыг та суулгах хэрэгтэй.

## 25.2. FreeBSD-ийн шинэчлэлт

Аюулгүй байдлын засваруудыг хийнэ гэдэг компьютерийн програм хангамж ялангуяа үйлдлийн системийг арчлалтын чухал хэсгийн нэг юм. Удаан хугацааны туршид FreeBSD дээр энэ процесс хялбар биш байлаа. Засваруудыг эх код руу хийж кодыг хоёртын хэлбэр рүү дахин бүтээж дараа нь хоёртын файлуудыг дахин суулгах шаардлагатай байлаа.

Энэ нь одоо тийм биш болсон бөгөөд FreeBSD нь `freebsd-update` гэгддэг хэрэгслийг агуулдаг. Энэ хэрэгсэл нь хоёр тусдаа функцээр хангадаг. Нэгдүгээрт энэ нь бүтээлт болон суулгах шаардлагагүйгээр хоёртын аюулгүй байдал болон алдааны шинэчлэлтүүдийг FreeBSD-ийн үндсэн системд оруулах боломжийг олгодог. Хоёрдугаарт уг хэрэгсэл бага болон том хувилбарын шинэчлэлтүүдийг дэмждэг.

Аюулгүй байдлын багаар дэмжигдсэн бүх архитектур болон хувилбаруудын хувьд хоёртын шинэчлэлтүүд байдаг. Шинэ хувилбар руу шинэчлэхээс өмнө хүссэн хувилбарт чинь хамаатай чухал мэдээлэл байж болох учир одоогийн хувилбарын зарлалуудыг дахин үзэх хэрэгтэй. Тэдгээр зарлалуудыг дараах холбоосоос үзэж болно: http://www.FreeBSD.org/releases/.

Хэрэв `crontab` нь `freebsd-update`-ийн боломжуудыг хэрэглэж байвал дараах үйлдлийг эхлэхээс өмнө түүнийг болиулах хэрэгтэй.

### 25.2.1. Тохиргооны файл

Процессийг илүү хянах боломжтой болгож зарим хэрэглэгчид /etc/freebsd-update.conf анхны тохиргооны файлыг өөрчлөхийг хүсэж болох юм. Тохиргоонууд нь маш сайн баримтжуулагдсан байдаг боловч дараах хэдэн зүйлийг арай илүү тайлбарлах шаардлагатай байж болох юм:

```
# Components of the base system which should be kept updated.
Components src world kernel
```

Энэ өгөгдөл нь FreeBSD-ийн аль хэсгийг шинэчлэхийг хянадаг. Анхдагчаар эх код, үндсэн систем бүтнээрээ ба цөмийг шинэчлэх байдаг. Бүрэлдэхүүн хэсгүүд нь суулгах явцад байдагтай адил байдаг бөгөөд жишээ нь `world/games` гэдгийг энд нэмэхэд тоглоомын засваруудыг хийх боломжийг олгоно. `src/bin` гэдгийг ашиглах нь src/bin дахь эх кодыг шинэчлэх боломжийг олгодог.

Хамгийн сайн тохиргоо бол үүнийг анхдагчаар нь үлдээх бөгөөд тусгай зүйлс оруулж үүнийг өөрчилснөөр хэрэглэгчид өөрсдийн шинэчлэхийг хүссэн зүйл болгоноо жагсааж оруулах шаардлагатай болно. Ингэснээр эх код болон хоёртын файлуудын хоорондох уялдаа алдагдаж гамшигт үр дагаварт хүргэж болно.

```
# Paths which start with anything matching an entry in an IgnorePaths
# statement will be ignored.
IgnorePaths
```

/bin эсвэл /sbin зэрэг замуудыг нэмж эдгээр тусгай сангуудыг шинэчлэх процессийн явцад оролдохгүй орхиж болно. Энэ тохиргоо нь `freebsd-update` локал өөрчлөлтүүдийг дарж бичихээс хамгаалахад хэрэглэж болно.

```
# Paths which start with anything matching an entry in an UpdateIfUnmodified
# statement will only be updated if the contents of the file have not been
```

```
# modified by the user (unless changes are merged; see below).
UpdateIfUnmodified /etc/ /var/ /root/ /.cshrc /.profile
```

Заагдсан сангууд дахь тохиргооны файлууд өөрчлөгдөөгүй тохиолдолд шинэчилнэ. Хэрэглэгчийн хийсэн өөрчлөлтүүд эдгээр файлуудын автомат шинэчлэлтийг хүчингүй болгоно. `freebsd-update`-г нийлүүлэх явцад өөрчлөлтүүдийг хадгалахыг тушаах `KeepModifiedMetadata` гэсэн өөр нэг тохиргоо байдаг.

```
# When upgrading to a new FreeBSD release, files which match MergeChanges
# will have any local changes merged into the version from the new release.
MergeChanges /etc/ /var/named/etc/
```

`freebsd-update`-ийн нийлүүлэхийг оролдох тохиргооны файлуудтай сангуудын жагсаалт. Файл нийлүүлэх процесс нь цөөн тохиргоотой mergemaster(8)-тай төстэй diff(1) засварууд бөгөөд нийлүүлэлтийг эсвэл хүлээн авах юм уу эсвэл засварлагч нээж эсвэл `freebsd-update` ажиллагаагаа зогсоох болно. Эргэлзэж байвал /etc санг нөөцөлж аваад нийлүүлэлтүүдийг хүлээн авах хэрэгтэй. `mergemaster` тушаалын талаар дэлгэрэнгүй мэдээллийг `mergemaster`-с үзнэ үү.

```
# Directory in which to store downloaded updates and temporary
# files used by FreeBSD Update.
# WorkDir /var/db/freebsd-update
```

Энэ сан нь бүх засварууд болон түр зуурын файлууд байх сан юм. Хэрэглэгч хувилбар шинэчлэлт хийж байвал энэ байрлал нь хамгийн багаар бодоход гигабайт дискийн зайтай байх шаардлагатай.

```
# When upgrading between releases, should the list of Components be
# read strictly (StrictComponents yes) or merely as a list of components
# which *might* be installed of which FreeBSD Update should figure out
# which actually are installed and upgrade those (StrictComponents no)?
# StrictComponents no
```

`yes` гэж тохируулсан үед `freebsd-update` нь `Components` буюу бүрэлдэхүүн хэсгүүдийн жагсаалт бүрэн гэж тооцох бөгөөд жагсаалтаас гадна өөрчлөлт хийхийг оролдохгүй. `freebsd-update` нь `Components`-ийн жагсаалтад хамаарах файл бүрийг шинэчлэхийг оролдох болно.

### 25.2.2. Аюулгүй байдлын засварууд

Аюулгүй байдлын засварууд нь алсын машин дээр хадгалагддаг бөгөөд дараах тушаал ашиглан татан авч суулгаж болно:

```
# freebsd-update fetch
# freebsd-update install
```

Хэрэв цөмийн ямар нэг засвар хийгдсэн бол системийг дахин ачаалах хэрэгтэй. Хэрэв бүгд зөв явагдсан бол систем нь засвар хийгдсэн байх ёстой бөгөөд `freebsd-update`-ийг cron(8) ажлаас шөнө бүр ажиллуулж болно. Энэ ажлыг хийхэд дараах мөрийг /etc/crontab-д хийхэд хангалттай байх болно:

```
@daily                                  root    freebsd-update cron
```

Энэ мөр нь өдөр бүр `freebsd-update` хэрэгслийг ажиллуулахыг зааж байна. Ийм аргаар `cron` -ий нэмэлт өгөгдлийг ашиглан `freebsd-update` нь зөвхөн шинэчлэлт байгаа эсэхийг шалгах болно. Хэрэв засварууд байвал тэдгээрийг автоматаар диск рүү татаж авах бөгөөд гэхдээ засваруудыг хийхгүй. `root` хэрэглэгч рүү захиа илгээгдэх бөгөөд дараа нь тэд гараар суулгаж болох юм.

Хэрэв ямар нэг зүйл буруу болбол `freebsd-update` нь дараах тушаалаар сүүлийн өөрчлөлтүүдийг буцаах чадвартай байдаг:

```
# freebsd-update rollback
```

Дууссаны дараа цөм эсвэл цөмийн модулиуд өөрчлөгдсөн бол системийг дахин эхлүүлэх хэрэгтэй. Энэ нь шинэ хоёртын файлуудыг санах ой руу дуудах боломжийг FreeBSD-д олгоно.

`freebsd-update` хэрэгсэл нь автоматаар зөвхөн GENERIC цөмийг шинэчилж чаддаг. Хэрэв өөрчлөн тохируулсан цөм ашиглагдаж байвал `freebsd-update` нь шинэчлэлтийг хийж дууссаны дараа цөмийг дахин бүтээж суулгах шаардлагатай. Гэхдээ `freebsd-update` нь /boot/GENERIC (хэрэв байгаа бол) дахь GENERIC цөмийг системийн тухайн үеийн (ажиллаж байгаа) цөм биш байсан ч гэсэн олж шинэчилдэг.

/boot/GENERIC дахь GENERIC цөмийн хуулбарыг үргэлж хадгалах нь ер нь зүйтэй байдаг. Энэ нь төрөл бүрийн асуудлуудыг шинжлэх болон Major ба Minor буюу Том ба Бага шинэчлэлтүүд-т тайлбарласны дагуу `freebsd-update`-г ашиглан хувилбар шинэчлэлтийг хийхэд ач тустай байх болно.

/etc/freebsd-update.conf дахь анхдагч тохиргоо өөрчлөгдөөгүй л бол `freebsd-update` шинэчлэгдсэн цөмийн эхийг бусад шинэчлэлттэй цуг суулгах болно. Дараа нь өөрийн өөрчлөн тохируулсан цөмийг дахин бүтээж суулгахдаа энгийн сурсан аргаараа хийж болно.

`freebsd-update`-аар түгээгдсэн шинэчлэлтүүд нь цөмтэй үргэлж холбоотой байдаггүй. `freebsd-update install` тушаалаар цөмийн эх өөрчлөгдөөгүй бол өөрийн өөрчлөн тохируулсан цөмийг дахин бүтээх шаардлагагүй юм. Гэхдээ `freebsd-update` тушаал /usr/src/sys/conf/newvers.sh файлыг үргэлж шинэчлэх болно. Тухайн үеийн нөхөөсийн түвшинг (`uname -r` тушаалаас гарсан `-p` дугаараар заагдсан) энэ файлаас авдаг. Өөр бусад юмс өөрчлөгдөөгүй байсан ч гэсэн өөрийн өөрчлөн тохируулсан цөмийг дахин бүтээх нь системийн тухайн үеийн нөхөөсийн түвшинг зөв гаргах боломжийг uname(1)-д олгоно. Энэ нь олон системийг арчилж байгаа үед ялангуяа ач тустай байдаг бөгөөд ингэснээр тэр системүүд дээр суулгагдсан шинэчлэлтүүдэд түргэн шуурхай

> үнэлгээ өгөх боломжийг олгодог.

### 25.2.3. Major ба Minor буюу Том ба Бага шинэчлэлтүүд

Энэ процесс нь ихэнх гуравдагч талын прорамуудыг эвдэх хуучин обьект файлууд болон сангуудыг устгах болно. Бүх суулгагдсан портуудыг устгаж дахин суулгах эсвэл ports-mgmt/portupgrade хэрэгсэл ашиглан сүүлд нь шинэчлэхийг зөвлөдөг. Дараах тушаалыг ашиглан ихэнх хэрэглэгчид тест бүтээлтийг ажиллуулахыг хүснэ:

```
# portupgrade -af
```

Энэ нь бүгдийг зөв дахин суулгах баталгаа болох юм. BATCH орчны хувьсагчийг yes гэж тохируулснаар энэ процессийн явцад гарч ирэх хүлээх мөрөнд yes гэж хариулан бүтээх процессийн үед оролцох шаардлагыг үгүй болгоно.

Хэрэв өөрчлөн тохируулсан цөм ашиглагдаж байгаа бол шинэчлэх процесс арай илүү ажиллагаатай. GENERIC цөмийн хуулбар шаардлагатай бөгөөд /boot/GENERIC-д байрлуулах шаардлагатай. Хэрэв GENERIC цөм системд байхгүй бол үүнийг доор дурдсан аргуудын аль нэгийг ашиглан олж авч болно:

- Хэрэв өөрчлөн тохируулсан цөм зөвхөн нэг удаа бүтээгдсэн бол /boot/kernel.old дахь цөм нь GENERIC цөм юм. Энэ санг /boot/GENERIC гэж өөрчлөхөд л болно.
- Машинд физикээр хандах боломжтой гэж тооцвол GENERIC цөмийн хуулбарыг CD-ROM зөөвөрлөгчөөс суулгаж болно. Өөрийн суулгац дискийг хийж дараах тушаалуудыг ашиглана:

  ```
  # mount /cdrom
  # cd /cdrom/X.Y-RELEASE/kernels
  # ./install.sh GENERIC
  ```

  X.Y-RELEASE-г өөрийн ашиглаж байгаа хувилбараар солих хэрэгтэй. GENERIC цөм анхдагчаар /boot/GENERIC-д суулгагдах болно.

- Дээр дурдсан бүгдийг хийх боломжгүй бол GENERIC цөмийг эхээс нь дахин бүтээж суулгаж болох юм:

  ```
  # cd /usr/src
  # env DESTDIR=/boot/GENERIC make kernel
  # mv /boot/GENERIC/boot/kernel/* /boot/GENERIC
  # rm -rf /boot/GENERIC/boot
  ```

  Энэ цөмийг freebsd-update хэрэгсэлд GENERIC гэж харуулахын тулд GENERIC тохиргооны файлыг ямар ч тохиолдолд өөрчилсөн байх ёсгүй. Бас ямар нэг тусгай сонголтуудгүйгээр (аль болох хоосон /etc/make.conf файлтайгаар) бүтээгдсэн байх ёстойг зөвлөдөг.

Энэ үед GENERIC цөм рүү дахин ачаалах шаардлагагүй юм.

Том ба бага хувилбарын шинэчлэлтүүдийг `freebsd-update` тушаалд хувилбарын дугаарыг өгч гүйцэтгэж болно, жишээ нь дараах тушаал FreeBSD 8.1 руу шинэчилнэ:

```
# freebsd-update -r 8.1-RELEASE upgrade
```

Тушаал хүлээн авсны дараа `freebsd-update` системийг шинэчлэхэд шаардлагатай мэдээллийг цуглуулахын тулд тохиргооны файл болон одоогийн системийг шалгана. Ямар бүрэлдэхүүн хэсгүүд илрүүлэгдсэн болон ямар бүрэлдэхүүн хэсгүүд илрүүлэгдээгүй гэдгийг дэлгэц дээр үзүүлнэ. Жишээ нь:

```
Looking up update.FreeBSD.org mirrors... 1 mirrors found.
Fetching metadata signature for 8.1-RELEASE from update1.FreeBSD.org... done.
Fetching metadata index... done.
Inspecting system... done.

The following components of FreeBSD seem to be installed:
kernel/smp src/base src/bin src/contrib src/crypto src/etc src/games
src/gnu src/include src/krb5 src/lib src/libexec src/release src/rescue
src/sbin src/secure src/share src/sys src/tools src/ubin src/usbin
world/base world/info world/lib32 world/manpages

The following components of FreeBSD do not seem to be installed:
kernel/generic world/catpages world/dict world/doc world/games
world/proflibs

Does this look reasonable (y/n)? y
```

Энд хүрэхэд `freebsd-update` шинэчлэлтэд шаардлагатай бүх файлуудыг татан авахаар оролдох болно. Зарим тохиолдолд хэрэглэгчээс юу суулгах эсвэл хэрхэн цааш үргэлжлүүлэх талаар асуултууд асууж болох юм.

Өөрчлөн тохируулсан цөмийг ашиглаж байх үед дээрх алхам дараахтай төстэй анхааруулгыг харуулах болно:

```
WARNING: This system is running a "MYKERNEL" kernel, which is not a
kernel configuration distributed as part of FreeBSD 8.0-RELEASE.
This kernel will not be updated: you MUST update the kernel manually
before running "/usr/sbin/freebsd-update install"
```

Энэ анхааруулгыг энэ үед орхигдуулахад аюулгүй байдаг. Шинэчлэгдсэн GENERIC цөм шинэчлэлтийн явцад завсрын алхам болон ашиглагдах болно.

Бүх засварууд локал систем рүү татагдсаны дараа тэдгээрийг хийж өгөх болно. Машины хурд болон ачааллаас хамаарч энэ процесс нь хугацаа шаардаж болох юм. Тохиргооны файлуудыг нийлүүлэх болно. Файл нийлүүлэгдэх юм уу эсвэл гараар нийлүүлэхэд зориулж засварлагч дэлгэц дээр гарч ирч болох учир процессийн энэ хэсэг хэрэглэгчийн оролцоо шаардана. Амжилттай нийлүүлэлт болгоны үр дүн хэрэглэгчид харуулагдаж процесс

үргэлжлэх болно. Амжилттүй болсон эсвэл орхигдсон нийлүүлэлт нь процессийг зогсоох болно. Хэрэглэгчид нь /etc сангийн нөөцийг хийж master.passwd эсвэл group зэрэг чухал файлуудыг гараар сүүлд нь нийлүүлэхийг хүсэж болох юм.

Систем нь өөрчлөгдөөгүй байгаа бөгөөд бүх засвар оруулалт болон нийлүүлэлт өөр сан дотор болж байгаа болно. Бүх засварууд амжилттай хийгдэж бүх тохиргооны файлууд нийлүүлэгдэж процесс тэгш явагдаж байгаа мэт санагдвал хэрэглэгч өөрчлөлтүүдийг хийх хэрэгтэй.

Энэ процесс дууссаны дараа шинэчлэлтийг дараах тушаалыг ашиглан диск рүү хийж болно.

```
# freebsd-update install
```

Цөм болон цөмийн модулиудад эхлээд засвар хийнэ. Энэ үед машиныг дахих ачаалах ёстой. Хэрэв систем өөрчлөн тохируулсан цөмөөр ажиллаж байгаа бол цөмийг /boot/GENERIC (шинэчлэгдсэн) цөмөөр дараа нь ачаалахаар болгохын тулд nextboot(8)-ийг ашиглана:

```
# nextboot -k GENERIC
```

GENERIC цөмөөр ачаалахаас өмнө (хэрэв шинэчлэгдэж байгаа машинд алсаас хандаж байгаа бол сүлжээнд холбогдон) таны систем зөв ачаалахын тулд шаардлагатай бүх драйверуудыг агуулсан эсэхийг шалгах хэрэгтэй. Ялангуяа хэрэв өмнө нь ажиллаж байсан өөрчлөн тохируулсан цөм ерөнхийдөө цөмийн модулиудаар хангагдаж байдаг ажиллагааг өөртөө агуулсан бол /boot/loader.conf боломжийг ашиглан эдгээр модулиудыг GENERIC цөмд түр зуур ачаалахаа мартуузай. Шинэчлэх процесс бүрэн дуустал шаардлагагүй үйлчилгээнүүд, диск болон сүлжээний холболтууд гэх мэтийг та бас хааж өгч болох юм.

Одоо машин шинэчлэгдсэн цөмөөр ачаалагдах ёстой:

```
# shutdown -r now
```

Систем буцаж ассаны дараа `freebsd-update`-г дахин эхлүүлэх хэрэгтэй. Процессийн төлөв хадгалагдсан болохоор `freebsd-update` эхнээсээ эхлэхгүй бөгөөд бүх хуучин хуваалцсан сангууд болон обьект файлуудыг устгах болно. Энэ шатыг үргэлжлүүлэхийн тулд дараах тушаалыг ажиллуулна:

```
# freebsd-update install
```

Сангуудын хувилбарын тоо дээшилсэн эсэхээс хамаарч суулгах гурван шатны оронд хоёр шат байж болох юм.

Бүх гуравдагч талын програм хангамжийг дахин бүтээж дахин суулгах хэрэгтэй.

Суулгагдсан програм хангамж нь шинэчлэлтийн процессийн явцад устгагдсан сангуудаас хамаарч болох учраас энэ нь шаардлагатай юм. ports-mgmt/portupgrade тушаалыг үүнийг автоматжуулахад ашиглаж болох юм. Энэ процессийг эхлүүлэхийн тулд дараах тушаалыг хэрэглэж болно:

```
# portupgrade -f ruby
# rm /var/db/pkg/pkgdb.db
# portupgrade -f ruby18-bdb
# rm /var/db/pkg/pkgdb.db /usr/ports/INDEX-*.db
# portupgrade -af
```

Үүнийг дууссаны дараа шинэчлэлтийн процессийг `freebsd-update`-ийг сүүлийн удаа дуудаж төгсгөнө. Шинэчлэлтийн процессийн сул байгаа бүх зүйлсийг гүйцээхийн тулд дараах тушаалыг ажиллуулна:

```
# freebsd-update install
```

Хэрэв GENERIC цөм түр зуур ашиглагдаж байсан бол өөрчлөн тохируулсан шинэ цөмийг ердийн хэвшсэн аргаар одоо бүтээж суулгах цаг болжээ.

Машинаа FreeBSD-ийн шинэ хувилбар руу дахин ачаалах хэрэгтэй. Процесс дууслаа.

### 25.2.4. Системийн төлвийн харьцуулалт

`freebsd-update` хэрэгслийг FreeBSD-ийн суулгагдсан хувилбарын төлвийг байгаа зөв хуулбарын эсрэг тест хийхэд хэрэглэж болно. Энэ сонголт нь системийн хэрэгслүүд сангууд болон тохиргооны файлуудын одоогийн хувилбаруудыг шалгадаг. Харьцуулалтыг эхлүүлэхийн тулд дараах тушаалыг ажиллуулна:

```
# freebsd-update IDS >> outfile.ids
```

Тушаалын нэр IDS боловч энэ нь ямар ч тохиолдолд security/snort зэрэг халдлага илрүүлэгч системийг солих зориулалттай биш юм. `freebsd-update` нь өгөгдлийг диск дээр хадгалдаг бөгөөд түүнийг өөрчлөх боломж тодорхой юм. `kern.securelevel` тохиргоог ашиглах болон `freebsd-update`-ийн өгөгдлийг зөвхөн уншигдах файлын систем дээр ашиглагдаагүй тохиолдолд хадгалах нь энэ боломжийг багасгах боловч илүү сайн шийдэл нь системийг DVD эсвэл нууцлаг хадгалсан гадаад USB дискийн төхөөрөмж зэрэг нууцлаг дисктэй харьцуулах явдал юм.

Систем нь одоо шалгагдах бөгөөд файлууд нь өөрсдийн sha256(1) утгуудын хамт, хувилбар дахь мэдэгдэж байгаа сайн утгууд болон одоо суугдсан байгаа утгуудын хамт хэвлэгдэн харуулагдана. Энэ нь яагаад гаралт outfile.ids файл руу илгээгдсэн шалтгаан юм. Үүнийг нүдээр шалгахад хэтэрхий хурдан дээш гүйж удалгүй консолын буфферийг дүүргэх болно.

Эдгээр мөрүүд нь бас хэтэрхий урт боловч гаралтын хэлбэрийг хялбараар задлан ялгаж

болно. Жишээ нь хувилбарт байгаагаас ондоо бүх файлуудын жагсаалтыг авахын тулд дараах тушаалыг ажиллуулна:

```
# cat outfile.ids | awk '{ print $1 }' | more
/etc/master.passwd
/etc/motd
/etc/passwd
/etc/pf.conf
```

Энэ гаралт нь тайрагдсан бөгөөд олон файл байгаа болно. Эдгээр файлуудын зарим нь төрөлхийн өөрчлөлтүүдтэй байна, жишээ нь /etc/passwd нь хэрэглэгч системд нэмэгдсэн болохоор өөрчлөгдсөн байна. Зарим тохиолдолд `freebsd-update` нь шинэчилсэн байж болзошгүй учир цөмийн модулиуд зэрэг бусад файлууд өөр байж болох юм. Тусгай файлууд болон сангуудыг хасахын тулд тэдгээрийг /etc/freebsd-update.conf файлын `IDSIgnorePaths` тохиргоонд нэмж өгнө.

Өмнө хэлэлцсэн хувилбараас гадна нарийн нягт шинэчлэлтийн аргын хэсэг болгон энэ системийг ашиглаж болно.

## 25.3. Portsnap: Портын цуглуулгыг шинэчлэх хэрэгсэл

FreeBSD-ийн үндсэн систем портын цуглуулгыг бас шинэчилдэг portsnap(8) хэрэгслийг агуулдаг. Ажиллуулсны дараа энэ нь алсын сайт руу холбогдож нууц түлхүүрийг шалгаж портын цуглуулгын шинэ хуулбарыг татан авдаг. Түлхүүр нь бүх татаж авсан файлууд татагдаж байхдаа өөрчлөгдөөгүй эсэхийг хянан бүрэн бүтэн байдлыг шалгахад ашиглагддаг. Хамгийн сүүлийн үеийн портын цуглуулгыг татаж авахын тулд дараах тушаалыг ажиллуулна:

```
# portsnap fetch
Looking up portsnap.FreeBSD.org mirrors... 9 mirrors found.
Fetching snapshot tag from geodns-1.portsnap.FreeBSD.org... done.
Fetching snapshot metadata... done.
Updating from Tue May 22 02:12:15 CEST 2012 to Wed May 23 16:28:31 CEST 2012.
Fetching 3 metadata patches.. done.
Applying metadata patches... done.
Fetching 3 metadata files... done.
Fetching 90 patches.....10....20....30....40....50....60....70....80....90. done.
Applying patches... done.
Fetching 133 new ports or files... done.
```

Энэ жишээ нь юу үзүүлж байна вэ гэхээр portsnap(8) одоо байгаа портын өгөгдөлд хэд хэдэн засварууд байгааг олж шалгаж байна. Энэ нь бас уг хэрэгсэл өмнө нь ажилласныг харуулж байгаа бөгөөд хэрэв эхний удаа ажилласан бол цуглуулга татагдан авагдах байсан юм.

portsnap(8) нь `fetch` үйлдлийг хийж дууссаны дараа локал систем дээр байгаа портын

цуглуулга болон дараа дараагийн засваруудыг шалгалтад дамжуулна. `portsnap`-ийг эхний удаа ажиллуулахдаа `extract`-г ашиглан татан авсан файлуудыг суулгаж болно:

```
# portsnap extract
/usr/ports/.cvsignore
/usr/ports/CHANGES
/usr/ports/COPYRIGHT
/usr/ports/GIDs
/usr/ports/KNOBS
/usr/ports/LEGAL
/usr/ports/MOVED
/usr/ports/Makefile
/usr/ports/Mk/bsd.apache.mk
/usr/ports/Mk/bsd.autotools.mk
/usr/ports/Mk/bsd.cmake.mk
...
```

Аль хэдийн суулгасан портын цуглуулгыг шинэчлэхдээ `portsnap update` тушаалыг ашиглах ёстой:

```
# portsnap update
```

Процесс одоо дууссан бөгөөд портын цуглуулыг ашиглан програмуудыг суулгаж эсвэл шинэчилж болно.

`fetch` болон `extract` эсвэл `update` үйлдлүүдийг доор харуулсан шиг дараалуулан ажиллуулж болно:

```
# portsnap fetch update
```

Энэ тушаал нь портын цуглуулгын сүүлийн хувилбарыг татан авч таны машин дээр байгаа локал хувилбарыг /usr/ports санд шинэчилдэг.

# 25.4. Баримтын цуглуулгыг шинэчлэх нь

Үндсэн систем болон портын цуглуулгаас гадна баримтууд нь FreeBSD үйлдлийн системийн салшгүй хэсэг юм. Хэдийгээр FreeBSD-ийн хамгийн сүүлийн үеийн баримтын цуглуулга FreeBSD вэб сайт дээр үргэлж байдаг боловч зарим хэрэглэгчид удаан сүлжээний холболттой эсвэл бүр тогтмол сүлжээний холболтгүй байж болох юм. Азаар FreeBSD-ийн хамгийн сүүлийн үеийн баримтын цуглуулгын локал хуулбарыг арчлан хувилбар бүртэй цуг ирдэг баримтыг шинэчлэх хэд хэдэн арга байдаг.

## 25.4.1. Баримтыг шинэчлэхийн тулд Subversion-г ашиглах нь

FreeBSD-ийн баримтуудын эхийг Subversion ашиглан авч болно. Энэ хэсэг дараах зүйлсийг тайлбарладаг:

- FreeBSD-ийн баримтуудыг эхээс нь бүтээхэд шаардлагатай хэрэгслүүд, баримтын хэрэгслүүдийг хэрхэн суулгах талаар.
- Subversion ашиглан /usr/doc дахь баримтын эхийн хуулбарыг хэрхэн татаж авах талаар.
- FreeBSD-ийн баримтыг эхээс нь хэрхэн бүтээж /usr/shared/doc дотор суулгах талаар.
- Баримтыг бүтээх системийн дэмждэг бүтээлтийн зарим тохируулгууд, өөрөөр хэлбэл баримтын зарим нэг хэл дээрх орчуулгыг зөвхөн бүтээдэг тохируулгууд эсвэл тусгай гаралтын хэлбэржүүлэлтийг сонгодог тохируулгууд.

### 25.4.2. Subversion болон баримтын хэрэгслүүдийг суулгах нь

FreeBSD-ийн баримтыг эхээс нь бүтээхэд нэлээн олон тооны хэрэгслүүдийг шаарддаг. Эдгээр хэрэгслүүд нь FreeBSD-ийн үндсэн системийн хэсэг биш байдаг. Учир нь эдгээр нь ихээхэн хэмжээний дискийн зай шаарддаг бөгөөд FreeBSD-ийн бүх хэрэглэгчдэд хэрэгтэй байдаггүй. Тэдгээр нь FreeBSD-д зориулж шинэ баримтууд идэвхтэй бичдэг эсвэл өөрсдийн баримтыг эхээс нь байнга шинэчилдэг хэрэглэгчдэд зөвхөн хэрэгтэй байдаг.

Бүх шаардлагатай хэрэгслүүд портын цуглуулгад байдаг. Эдгээр хэрэгслүүдийн эхний суулгалт болон хожмын шинэчлэлтүүдийг хялбаршуулах textproc/docproj порт нь FreeBSD-ийг баримтжуулах төслөөс хөгжүүлсэн мастер порт юм.

PostScript® эсвэл PDF баримт шаардлагагүй үед харин textproc/docproj-nojadetex портыг суулгаж болох юм. Баримтын хэрэгслийн энэ хувилбар нь teTeX тайпсет хөдөлгүүрээс бусад бүгдийг багтаасан байдаг. teTeX нь маш олон хэрэгслүүдийн цуглуулга учир PDF гаралт үнэхээр шаардлагагүй тохиолдолд суулгахгүй байх нь зохимжтой байдаг.

Subversion нь textproc/docproj порттой цуг суудаг.

### 25.4.3. Баримтын эхийг шинэчлэх нь

Subversion нь баримтын эхийн цэвэр хуулбарыг татан авч чаддаг.

```
# svn checkout svn://svn.FreeBSD.org/doc/head /usr/doc
```

Баримтын эхийн эхний таталт хугацаа шаардаж болох юм. Дуустал нь хүлээх хэрэгтэй.

Баримтын эхийн дараа дараагийн шинэчлэлтүүдийг доорх тушаалыг ашиглан татан авч болно.

```
# svn update /usr/doc
```

Эхийг татан авсныхаа дараа баримтыг шинэчлэх өөр нэг арга нь /usr/doc сангийн Makefile -аар дэмжигдсэн байдаг бөгөөд дараахийг ажиллуулна:

```
# cd /usr/doc
```

```
# make update
```

### 25.4.4. Баримтын эхийн тааруулж болох тохируулгууд

FreeBSD-ийн баримтжуулалтыг бүтээж шинэчлэх систем нь баримтын зөвхөн тодорхой хэсгийг шинэчлэх эсвэл тусгай орчуулгыг бүтээх процессийг амарчлах хэдэн тохируулгыг дэмждэг. Эдгээр тохируулгуудыг /etc/make.conf файлд бүхэл системийн хувьд зааж өгөх юм уу эсвэл make(1) хэрэгсэлд тушаалын мөрийн тохиргоо маягаар зааж өгч болно.

Дараах тохируулгууд нь эдгээрийн зарим нь юм:

`DOC_LANG`

Бүтээж суулгах хэл ба кодчилолын жагсаалт, жишээ нь Англи баримтад зөвхөн зориулсан `en_US.ISO8859-1` байна.

`FORMATS`

Бүтээх ганц хэлбэржүүлэлт эсвэл гаралтын хэлбэржүүлэлтийн жагсаалт. Одоогоор `html`, `html-split`, `txt`, `ps`, `pdf`, болон `rtf` дэмжигдсэн байгаа.

`DOCDIR`

Баримтыг суулгах газар. Анхдагчаар /usr/shared/doc байдаг.

FreeBSD дээрх системийн тохируулга болон дэмжигдсэн бүтээлтийн хувьсагчуудын талаар дэлгэрэнгүйг make.conf(5)-с үзнэ үү.

FreeBSD-ийн баримтжуулалт бүтээх системийн дэмждэг бүтээлтийн хувьсагчуудын талаар дэлгэрэнгүйг Шинэ хувь нэмэр оруулагчдад зориулсан FreeBSD баримтжуулах төслийн гарын авлагаас үзнэ үү.

### 25.4.5. FreeBSD-ийн баримтуудыг эхээс суулгах нь

Баримтын эхийн хамгийн сүүлийн хормын хувилбарыг /usr/doc санд татаж авснаар суулгагдсан баримтын шинэчлэлтийг хийхэд бүх юм бэлэн болно.

`DOC_LANG` makefile-ийн тохиргоонд заагдсан бүх хэлний бүрэн шинэчлэлтийг дараахийг бичин хийж болно:

```
# cd /usr/doc
# make install clean
```

Хэрэв зөвхөн тусгай хэлний шинэчлэлт хэрэгтэй бол /usr/doc-ийн тухайн хэлний тусгай дэд санд make(1)-ийг ажиллуулж болно, жишээ нь:

```
# cd /usr/doc/en_US.ISO8859-1
# make update install clean
```

Суулгах гаралтын хэлбэржүүлэлтийг `FORMATS` бүтээлтийн хувьсагчийг зааж өгөн хийж өгч

болно, жишээ нь:

```
# cd /usr/doc
# make FORMATS='html html-split' install clean
```

## 25.4.6. Баримтжуулах портуудыг ашиглах нь

Өмнөх хэсэгт FreeBSD-ийн баримтжуулалтыг эхээс нь шинэчлэх аргыг бид танилцуулсан. FreeBSD-ийн бүх системүүдийн хувьд эх дээр тулгуурласан шинэчлэлтүүд нь боломжтой эсвэл практикийн биш байж болох юм. Баримтжуулалтын эхүүдийг бүтээх нь нэлээн их хэмжээний хэрэгслийн цуглуулга буюу *баримтжуулалт бүтээх хэрэгслийн олонлог*, Subversion-ийг тодорхой хэмжээгээр мэдэх, репозиториос эхийг татаж авах болон татаж авсан эхээ бүтээх хэд хэдэн шат дарааллуудыг шаарддаг. Энэ хэсэгт бид FreeBSD-ийн баримтжуулалтын суулгагдсан хуулбаруудыг шинэчлэх өөр аргыг тайлбарлах болно. Энэ нь портын цуглуулгыг ашиглах бөгөөд дараах боломжийг бүрдүүлнэ:

- Бүгдийг бүтээлгүйгээр баримтжуулалтын урьдчилан бүтээсэн хормын хувилбарыг татан авч суулгах (ингэснээр баримтжуулалт бүтээх хэрэгслийн олонлогийг бүхэлд нь суулгах шаардлагагүй болно).
- Баримтжуулалтын эхийг татаж аван портын тогтолцоог ашиглан бүтээх (татаж аван бүтээх алхмуудыг арай хялбар болгодог).

FreeBSD-ийн баримтжуулалтыг шинэчлэх эдгээр хоёр арга нь Баримтжуулалт Инженерчлэлийн Баг <doceng@FreeBSD.org>-ийн сар бүр шинэчилдэг *баримтжуулалтын портуудын* цуглуулгаар дэмжигддэг. Эдгээр нь FreeBSD-ийн портын цуглуулгад docs виртуал төрөлд байдаг.

### 25.4.6.1. Баримтжуулалтын портуудыг бүтээж суулгах нь

Баримтжуулалтын портууд нь баримтжуулалтын бүтээлтийг хялбар болгохын тулд порт бүтээх тогтолцоог хэрэглэдэг. make(1)-ийг тохирох орчны тохиргоонууд болон тушаалын мөрийн тохиргоонуудын хамтаар ажиллуулж баримтжуулалтын эхийг татаж авах процессыг тэд автоматжуулдаг бөгөөд баримтжуулалтын суулгалт болон устгалтыг FreeBSD-ийн бусад порт эсвэл багцын суулгалтын нэгэн адил хялбар болгодог.

Мөн баримтжуулалтын портуудыг бүтээсний дараа тэд хамааралтай *баримтжуулалтыг бүтээх хэрэгслийн олонлогийн* портуудыг бүртгэдэг бөгөөд тэдгээрийг автоматаар бас суулгадаг.

Баримтжуулалтын портуудын зохион байгуулалт нь дараах хэлбэрийн байна:

- Баримтжуулалтын портын файлууд байдаг misc/freebsd-doc-en"мастер порт" байдаг. Энэ нь бүх баримтжуулалтын портуудын үндэс болдог. Анхдагчаар энэ нь Англи баримтжуулалтыг зөвхөн бүтээдэг.
- "Нэг портод бүгд багтсан"misc/freebsd-doc-all байдаг бөгөөд энэ нь байгаа бүх хэл дээр бүх баримтжуулалтыг бүтээж суулгадаг.
- Эцэст нь орчуулга бүрийн хувьд "зарц порт" байдаг, жишээ нь Унгар хэл дээрх

баримтуудад зориулсан misc/freebsd-doc-hu-г дурдаж болно. Эдгээр нь бүгд мастер портоос хамаарах бөгөөд тухайн хэлний орчуулсан баримтжуулалтыг суулгадаг.

Баримтжуулалтын портыг эхээс суулгахын тулд дараах тушаалуудыг ажиллуулна (root эрхээр):

```
# cd /usr/ports/misc/freebsd-doc-en
# make install clean
```

Энэ нь Англи баримтжуулалтыг хуваагдсан HTML хэлбэрээр (http://www.FreeBSD.org дээр ашигладагийн адилаар) бүтээж /usr/local/shared/doc/freebsd санд суулгадаг.

##### 25.4.6.1.1. Нийтлэг Knob болон тохируулгууд

Баримтжуулалтын портуудын анхдагч байдлыг өөрчлөх олон тохиргоо байдаг. Доор цөөхөн хэдэн жагсаалтыг дурдав:

WITH_HTML

HTML хэлбэрээр бүтээхийг зөвшөөрдөг: баримт бүрийн хувьд нэг HTML файл. Хэлбэршүүлсэн баримтжуулалт нь тохирох article.html юм уу эсвэл book.html гэсэн файлуудад зургийн хамтаар хадгалагддаг.

WITH_PDF

Adobe® Acrobat Reader®, Ghostscript эсвэл бусад PDF уншигчдыг ашиглах Adobe®-ийн хөрвөх баримтын хэлбэрээр бүтээхийг зөвшөөрдөг. Хэлбэршүүлсэн баримтжуулалт нь тохирох article.pdf юм уу эсвэл book.pdf гэсэн файлуудад хадгалагддаг.

DOCBASE

Баримтжуулалтын суулгах байрлал. Энэ нь анхдагчаар /usr/local/shared/doc/freebsd байдаг.

Анхдагч суулгах сан нь Subversion аргын ашигладаг сангаас ялгаатайг санаарай. Энэ нь яагаад гэвэл бид порт суулгаж байгаа бөгөөд портууд нь ихэвчлэн /usr/local санд суудаг. Үүнийг PREFIX хувьсагчийг нэмэн өөрчилж болдог.

Энд Унгар баримтжуулалтыг Хөрвөх Баримтын Хэлбэрээр (PDF) суулгахын тулд дээр дурдсан хувьсагчуудыг хэрхэн ашиглахыг харуулсан жишээг үзүүлэв:

```
# cd /usr/ports/misc/freebsd-doc-hu
# make -DWITH_PDF DOCBASE=share/doc/freebsd/hu install clean
```

#### 25.4.6.2. Баримтжуулалтын багцуудыг ашиглах нь

Өмнөх хэсэгт тайлбарласнаар баримтжуулалтын портуудыг эхээс бүтээх нь баримтжуулалтыг бүтээх хэрэгслийн олонлогийг суулгах болон портуудыг бүтээхэд тодорхой хэмжээний дискийн зай шаарддаг. Баримтжуулалтын хэрэгслүүдийг суулгахад эх

үүсвэр хүрэлцэхгүй үед эсвэл эхээс бүтээх нь ихээхэн хэмжээний дискийн зай эзлэх бол баримтжуулалтын портуудын урьдчилан бүтээсэн хормын хувилбаруудыг суулгах боломж бас байдаг.

Баримтжуулалт Инженерчлэлийн Баг <doceng@FreeBSD.org> нь FreeBSD-ийн баримтжуулалтын багцуудын сар бүрийн хормын хувилбаруудыг бэлддэг. Эдгээр хоёртын багцуудыг pkg_add(1), pkg_delete(1) гэх зэрэг багцын хэрэгслүүдийн хамтаар ашиглаж болдог.

Хоёртын багцуудыг ашиглаж байгаа үед FreeBSD-ийн баримтжуулалт нь тухайн хэлний хувьд байгаа *бүх* хэлбэрээр суудаг.

Жишээ нь дараах тушаал Унгар баримтжуулалтын хамгийн сүүлийн урьдчилан бүтээсэн багцыг суулгах болно:

```
# pkg_add -r hu-freebsd-doc
```

Багцууд нь харгалзах портын нэрнээсээ ялгаатай дараах нэрийн хэлбэртэй байдаг: `lang-freebsd-doc`. Энд *lang* нь хэлний кодын богино хэлбэр юм, жишээ нь `hu` нь Унгар, эсвэл `zh_cn` нь хялбаршуулсан Хятад хэл юм.

#### 25.4.6.3. Баримтжуулалтын портуудыг шинэчлэх нь

Өмнө нь суулгасан баримтжуулалтын портыг шинэчлэхийн тулд портууд шинэчлэх аль ч хэрэгсэл байхад хангалттай. Жишээ нь дараах тушаал суулгасан Унгар баримтжуулалтыг ports-mgmt/portupgrade хэрэгслийн тусламжтайгаар зөвхөн багцуудыг ашиглан шинэчилнэ:

```
# portupgrade -PP hu-freebsd-doc
```

## 25.5. Хөгжүүлэлтийн салбарыг дагах нь

FreeBSD-ийн хоёр хөгжүүлэлтийн салбар байдаг: FreeBSD-CURRENT болон FreeBSD-STABLE. Энэ хэсэгт эдгээр тус бүрийг тайлбарлаж өөрийн системийг тус тусын модны хувьд хамгийн шинэ хэлбэрт байнга байлгах талаар тайлбарлах болно. FreeBSD-CURRENT эхлээд хэлэлцэгдэх бөгөөд дараа нь FreeBSD-STABLE-ийн тухай яригдах болно.

### 25.5.1. FreeBSD-ийн одоо үеийн хэлбэрт байх нь

Та үүнийг уншихдаа FreeBSD-CURRENT нь FreeBSD-ийн хөгжүүлэлтийн "bleeding edge салбар буюу амжилт ололтын хамгийн тэргүүний салбар" гэдгийг санаарай. FreeBSD-CURRENT хэрэглэгчдийг техникийн өндөр чадавхитай бөгөөд системийн хүнд хэцүү асуудлуудыг өөрсдөө шийдвэрлэх чадвартай байна гэж тооцдог. Хэрэв та FreeBSD-д анхлан суралцагч бол үүнийг суулгахаасаа өмнө дахин сайн бодоорой.

#### 25.5.1.1. FreeBSD-CURRENT гэж юу вэ?

FreeBSD-CURRENT нь FreeBSD-ийн хамгийн сүүлийн үеийн ажлын эх юм. Энэ нь хийгдэж

байгаа ажлууд, туршилтын өөрчлөлтүүд болон програм хангамжийн дараагийн албан ёсны хувилбарт байхгүй ч байж болох эсвэл байж ч болох шилжилтийн аргуудыг багтаадаг. FreeBSD-ийн олон хөгжүүлэгчид FreeBSD-CURRENT-ийн эх кодыг өдөр болгон эмхэтгэн хөрвүүлж байдаг боловч эхийг бүтээх боломжгүй үе бас байдаг. Эдгээр асуудлууд нь боломжийн хэрээр хурдан шийдэгддэг боловч FreeBSD-CURRENT нь сүйрэл авчрах эсвэл тун их хүссэн ажиллагааг авчрах эсэх нь та яг ямар агшинд эх кодыг татаж авснаас хамаарах юм!

#### 25.5.1.2. FreeBSD-CURRENT хэнд хэрэгтэй вэ?

FreeBSD-CURRENT нь үндсэн 3 сонирхлын бүлэгт зориулагдан хийгдсэн:

1. Эх модны зарим хэсэг дээр идэвхтэйгээр ажиллаж байгаа FreeBSD-ийн хүрээний гишүүд болон "current буюу одоо үеийн хэлбэрт" байлгах нь туйлын шаардлага болсон хүмүүст.
2. FreeBSD-CURRENT-г аль болох ухаалаг байлгахыг хичээж асуудлуудыг шийдвэрлэхэд цагаа зарах хүсэлтэй байдаг идэвхтэй тест хийгч FreeBSD-ийн хүрээний гишүүд. Эдгээр хүмүүс нь өөрчлөлтүүд болон FreeBSD-ийн ерөнхий чиглэлд цаг үеийн саналуудыг тусгахыг хүсэж тэдгээрийг шийдэх засваруудыг илгээдэг бас хүмүүс юм..
3. Юу болж байгааг зөвхөн харж мэдэж байхыг хүссэн эсвэл одоо үеийн эхийг лавлагааны зорилгоор ашиглахыг зөвхөн хүссэн хүмүүс (өөрөөр хэлбэл ажиллуулах биш *унших* зорилгоор). Эдгээр хүмүүс нь хааяа бас санал гаргаж кодонд хувь нэмэр оруулдаг.

#### 25.5.1.3. FreeBSD-CURRENT нь юу *Биш* вэ?

1. Та зарим нэг дажгүй шинэ боломж байгааг сонссон учраас бусдаас түрүүлж урьдчилсан хувилбарын тэдгээр битүүдийг авах таны нэн тэргүүний арга зам. Шинэ боломжийг авч эхэнд байна гэдэг нь та шинэ алдаанууд, хорхойнуудыг бас авч эхэнд байна гэсэн үг юм.
2. Алдааны засваруудыг хурдан авах арга зам. FreeBSD-CURRENT-ийн өгөгдсөн дурын хувилбар нь илэрсэн алдаануудыг засахын хажуугаар бас магадгүй шинэ алдаанууд бас гаргаж байдаг.
3. Аль ч үед "албан ёсоор дэмжигдсэн". Бид өөрсдийн чадлын хирээр "хууль ёсны" 3 FreeBSD-CURRENT бүлгийн аль нэгэнд хүмүүст бодитоор туслахыг хичээдэг, гэхдээ бидэнд ердөө л техникийн дэмжлэг үзүүлэх *цаг байдаггүй*. Энэ нь бид хүмүүст туслах дургүй өөдгүй муухай хүмүүс учраас гэсэн үг биш юм (хэрэв бид байгаагүй бол бид FreeBSD-г хийж байхгүй байх байсан). Бид ердөө л өдрийн хэдэн зуун захидлуудад хариулахын *хажуугаар* FreeBSD дээр ажиллаж чаддаггүй! FreeBSD-г сайжруулах болон туршилтын кодон дээр тавигдсан маш олон асуултуудад хариулах хоёр сонголтын эхнийхийг хөгжүүлэгчид сонгосон юм.

#### 25.5.1.4. FreeBSD-CURRENT ашиглах нь

1. FreeBSD-CURRENT захидлын жагсаалт болон head/-current src модны SVN нийлүүлэлтийн зурвасууд жагсаалтуудад элсэн орно уу. Энэ нь зөвхөн сайн санаанаас гадна бас *чухал* юм. Хэрэв та *FreeBSD-CURRENT захидлын жагсаалт* жагсаалтад ороогүй бол системийн одоогийн төлвийн талаар хүмүүсийн өгч байгаа санал хүсэлтүүдийг харахгүй учраас бусдын аль хэдийн олоод шийдсэн маш их асуудлууд дээр магадгүй та бүдрэн төөрөлдөж дуусах биз ээ. Бүр илүү чухал зүйл нь юу вэ гэвэл таны системийн эрүүл мэндэд эгзэгтэй

байж болох чухал мэдээнүүдээс та хоцрох болно.

head/-current src модны SVN нийлүүлэлтийн зурвасууд жагсаалт нь кодонд оруулсан өөрчлөлт бүрийн бүртгэл оруулгыг болзошгүй сөрөг нөлөөнүүдийн талаар тохирсон мэдээллийн хамтаар танд харах боломжийг олгодог.

Эдгээр жагсаалтууд эсвэл байгаа бусдын аль нэгэнд элсэхийн тулд https://lists.freebsd.org хаяг уруу орж элсэхийг хүссэн жагсаалтаа сонгоорой. Дарааллын үлдсэн зааврууд тэнд байгаа болно. Хэрэв та бүх л эх модон дахь өөрчлөлтийг дагах сонирхолтой байгаа бол Бүх src модны SVN нийлүүлэлтийн зурвасууд("user" ба "projects"-с бусад) жагсаалтад бүртгүүлэхийг бид зөвлөж байна.

2. FreeBSD-ийн толин тусгалаас эхийг авна. Та үүнийг гурван аргаар хийж болно:

   a. Хүссэн хөгжүүлэлт эсвэл салбар хувилбарыг татаж авахдаа svn програмыг ашиглах хэрэгтэй. Энэ аргыг FreeBSD-н хөгжүүлэлтэд хандахад зөвлөдөг. -CURRENT суурь системийн Subversion татаж авах үндсэн URL нь `http://svn.freebsd.org/base/head/` бөгөөд репозиторын хэмжээ их тул зөвхөн хүссэн дэд модоо татаж авахыг зөвлөдөг.

   b. /usr/shared/examples/cvsup санд байх standard-supfile гэж нэрлэгдсэн supfile-тай цуг cvsup програм ашигла. Та дээр дурдсан жишээ supfile-г өөрчлөн cvsup-г өөрийн орчны хувьд тохируулах хэрэгтэй.

`cvsup`-г ашиглах нь хуучирсан бөгөөд төсөл ашиглахыг зөвлөдөггүй.

Жишээ standard-supfile нь FreeBSD-CURRENT-ийн биш FreeBSD-ийн аюулгүй байдлын тусгай салбарыг дагахад хэрэглэгдэнэ. Танд энэ файлыг засварлаж дараах мөрийг өөрчлөх хэрэгтэй болно:

```
*default release=cvs tag=RELENG_X_Y
```

Дээрх мөрийг дараах мөрөөр сольно:

```
*default release=cvs tag=.
```

Хэрэгцээтэй хаяг/шошгонуудын дэлгэрэнгүй тайлбарыг гарын авлагын CVS хаяг/шошгонууд хэсгээс үзнэ үү.

   c. CTM хэрэгслийг ашигла. Хэрэв та маш муу холболттой (өндөр үнэтэй холболтууд эсвэл зөвхөн цахим захидлын хандалт) бол CTM нь сонголт болох юм. Гэхдээ энэ нь бөөн зовлон бөгөөд та эвдэрсэн файлуудтай үлдэж болох юм. Энэ нь үүнийг ховор ашиглахад хүргэдэг бөгөөд ингэснээр ажиллахгүй байх боломжийг нэлээн удаан хугацаагаар ихэсгэдэг. Бид Интернэт холболттой хүмүүст Subversion-г ашиглахыг зөвлөдөг.

3. Хэрэв та эхийг зөвхөн харахаар биш ажиллуулахаар татаж авч байгаа бол зөвхөн сонгосон хэсгүүдийг биш FreeBSD-CURRENT-ийн *бүх* эхийг татаж аваарай. Үүний шалтгаан нь эхийн төрөл бүрийн хэсгүүд нь бусад хаа нэгтээ байгаа шинэчлэлтүүдээс

хамаардаг бөгөөд зөвхөн хэсэг бүлэг эхийг хөрвүүлэхийг оролдох нь таныг бараг л баталгаатайгаар асуудалтай учруулах болно.

FreeBSD-CURRENT-ийг хөрвүүлэхээсээ өмнө /usr/src дахь Makefile-г анхааралтай уншина уу. Эхний удаа та хамгийн багаар бодоход шинэчлэлтийн процессийн хэсэг болох шинэ цөмийг суулгаж ертенцийг дахин бүтээх хэсгээр дамжих хэрэгтэй. FreeBSD-CURRENT захидлын жагсаалт болон /usr/src/UPDATING файлыг унших нь биднийг дараагийн хувилбар уруу шилжихэд заримдаа шаардлагатай болдог бусад эхлүүлэх процедуруудын хувьд хамгийн сүүлийн мэдээлэлтэй байлгах боломжийг бидэнд олгодог.

4. Идэвхтэй бай! Хэрэв та FreeBSD-CURRENT ажиллуулж байгаа бол түүний талаар таныг юу хэлэхийг ялангуяа хэрэв танд өргөжүүлэлт эсвэл алдааны засваруудын талаар санал хүсэлт байвал түүнийг бид мэдэхийг хүсдэг юм. Хавсаргасан кодтой санал хүсэлтүүдийг хамгийн их урам зоригтойгоор хүлээн авдаг билээ!

## 25.5.2. FreeBSD-ийн тогтвортой хэлбэрт байх нь

### 25.5.2.1. FreeBSD-STABLE гэж юу вэ?

FreeBSD-STABLE нь үндсэн хувилбарууд гардаг бидний хөгжүүлэлтийн салбар юм. Өөрчлөлтүүд нь эхлээд тест хийгдэх зорилгоор FreeBSD-CURRENT уруу ордог гэсэн ерөнхий төсөөлөл/таамаглалтайгаар янз бүрийн зөвшөөрлөөр энэ салбар уруу ордог. Энэ нь *одоо болтол* хөгжүүлэлтийн салбар бөгөөд гэхдээ энэ нь ямар ч үед FreeBSD-STABLE-д зориулагдсан эх нь ямар ч зорилгод тохирч эсвэл тохирохгүй байж болно гэсэн үг юм. Энэ нь эцсийн хэрэглэгчид зориулагдсан эх үүсвэр бус ердөө л өөр нэг инженерчлэлийн хөгжүүлэлтийн арга зам юм.

### 25.5.2.2. FreeBSD-STABLE хэнд хэрэгтэй вэ?

Хэрэв та FreeBSD-ийн хөгжүүлэлтийн процессод хувь нэмэр оруулах сонирхолтой, энэ нь ялангуяа FreeBSD-ийн дараагийн "гарах" хувилбартай холбоотой байдаг, эсвэл юу болж байгааг мэдэж байх сонирхолтой байгаа бол та дараах FreeBSD-STABLE-г бодолцох хэрэгтэй.

Аюулгүй байдлын засварууд бас FreeBSD-STABLE салбар уруу орж байдаг нь үнэн боловч та үүнийг хийхийн тулд FreeBSD-STABLE-г заавал дагах *хэрэггүй*. FreeBSD-ийн аюулгүй байдлын зөвлөмжүүд нь тухайн хувилбарт хамааралтай асуудлыг хэрхэн засах тухай тайлбарладаг бөгөөд зөвхөн аюулгүй байдлын үүднээс бүхэл бүтэн хөгжүүлэлтийн салбарыг дагаж байна гэдэг бас зөндөө олон хүсээгүй өөрчлөлтүүдийг авчрах магадлалтай юм.

Бид FreeBSD-STABLE салбар үргэлж хөрвүүлэгдэн эмхэтгэгдэж дандаа ажилладаг байлгахаар чармайж байдаг боловч энэ нь баталгаатай биш юм. Нэмж хэлэхэд код нь FreeBSD-STABLE-д орохоосоо өмнө FreeBSD-CURRENT-д хөгжүүлэгдэж байдаг боловч FreeBSD-CURRENT-г ашиглан ажиллуулдгаас илүү FreeBSD-STABLE-г хүмүүс ажиллуулдаг болохоор FreeBSD-CURRENT-ийн хувьд илэрхий биш байсан алдаанууд болон булангийн тохиолдлууд FreeBSD-STABLE-д илрэх нь заримдаа зайлшгүй юм.

Эдгээр шалтгаануудаас болоод бид FreeBSD-STABLE-г сохроор дагахыг танд *зөвлөдөггүй* бөгөөд энэ нь өөрийн хөгжүүлэлтийн орчиндоо кодыг эхлээд сайтар тест хийлгүйгээр үйлдвэрлэлд (production) ашиглаж байгаа серверүүдээ FreeBSD-STABLE уруу шинэчлэхгүй

байхад танд ялангуяа чухал ач холбогдолтой юм.

Хэрэв танд үүнийг хийх эх үүсвэрүүд байхгүй бол бид FreeBSD-ийн хамгийн сүүлийн үеийн хувилбарыг ажиллуулж хоёртын шинэчлэлт хийх аргыг хувилбараас хувилбар уруу шилжихдээ ашиглахыг танд зөвлөж байна.

#### 25.5.2.3. FreeBSD-STABLE ашиглах нь

1. FreeBSD-STABLE; захидлын жагсаалт жагсаалтад элсэн орно уу. Энэ нь FreeBSD-STABLE-д илэрч болох бүтээлтийн хамаарлууд эсвэл тусгайлсан анхаарал шаардлагатай өөр бусад асуудлуудын талаар танд мэдээлж байх болно. Хөгжүүлэгчид нь зарим нэг маргаантай засвар эсвэл шинэчлэлийн талаар бодож байгаа талаараа бас энэ захидлын жагсаалтад мэдээлдэг бөгөөд ийнхүү санал болгож байгаа өөрчлөлтийн талаар хэрэглэгчдэд ямар нэг асуудал байвал тэдэнд эргээд хариу өгөх боломж олгодог юм.

   Өөрийн дагаж байгаа салбарын тохирох SVN жагсаалтад элсэн орох хэрэгтэй. Жишээ нь хэрэв та 9-STABLE салбарыг дагаж байгаа бол Зөвхөн 9-stable src модны SVN нийлүүлэлтийн зурвасууд жагсаалтад элсэн ороорой. Энэ нь кодонд оруулсан өөрчлөлт бүрийн бүртгэл оруулгыг болзошгүй сөрөг нөлөөнүүдийн талаар тохирсон мэдээллийн хамтаар танд харах боломжийг олгодог.

   Эдгээр жагсаалтууд эсвэл байгаа бусдын аль нэгэнд элсэхийн тулд https://lists.freebsd.org хаяг уруу орж элсэхийг хүссэн жагсаалтаа сонгоорой. Дарааллын үлдсэн зааврууд тэнд байгаа болно. Хэрэв та бүх л эх модон дахь өөрчлөлтийг дагах сонирхолтой байгаа бол Бүх src модны SVN нийлүүлэлтийн зурвасууд("user" ба "projects"-с бусад) жагсаалтад бүртгүүлэхийг бид зөвлөж байна.

2. Хэрэв та шинэ систем суулгаж FreeBSD-STABLE-ээс бүтээсэн сар бүрийн хормын агшны хувилбарыг түүн дээр ажиллуулахыг хүсэж байгаа бол дэлгэрэнгүй мэдээллийн талаар Хормын агшны хувилбарууд вэб хуудаснаас шалгана уу. Үүнээс гадна хамгийн сүүлийн үеийн FreeBSD-STABLE хувилбарыг толин тусгалын хаягуудаас татан авч суулгах боломжтой бөгөөд доор дурдсан зааврыг дагаж өөрийн системийг хамгийн сүүлийн үеийн FreeBSD-STABLE эх код уруу шинэчилж болох юм.

   Хэрэв та FreeBSD-ийн урдны хувилбар аль хэдийн ажиллуулж байгаа бөгөөд эхээс шинэчлэхийг хүсэж байгаа бол FreeBSD-ийн толин тусгал хуудасаас хялбараар хийж болно. Үүнийг гурван аргаар хийж болно:

   a. Хүссэн хөгжүүлэлт эсвэл салбар хувилбарыг татахдаа svn програмыг ашиглах хэрэгтэй. Энэ аргыг FreeBSD-н хөгжүүлэлтэд хандахад зөвлөдөг. Салбарын нэрсэд одоогийн хөгжүүлэлтийн толгой хувилбарын хувьд `head`, болон `stable/9` эсвэл `releng/9.0` гэх зэрэг хувилбар инженерчлэлийн хуудас дахь салбарууд ордог. Суурь системийн Subversion татаж авах үндсэн URL нь `http://svn.freebsd.org/base/` бөгөөд репозиторын хэмжээ их тул зөвхөн хүссэн дэд модоо татаж авахыг зөвлөдөг.

   b. /usr/shared/examples/cvsup санд байх standard-supfile гэж нэрлэгдсэн supfile-тай цуг cvsup програм ашигла. Та дээр дурдсан жишээ supfile-г өөрчлөн cvsup-г өөрийн орчны хувьд тохируулах хэрэгтэй.

      `cvsup` нь хуучирсан бөгөөд төсөл ашиглахыг зөвлөдөггүй.

c. CTM хэрэгслийг ашигла. Хэрэв танд Интернэт уруу холбогдсон хурдан хямд холболт байхгүй бол энэ аргыг та ашиглах хэрэгтэй.

3. Гол нь хэрэв та эхэд хурдан, шаардлагын улмаас хандах хэрэгтэй болоод холболтуудын зурвасын өргөн ач холбогдолгүй бол `cvsup` эсвэл `ftp` ашиглаарай. Бусад тохиолдолд CTM-г ашигла.

4. FreeBSD-CURRENT-ийг хөрвүүлэхээсээ өмнө /usr/src дахь Makefile-г анхааралтай уншина уу. Эхний удаа та хамгийн багаар бодоход шинэчлэлтийн процессийн хэсэг болох шинэ цөмийг суулгаж ертенцийг дахин бүтээх хэсгээр дамжих хэрэгтэй. FreeBSD-CURRENT захидлын жагсаалт болон /usr/src/UPDATING файлыг унших нь биднийг дараагийн хувилбар уруу шилжихэд заримдаа шаардлагатай болдог бусад эхлүүлэх процедуруудын хувьд хамгийн сүүлийн мэдээлэлтэй байлгах боломжийг бидэнд олгодог.

## 25.6. Өөрийн эхийг хамгийн сүүлийн хэлбэрт аваачих нь

Интернетийн (эсвэл цахим захидал) холболт ашиглан FreeBSD төслийн эхүүдийн аль ч хэсгийн хувьд эсвэл таны юу сонирхож байгаагаас хамааран бүх хэсгүүдийг хамгийн шинэ байлгаж байх төрөл бүрийн аргууд байдаг. Бидний санал болгодог үндсэн үйлчилгээнүүд бол Subversion, Anonymous буюу нэргүй CVS, CVSup болон CTM юм.

Өөрийн эх модны зөвхөн зарим хэсгийг шинэчлэх боломжтой боловч цорын ганц шинэчлэх арга бол модыг бүтнээр нь шинэчилж хэрэглэгчийн талбар (өөрөөр хэлбэл /bin болон /sbin гэх мэт дэх хэрэглэгчийн талбарт ажилладаг бүх програмууд) болон цөмийн эхүүдийг дахин эмхэтгэх явдал юм. Өөрийн эх модны зөвхөн нэг хэсэг зөвхөн цөм эсвэл зөвхөн хэрэглэгчийн талбарыг шинэчлэх нь асуудлууд гарахад ихэвчлэн хүргэдэг. Эдгээр асуудлууд нь эмхэтгэлтийн үеийн алдаануудаас авахуулаад цөмийн сүйрлүүд эсвэл өгөгдлийн эвдрэлийг хүртэл хамардаг.

Subversion, Нэргүй CVS болон CVSup нь эхийг шинэчлэхдээ *mamax* загварыг хэрэглэдэг. Subversion-ийн хувьд хэрэглэгч (эсвэл `cron` скрипт) `svn` програмыг эхлүүлэн файлуудыг хамгийн шинэ хэлбэрт авчирдаг. Локал эх модыг шинэчлэхэд зөвлөдөг арга бол Subversion юм. `cvsup` ба `cvs` нь ижил зарчмаар ажиллах боловч хуучирсан бөгөөд Subversion-ийг ашиглахыг зөвлөдөг. Таны хүлээн авах шинэчлэлтүүд нь хамгийн сүүлийн минут хүртэлх үеийнх байх бөгөөд та тэдгээрийг зөвхөн өөрийн хүссэн тэр үедээ авдаг. Та өөрийн шинэчлэлтүүдийг таны сонирхож байгаа тусгайлсан файлууд эсвэл сангуудаар хялбараар хязгаарлаж болно. Шинэчлэлтүүд нь таны юуг авахыг хүссэн болон танд юу байгаагаас хамааран серверээр тухайн үед үүсгэгддэг. Үнэхээр шаардлагагүй л бол хуучирч ирээдүйд үргэлжүүлэн ашиглахаа болих бусад синхрон хийх аргуудаас илүүтэй Subversion-г ашиглах ёстой юм.

Нөгөө талаас CTM нь танд байгаа эхийг мастер архив дахь эхтэй лавлаж асуух зарчмаар харьцуулдаггүй бөгөөд өөрөөр хэлбэл тэдгээрийг татаж авдаггүй. Ингэхийн оронд харин өмнө нь ажиллуулснаас хойшх файл дахь өөрчлөлтүүдийг таньдаг скрипт өдөрт хэд хэдэн удаа мастер CTM машин дээр ажиллаж илэрсэн өөрчлөлтүүдийг шахаж дарааллын-дугаар тавин цахим захидлаар дамжуулахад зориулан кодчилдог (зөвхөн хэвлэгдэх боломжтой

ASCII хэлбэрээр). Эдгээр "CTM дельтануудыг" авсаны дараа тэдгээрийг автоматаар декод хийж шалган хэрэгчид байгаа эхийн хуулбарт өөрчлөлтүүдийг хийх ctm_rmail(1) хэрэгсэл уруу өгдөг. Энэ процесс нь CVSup-с хамаагүй илүү үр дүнтэй бөгөөд энэ нь *маш* биш харин *түлхэх* загвар учраас бидний серверийн эх үүсвэрт бага ачаалал учруулдаг юм.

Мэдээж үүнээс гадна харилцан сул болон давуу талуудтай асуудлууд байдаг. Хэрэв та санамсаргүйгээр өөрийн архивын хэсгийг устгачих юм бол CVSup үүнийг илрүүлж эвдэрсэн хэсгүүдийг дахин бүтээж өгдөг. CTM ингэж хийдэггүй бөгөөд хэрэв та өөрийн эх модны зарим хэсгийг устгасан (бас нөөцлөн аваагүй) бол та дахин шинээр эхнээс нь (хамгийн сүүлийн үеийн CVS "суурь дельтагаас") эхэлж CTM-ийн тусламжтайгаар бүгдийг дахин бүтээх буюу эсвэл Нэргүй CVS-ийн тусламжтайгаар муу битүүдийг ердөө л устгаж дахин сүүлийн хэлбэрт аваачих хэрэгтэй болно.

## 25.7. "Ертөнц"ийг дахин бүтээх нь

Та өөрийн локал эх модоо FreeBSD-ийн тухайн хувилбарын (FreeBSD-STABLE, FreeBSD-CURRENT, гэх зэрэг) хамгийн сүүлийн үеийн хэлбэрт аваачсаныхаа дараа та эх модоо ашиглан системийг дахин бүтээж болно.

*Нөөц хий*

Та дээрхийг хийхээсээ *өмнө* өөрийн системийг нөөцлөн авах нь ямар чухал болохыг энэ нь хангалттай хэлж өгч чаддаггүй. Ертөнцийг дахин бүтээх нь (хэрэв та эдгээр зааврууды дагасан тохиолдолд) хялбар боловч таныг алдаа гаргахад эсвэл бусдын эх модонд хийсэн алдаанууд нь таны системийг ачаалагдахгүй болгох нөхцөлд зайлшгүй хүргэдэг.

Нөөц хийж авсан эсэхээ шалгаарай. Засварлах уян диск эсвэл ачаалагдах CD-г гарын дор байлгаарай. Магадгүй та үүнийг хэзээ ч хэрэглэхгүй байж болох юм, гэхдээ харамсахаасаа өмнө аюулгүй байж байх нь илүү дээр юм!

*Тохирох захидлын жагсаалтад бүртгүүл*

FreeBSD-STABLE болон FreeBSD-CURRENT салбарууд нь угаасаа *хөгжүүлэлтэд байдаг*. FreeBSD-д хувь нэмэр оруулж байгаа хүмүүс нь хүн л учраас алдаанууд заримдаа гардаг.

Заримдаа эдгээр алдаанууд нь нэг их хор хөнөөлгүй бөгөөд ердөө л таны системийг шинэ оношлогооны анхааруулга хэвлэхэд хүргэдэг. Эсвэл өөрчлөлт нь сүйрлийн байж болзошгүй байдаг бөгөөд таны системийг ачаалагдахгүй болгож эсвэл файлын системүүдийг чинь устгаж (эсвэл бүр муу юм болж) болох юм.

Эдгээртэй адил асуудлууд гарвал асуудлын учир шалтгаан болон аль систем дээр энэ асуудал хамааралтайг тайлбарласан "heads up буюу бүхний сонорт" хандсан зарлал тохирох захидлын жагсаалтад илгээгддэг. Тэгээд "all clear буюу бүгд цэвэр" зарлал асуудал шийдэгдсэний дараа тавигддаг.

Хэрэв та FreeBSD-STABLE эсвэл FreeBSD-CURRENT-ийг дагахыг оролдож FreeBSD-STABLE; захидлын жагсаалт эсвэл FreeBSD-CURRENT захидлын

жагсаалт-г харгалзуулан уншихгүй байгаа бол энэ нь та өөртөө гай төвөг асууж байна л гэсэн үг юм.

`make world` *тушаалыг бүү ашигла*

Ихэнх хуучин баримтууд үүнд зориулан `make world` тушаалыг ашиглахыг зөвлөдөг. Энэ тушаалыг ажиллуулснаар зарим нэг чухал алхмуудыг алгасах бөгөөд та юу хийж байгаагаа мэдэж байгаа тохиолдолд үүнийг зөвхөн ашиглах хэрэгтэй. Бараг ихэнх тохиолдолд `make world` хийх нь буруу зүйл бөгөөд энд тайлбарласан процедурыг түүний оронд ашиглах ёстой юм.

### 25.7.1. Шалгагдсан аргаар өөрийн системийг шинэчлэх нь

Өөрийн системийг шинэчлэхийн тулд өөрт чинь байгаа эхийн хувилбарт шаардлагатай байгаа бүтээхээс урьдах алхмууд та /usr/src/UPDATING файлд байгаа эсэхийг шалгах хэрэгтэй бөгөөд үүний дараа энд дурдсан процедурыг ашиглана.

Эдгээр шинэчлэлтийн алхмууд нь таныг хуучин хөрвүүлэгч, хуучин цөм, хуучин ертөнц болон хуучин тохиргооны файлууд бүхий FreeBSD-ийн хуучин хувилбар ашиглаж байгаа гэж тооцдог. "Ертөнц" гэдгийг бид энд системийн гол хоёртын файлууд, сангууд болон програмын файлууд гэж ойлгоно. Хөрвүүлэгч нь "ертөнц"ийн хэсэг бөгөөд цөөн асуудлуудтай байдаг.

Таныг шинэ системийн эхийг аль хэдийн авсан байгаа гэж бид бас энд тооцдог. Тухайн систем дээр байгаа эхүүд бас хуучин байвал шинэ хувилбар руу шилжүүлэх талаар бичсэн Өөрийн эхийг хамгийн сүүлийн хэлбэрт аваачих нь-с дэлгэрэнгүйг үзнэ үү.

Системийг эхээс шинэчлэх нь эхлээд санагдсанаасаа илүү нарийн байдаг бөгөөд тойрон гарах боломжгүй, хамаарлууд бүхий шинэ асуудлууд гардгаас болоод FreeBSD-ийн хөгжүүлэгчид зөвлөдөг чиг хандлагаа жил ирэх тутам нэлээнээр өөрчлөх шаардлагатай болсон. Энэ хэсгийн үлдсэн хэсэг нь одоогоор зөвлөж байгаа шинэчлэх дарааллын талаар тайлбарлах болно.

Амжилттай болох шинэчлэх дараалал бүр дараах асуудлуудыг шийдэх ёстой:

- Хуучин хөрвүүлэгч шинэ цөмийг бүтээж чадахгүй байж болох юм. (Хуучин хөрвүүлэгчид заримдаа алдаатай байдаг.) Тиймээс шинэ цөмийг шинэ хөрвүүлэгчээр бүтээх ёстой. Ялангуяа шинэ цөм бүтээхээсээ өмнө шинэ хөрвүүлэгчийг бүтээх хэрэгтэй. Энэ нь шинэ хөрвүүлэгчийг заавал шинэ цөмөөс өмнө *суулгасан* байх ёстой гэсэн үг биш юм.
- Шинэ ертөнц шинэ цөмийн боломжууд дээр тулгуурлаж байж болох юм. Тиймээс шинэ цөмийг шинэ ертөнцийг суулгахаасаа өмнө суулгасан байх шаардлагатай.

Эдгээр хоёр асуудал нь бидний дараагийн хэсгүүдэд тайлбарлах гол `buildworld`, `buildkernel`, `installkernel`, `installworld` дарааллын үндэс болдог. Энэ нь одоогоор зөвлөдөг шинэчлэлтийн процессийг та яагаад заавал сонгох ёстойг харуулсан бүх шалтгаануудын бүрэн дүүрэн жагсаалт биш юм. Зарим нэг тийм ч мэдээжийн биш зүйлсийг доор жагсаав:

- Хуучин ертөнц шинэ цөм дээр зөв ажиллахгүй байж болох учир та шинэ цөм суулгасныхаа дараа шинэ ертөнцийг даруйхан суулгах ёстой.

- Шинэ ертөнц суулгахаасаа өмнө зарим нэг тохиргооны өөрчлөлтүүдийг хийх ёстой боловч зарим нь хуучин ертөнцийг эвдэж магадгүй юм. Тийм болохоор хоёр өөр тохиргооны шинэчлэлтийн алхам ерөнхийдөө шаардлагатай байдаг.
- Ихэнх хэсгийн хувьд шинэчлэх процесс нь зөвхөн файлуудыг солих юм уу эсвэл нэмдэг бөгөөд байгаа хуучин файлуудыг устгадаггүй. Цөөн тохиолдолд энэ нь асуудал үүсгэж болох юм. Үүний дүнд шинэчлэх арга зам нь зарим нэг алхам дээр гараар устгах тодорхой файлуудыг заримдаа зааж өгдөг. Үүнийг ирээдүйд автоматчилах юм уу эсвэл үгүй ч байж болох юм.

Эдгээр зүйлс нь дараах зөвлөсөн дараалалд хүргэдэг. Тухайн шинэчлэлтүүдэд зориулсан дэлгэрэнгүй дараалал нь нэмэлт алхмуудыг шаардаж болохыг санаарай. Гэхдээ эдгээр гол процессууд тодорхой хугацаагаар өөрчлөгдөхгүй байх ёстой юм:

1. `make buildworld`

   Энэ нь эхлээд шинэ хөрвүүлэгч болон хамааралтай цөөн хэрэгслүүдийг бүтээж дараа нь шинэ ертөнцийн бусдыг хөрвүүлэхийн тулд шинэ хөрвүүлэгчийг ашигладаг. Үр дүн нь /usr/obj-д хадгалагддаг.

2. `make buildkernel`

   config(8) болон make(1)-ийг ашигладаг хуучин аргаасаа ялгаатай нь энэ тушаал /usr/obj санд байрлаж байгаа *шинэ* хөрвүүлэгчийг ашигладаг. Энэ нь хөрвүүлэгч болон цөмийн хооронд тохиромжгүй байдал үүсэхээс таныг хамгаалдаг.

3. `make installkernel`

   Шинэ цөм болон цөмийн модулиудыг дискэд байрлуулж шинээр шинэчилсэн цөмөөр ачаалах боломжийг бүрдүүлдэг.

4. Ганц хэрэглэгчийн горим руу ачаалан орно.

   Ганц хэрэглэгчийн горим нь ажиллаж байгаа програм хангамжуудыг шинэчлэхэд гарах асуудлуудыг багасгадаг. Энэ нь бас шинэ цөм дээр хуучин ертөнцийг ажиллуулахад гарах асуудлыг багасгадаг.

5. `mergemaster -p`

   Энэ нь шинэ ертөнцөд зориулж зарим нэг тохиргооны файлуудын эхний шинэчлэлтүүдийг хийдэг. Жишээ нь энэ нь шинэ хэрэглэгчийн бүлгийг системд нэмэх, эсвэл шинэ хэрэглэгчийн нэрсийг нууц үгийн мэдээллийн санд нэмж болох юм. Сүүлийн шинэчлэлтээс хойш шинэ бүлгүүд эсвэл системийн тусгай хэрэглэгчийн бүртгэлүүдийг нэмэх үед энэ нь ихэвчлэн шаардлагатай байдаг. Ингэснээр `installworld` алхам нь шинээр суулгагдсан системийн хэрэглэгч эсвэл системийн бүлгийн нэрсийг ямар ч асуудалгүйгээр ашиглах боломжтой болох юм.

6. `make installworld`

   /usr/obj сангаас ертөнцийг хуулдаг. Та одоо диск дээрээ шинэ цөм болон шинэ ертөнцтэй боллоо.

7. `mergemaster`

   Нэгэнт диск дээрээ шинэ ертөнцтэй болсон болохоор та одоо үлдсэн тохиргооны файлуудаа шинэчилж болно.

8. Дахин ачаална.

   Шинэ цөм болон шинэ ертөнцийг шинэ тохиргооны файлуудтай дуудахын тулд машиныг бүрэн дахин ачаалах хэрэгтэй.

Хэрэв та FreeBSD-ийн нэг салбар дотор нэг хувилбараас илүү сүүлийн хувилбар руу шинэчилж байгаа бол, өөрөөр хэлбэл 7.0-с 7.1 рүү шинэчилж байгаа бол хөрвүүлэгч, цөм, хэрэглэгчийн талбар болон тохиргооны файлуудын хооронд айхтар таарамжгүй байдлууд тантай бараг л тохиолдохгүй учир энэ арга нь заавал шаардлагатай биш байж болох юм. Хуучин арга болох `make world` болон шинэ цөмийг бүтээж суулгах нь жижиг шинэчлэлтийн хувьд хангалттай сайн ажиллаж болох юм.

Гэхдээ гол хувилбаруудын хооронд шинэчлэлт хийж байх үед энэ арга замыг дагахгүй байгаа хүмүүст зарим асуудлууд учирч болох юм.

Олон шинэчлэлтүүд (өөрөөр хэлбэл 4.*X*-с 5.0 руу) тусгайлсан нэмэлт алхмуудыг (жишээ нь installworld хийхээс өмнө тусгай файлуудын нэрийг өөрчлөх эсвэл устгах гэх мэт) шаардаж болохыг энд тэмдэглэх нь зүйтэй юм. /usr/src/UPDATING файлыг анхааралтай уншина уу, ялангуяа одоогоор зөвлөсөн байгаа шинэчлэх дарааллыг тусгайлан тайлбарласан төгсгөл хэсгийг уншаарай.

Зарим нэг тохиромжгүй байдалтай холбоотой асуудлуудаас бүрэн гүйцэд хамгаалах боломжгүйг хөгжүүлэгчид мэдсэнээр энэ арга нь цаг хугацааны туршид сайжруулагдсаар ирсэн юм. Одоогийн арга замууд нь удаан хугацааны туршид тогвортой байна гэдэгт найдаж байна.

Дүгнэхэд FreeBSD-г эхээс шинэчлэхэд одоогоор зөвлөдөг арга бол:

```
# cd /usr/src
# make buildworld
# make buildkernel
# make installkernel
# shutdown -r now
```

`buildworld` алхмаас өмнө `mergemaster -p` тушаалыг нэмж ажиллуулах цөөн ховор тохиолдлууд байдаг. Эдгээрийн талаар UPDATING файлд тайлбарласан байдаг. Хэрэв та FreeBSD-ийн нэг буюу олон голлох хувилбаруудын дагуу шинэчлэл хийхгүй байгаа бол ерөнхийдөө энэ алхмыг эмээлгүйгээр орхиж болох юм.

`installkernel` амжилттай дууссаны дараа та ганц хэрэглэгчийн горим уруу ачаалах хэрэгтэй (өөрөөр хэлбэл `boot -s` тушаалыг дуудагч мөрөөс ашиглана). Дараа нь доор дурдсан тушаалуудыг ажиллуулна:

```
# mount -u /
# mount -a -t ufs
# adjkerntz -i
# mergemaster -p
# cd /usr/src
# make installworld
# mergemaster
# reboot
```

*Тайлбаруудыг цааш уншина уу*

Дээр тайлбарласан дараалал нь зөвхөн таныг эхлэхэд туслах богино сэргээлт болох юм. Гэхдээ хэрэв та ялангуяа өөрчлөн тохируулсан цөмийн тохиргоо ашиглахыг хүсэж байгаа бол дараах хэсгүүдийг уншиж алхам бүрийг сайтар ойлгох хэрэгтэй.

### 25.7.2. /usr/src/UPDATING файлыг унш

Өөр юм хийж эхлэхээсээ өмнө та /usr/src/UPDATING-г (эсвэл эх кодын хуулбар хаана байгаа тэндээс үүнтэй төстэй файлыг ) уншаарай. Энэ файл нь танд учирч болзошгүй асуудлуудын талаар чухал мэдээлэл агуулдаг бөгөөд эсвэл таны ажиллуулах зарим нэг тушаалуудын дарааллын талаар заасан байдаг. Хэрэв UPDATING файл таны энд уншсантай зөрчилдөж байвал UPDATING файлд заасныг дагах хэрэгтэй.

UPDATING файлыг унших нь өмнө нь тайлбарласнаар зөв захидлын жагсаалтад бүртгүүлэхтэй харьцуулах юм бол хүлээн зөвшөөрч болохуйц орлогч байж чадахгүй юм. Энэ хоёр шаардлага нь нэмэлт бөгөөд заавал шаардлагатай биш юм.

### 25.7.3. /etc/make.conf файлыг шалга

/usr/shared/examples/etc/make.conf болон /etc/make.conf файлыг шалгаарай. Эхнийх нь зарим нэг анхдагч тодорхойлолтуудыг агуулдаг - тэдгээрийн ихэнх нь тайлбар болон хаагдсан байдаг. Та системээ эхээс нь дахин бүтээх үедээ тэдгээрийг ашиглахын тулд /etc/make.conf файлд нэмэх хэрэгтэй. /etc/make.conf файлд нэмсэн болгон `make` тушаалыг ажиллуулах бүрд бас ашиглагддаг учир өөрийн системдээ зориулан тэдгээрийг боломжийн утгаар тохируулж өгөх нь зүйтэй юм.

Ердийн хэрэглэгч /usr/shared/examples/etc/make.conf файлд байдаг `NO_PROFILE` мөрийг /etc/make.conf уруу хуулж тайлбар болгосныг болиулж нээхийг магадгүй хүсэж болох юм.

`NOPORTDOCS` гэх зэрэг бусад тодорхойлолтуудыг шалгаж танд хамаатай эсэхээс хамаарч оруулах эсэхээ шийдээрэй.

### 25.7.4. /etc дэх файлуудыг шинэчил

/etc сан нь таны системийн тохиргооны мэдээллийн ихэнх хэсгийг агуулдгаас гадна системийг эхлүүлэхэд ажилладаг скриптүүд энд байдаг. Эдгээр скриптүүдийн зарим нь

FreeBSD-ийн хувилбараас хувилбарт өөрчлөгддөг.

Тохиргооны файлуудын зарим нь бас системийг ажиллуулахад өдөр тутам хэрэглэгддэг. Ялангуяа /etc/group-г дурдаж болно.

`make installworld` тушаалын суулгалт хийх хэсэг нь зарим нэг хэрэглэгчийн нэр эсвэл бүлгүүд байж байна гэж тооцдог тохиолдлууд байдаг. Шинэчлэл хийж байх үед эдгээр хэрэглэгчид эсвэл бүлгүүд ихэнхдээ байхгүй байдаг. Энэ нь шинэчлэл хийхэд асуудал учруулдаг. Зарим тохиолдолд `make buildworld` нь эдгээр хэрэглэгчид эсвэл бүлгүүд байгаа эсэхийг шалгана.

Үүний нэг жишээ нь `smmsp` хэрэглэгч нэмэгдсэн тохиолдол юм. mtree(8) нь /var/spool/clientmqueue-г үүсгэхийг оролдох үед хэрэглэгчийн суулгалтын процесс энэ асуудлаас болж амжилтгүй болж байсан.

Үүний шийдэл нь mergemaster(8)-г ертөнцийг бүтээхээс урд `-p` тохируулгатай ажиллуулах явдал юм. Энэ нь `buildworld` эсвэл `installworld` тушаалыг амжилттай болгоход зөвхөн шаардлагатай файлуудыг харьцуулдаг.

Хэрэв та ялангуяа хэтэрхий санаа зовж байгаа бол тухайн бүлэгт харьяалагдаж байгаа нэрийг нь өөрчилж байгаа эсвэл устгаж байгаа ямар файлууд байгааг өөрийн системээс шалгаарай:

```
# find / -group GID -print
```

дээрх нь *GID* (энэ бүлгийн нэр байж болно эсвэл бүлгийн тоон ID байж болно) бүлгийн эзэмшдэг файлуудыг харуулна.

### 25.7.5. Ганц хэрэглэгчийн горимд шилж

Та системийг ганц хэрэглэгчийн горимд эмхэтгэхийг хүсэж болох юм. Энэ нь шинэчлэлтийг арай илүү хурдасгах илэрхий ашиг тустайгаас гадна системийг дахин суулгах нь системийн стандарт хоёртын файлууд, libraries буюу туслах сангууд, оруулгын файлууд гэх зэрэг системийн маш олон чухал файлуудыг хөнддөг. Ажиллаж байгаа систем дээр эдгээрийг өөрчлөх нь (ялангуяа хэрэв тухайн үед таны систем дээр идэвхтэй хэрэглэгчид байвал) гай төвгийг өөрөө эрж байна гэсэн үг юм.

Өөр нэг арга бол системийг олон хэрэглэгчийн горимд эмхэтгэж дараа нь суулгахдаа ганц хэрэглэгчийн горимд шилжин хийх явдал юм. Хэрэв та энэ замаар хийхийг хүсэж байвал бүтээлт дуустал дараах алхмууд дээр хүлээж байгаарай. Та `installkernel` эсвэл `installworld` хийх хүртлээ ганц хэрэглэгчийн горимд оролгүйгээр хүлээж байж болно.

Супер хэрэглэгч болоод та доор дурдсаныг:

```
# shutdown now
```

ажиллаж байгаа системээс ганц хэрэглэгчийн горим уруу оруулахдаа ажиллуулж болно.

Өөр нэг арга нь системийг дахин ачаалаад ачаалалтын тушаал хүлээх мөрөн дээр "single user буюу ганц хэрэглэгч" тохируулгыг сонгоорой. Ингэхэд систем ганц хэрэглэгчийг ачаална. Бүрхүүлийн тушаал хүлээх мөрөнд та доор дурдсан тушаалуудыг ажиллуулах шаардлагатай:

```
# fsck -p
# mount -u /
# mount -a -t ufs
# swapon -a
```

Энэ нь файлын системүүдийг шалгаж /-г дахин унших/бичихээр дахин холбож бусад бүх UFS файлын системүүдийг /etc/fstab-д заасны дагуу холбон дараа нь swap-ийг идэмвхжүүлэх болно.

Хэрэв таны CMOS цаг нь GMT биш локал хугацаагаар тохируулагдсан бол (хэрэв date(1) тушаалын гаралт зөв цаг болон бүсийг харуулахгүй бол энэ нь үнэн) та дараах тушаалыг бас ажиллуулах хэрэгтэй болж болох юм:

```
# adjkerntz -i
```

Энэ нь таны локал цагийн бүсийн тохируулгуудыг зөвөөр тохируулж өгдөг - үүнийг хийхгүй бол та дараа нь зарим асуудлуудтай тулгарч магадгүй.

## 25.7.6. /usr/obj-г устга

Системийн хэсгүүд дахин бүтээгдсэнийхээ дараа (анхдагчаар) /usr/obj дахь сангуудад байршдаг. Эдгээр сангууд нь /usr/src дотор байгааг халхалдаг.

Та `make buildworld` процессийг хурдасгаж болох бөгөөд энэ санг бас устгаснаар хамаарлын зовлонгуудаас өөрийгөө магадгүй аврах болно.

/usr/obj доторх зарим файлуудад immutable буюу хувиршгүй туг тавигдсан (дэлгэрэнгүй мэдээллийг chflags(1)-с үзнэ үү ) байж болох бөгөөд түүнийг эхлээд арилгах хэрэгтэй.

```
# cd /usr/obj
# chflags -R noschg *
# rm -rf *
```

## 25.7.7. Үндсэн системийг дахин эмхэтгэ

### 25.7.7.1. Гаралтыг хадгалах нь

make(1)-г ажиллуулахдаа гарах үр дүнг өөр файл уруу хадгалах нь зүйтэй юм. Хэрэв ямар нэг юм болохоо боливол та алдааны мэдэгдлийн хуулбартай байх болно. Энэ нь танд юу буруутсаныг шинжлэхэд чинь тус болохгүй байж болох боловч та өөрийн энэ асуудлаа FreeBSD-ийн аль нэг захидлын жагсаалт уруу илгээсэн тохиолдолд бусдад тус болж болох

юм.

Үүнийг хамгийн амраар хийхийн тулд script(1) тушаалыг бүх гаралтыг хадгалах файлын нэрийг заасан параметрийн хамтаар ашиглана. Та үүнийг ертөнцийг дахин бүтээхээс өмнөхөн нэн даруй хийж дараа нь процесс дууссаны дараа `exit` гэж бичиж гарна.

```
# script /var/tmp/mw.out
Script started, output file is /var/tmp/mw.out
# make TARGET
… compile, compile, compile …
# exit
Script done, …
```

Хэрэв та үүнийг хийх бол гаралтыг /tmp дотор *битгий* хадгалаарай. Энэ сан нь таныг дахин ачаалсны дараа цэвэрлэгдэж болох юм. Энэ файлыг хадгалах арай илүү боломжийн газар нь /var/tmp (өмнөх жишээн дээрх шиг) эсвэл `root` хэрэглэгчийн гэр сан байж болох юм.

#### 25.7.7.2. Үндсэн системийг эмхэтгэ

Та /usr/src сан дотор байх шаардлагатай:

```
# cd /usr/src
```

(гэхдээ мэдээж таны код өөр газар байгаа тохиолдолд тэр сан уруугаа орох хэрэгтэй).

Ертөнцийг дахин бүтээхдээ та make(1) тушаалыг ашиглана. Энэ тушаал нь FreeBSD-ийн агуулсан програмууд ямар дарааллаар дахин хэрхэн бүтээгдэх зэргийг тайлбарласан Makefile файлаас зааврууд уншдаг.

Таны бичих тушаалын мөрийн ерөнхий хэлбэр нь дараах байдлаар байна:

```
# make -x -DVARIABLE target
```

Энэ жишээн дээр `-x` нь make(1) уруу таны дамжуулах тохируулга юм. make(1)-н гарын авлагын хуудаснаас та дамжуулж болох тохируулгуудын жишээг үзнэ үү.

`-D_VARIABLE_` тохируулга нь Makefile уруу хувьсагч дамжуулж байна. Makefile-ийн ажиллагаа эдгээр хувьсагчуудаар хянагдана. Эдгээр нь /etc/make.conf дотор зааж өгсөн хувьсагчуудтай адил бөгөөд энэ нь тэдгээрийг тохируулах бас нэг өөр арга юм.

```
# make -DNO_PROFILE target
```

тушаал нь профиль хийгдсэн сангууд бүтээгдэх ёсгүйг заах өөр нэг арга бөгөөд энэ нь /etc/make.conf дахь дараах

```
NO_PROFILE=    true     #    Avoid compiling profiled libraries
```

мөрд харгалзах юм.

*target* нь make(1)-д таны юу хийхийг хэлж өгдөг. Makefile болгон өөр өөр "targets буюу даалгаврын төрлүүдийг" тодорхойлдог бөгөөд таны сонгосон төрөл юу болохыг тодорхойлдог.

Зарим төрлүүд Makefile-д жагсаагдсан байх бөгөөд гэхдээ эдгээр нь таныг ажиллуулахад зориулагдаагүй. Харин тэдгээр нь системийг дахин бүтээхэд шаардлагатай алхмуудыг хэд хэдэн дэд алхмуудад хуваахын тулд бүтээх процессод хэрэглэгддэг.

Ихэнх тохиолдолд та make(1) уруу ямар ч параметр дамжуулах хэрэггүй бөгөөд тэгэхээр таны тушаал дараахтай ижил байж болно:

```
# make target
```

дээрх *target* нь олон бүтээх тохируулгуудын нэг болно. Эхний төрөл нь үргэлж `buildworld` байх ёстой.

Нэртэйгээ адилаар `buildworld` нь /usr/obj дотор бүрэн гүйцэд шинэ модыг бүтээх бөгөөд өөр нэг төрөл болох `installworld` нь энэ модыг тухайн машин дээр суулгадаг.

Тусдаа тохируулгуудтай байх нь хоёр шалтгаанаар маш ач холбогдолтой юм. Нэгдүгээрт энэ нь бүтээлтийг таны ажиллаж байгаа системийн ямар ч хэсэгт нөлөөлөхгүйгээр аюулгүйгээр хийхийг танд зөвшөөрдөг. Бүтээлт нь "өөр дээрээ хийгдэнэ (self hosted)". Ийм болохоор та `buildworld` тушаалыг олон хэрэглэгчийн горимд ажиллаж байгаа машин дээр буруу нөлөөллөөс айлгүйгээр аюулгүйгээр хийж болно. Гэхдээ `installworld` хэсгийн хувьд ганц хэрэглэгчийн горимд хийхийг танд зөвлөдөг.

Хоёрдугаарт энэ нь сүлжээн дэх олон машинуудыг шинэчлэхэд NFS холболтуудыг ашиглахыг танд зөвшөөрдөг. Хэрэв танд гурван машин байгаа бөгөөд `A`, `B` болон `C` машинуудыг шинэчлэхийг хүсвэл `make buildworld` болон `make installworld` тушаалыг `A` дээр ажиллуулна. Дараа нь `B` болон `C` машинууд `A` дээрх /usr/src болон /usr/obj сангуудыг NFS холболт хийн `make installworld`-г ажиллуулж бүтээлтийн үр дүнг `B` болон `C` дээр суулгаж болох юм.

`world` төрөл байсаар байгаа хэдий ч танд түүнийг ашиглахгүй байхыг зөвлөж байна.

Дараах тушаалыг ажиллуул

```
# make buildworld
```

Хэд хэдэн зэрэгцээ процессуудыг үүсгэх `-j` тохируулгыг `make` тушаалд зааж өгөх боломжтой. Энэ нь олон CPU-тэй машинууд дээр хамгийн их ашигтай. Гэхдээ эмхэтгэх процессийн ихэнх нь CPU дээр биш IO дээр ажилладаг болохоор энэ нь бас нэг CPU-тэй машинууд дээр ашигтай юм.

Ердийн нэг CPU-тэй машин дээр та доор дурдсаныг ажиллуулж болох юм:

```
# make -j4 buildworld
```

make(1) нь 4 хүртэлх процессийг нэгэн зэрэг ажиллуулах юм. Захидлын жагсаалтуудад илгээгдсэн туршлагаас харахад энэ нь ерөнхийдөө ажиллагааг хамгийн сайн хангаж хурдасгадаг байна.

Хэрэв та олон CPU машинтай бөгөөд SMP тохируулагдсан цөм ашиглаж байвал утгыг 6-аас 10 хүртэл болгож хэр хурдсаж байгааг хараарай.

#### 25.7.7.3. Хугацаа

Бүтээхэд шаардагдах хугацаанд олон хүчин зүйлс нөлөөлдөг, гэхдээ нэлээн сүүлийн үеийн машинуудын хувьд FreeBSD-STABLE модыг процессийн явцад ямар нэгэн заль мэх эсвэл дөт зам ашиглалгүйгээр бүтээхэд зөвхөн нэг юм уу эсвэл хоёр цаг л шаардагдах болох юм. FreeBSD-CURRENT модны хувьд арай удах болов уу.

### 25.7.8. Шинэ цөмийг эмхэтгэж суулга

Та өөрийн шинэ системийн давуу талыг бүгдийг нь авахын тулд цөмөө дахин эмхэтгэх хэрэгтэй. Зарим нэг санах ойн бүтцүүд өөрчлөгдсөн байх талтай бөгөөд ps(1) болон top(1) зэрэг програмууд нь цөм болон эх кодын хувилбарууд адил болтол ажилладаггүй болохоор эмхэтгэх нь үнэндээ чухал хэрэгцээтэй юм.

Үүнийг хамгийн хялбараар аюулгүйгээр хийхийн тулд GENERIC дээр тулгуурласан цөмийг бүтээж суулгах явдал юм. GENERIC нь таны системийн хувьд хэрэгцээтэй төхөөрөмжүүдийг агуулаагүй байж болох боловч таны системийг ядаж ганц хэрэглэгчийн горимд ачаалахад шаардлагатай бүгдийг агуулсан байх ёстой. Шинэ систем зөв ажиллуулахад энэ сайн тест болж өгдөг. GENERIC-с ачаалж таны систем ажиллаж байгааг шалгасны дараа та өөрийн ердийн цөмийн тохиргооны файл дээр тулгуурлан шинэ цөмөө бүтээж болох юм.

FreeBSD дээр шинэ цөм бүтээхээсээ өмнө ертөнцийг бүтээх нь чухал юм.

Хэрэв та өөрчлөн тохируулсан цөмийг бүтээхийг хүсэж тохиргооны файлаа аль хэдийн үүсгэсэн бол доор дурдсантай адилаар `KERNCONF=MYKERNEL` гэж ашиглаарай:

```
# cd /usr/src
# make buildkernel KERNCONF=MYKERNEL
# make installkernel KERNCONF=MYKERNEL
```

Хэрэв та `kern.securelevel` хувьсагчийг 1-ээс дээш болгон ихэсгэсэн *бөгөөд* `noschg` эсвэл түүнтэй адил тугуудыг өөрийн цөмийн хоёртын файлд тавьсан бол `installkernel` хийхийн тулд та ганц хэрэглэгчийн горимд шилжин орох шаардлагатай байж болох юм. Үгүй бол та энэ хоёр тушаалыг олон хэрэглэгчийн горимоос ямар ч асуудалгүйгээр ажиллуулах ёстой. `kern.securelevel`-ийн талаар дэлгэрэнгүйг init(8) болон төрөл бүрийн файлын тугуудын

талаар дэлгэрэнгүйг chflags(1) гарын авлагын хуудаснуудаас үзнэ үү.

### 25.7.9. Ганц хэрэглэгчийн горим уруу дахин ачаалан ор

Та шинэ цөмийн ажиллагааг шалгахын тулд ганц хэрэглэгчийн горимд дахин ачаалан орох хэрэгтэй. Үүнийг Ганц хэрэглэгчийн горимд шилж дахь зааврууд дагуу хийнэ.

### 25.7.10. Шинэ системийн хоёртын файлуудыг суулга

Та шинэ системийн хоёртын файлуудыг суулгахын тулд `installworld` тушаалыг ашиглах шаардлагатай.

Доор дурдсаныг ажиллуулна

```
# cd /usr/src
# make installworld
```

Хэрэв та `make buildworld` тушаалын мөрөнд хувьсагчуудыг зааж өгсөн бол тэдгээр хувьсагчуудыг `make installworld` тушаалын мөрөнд бас адилаар зааж өгөх хэрэгтэй. Энэ бусад тохируулгуудын хувьд заавал шаардлагатай биш байж болох юм; жишээ нь `-j` тохируулга `installworld`-той цуг хэзээ ч хэрэглэгдэх ёсгүй.

Жишээ нь хэрэв та доор дурдсаныг ажиллуулсан бол:

```
# make -DNO_PROFILE buildworld
```

хоёртын файлуудыг дараах тушаалаар суулгана:

```
# make -DNO_PROFILE installworld
```

ингэхгүй бол `make buildworld` тушаалын ажиллах явцад бүтээгдээгүй профиль хийгдсэн сангуудыг (libraries) суулгахыг оролдох болно.

### 25.7.11. `make installworld` тушаалаар шинэчлэгдээгүй файлуудыг шинэчил

Ертөнцийг дахин бүтээх нь зарим нэг сангуудыг (ялангуяа /etc, /var болон /usr) шинэ болон өөрчлөгдсөн тохиргооны файлуудаар шинэчилдэггүй.

Эдгээр файлуудыг хамгийн амархнаар шинэчлэх арга нь mergemaster(8)-г ашиглах явдал юм, гэхдээ та хэрэв хүсвэл үүнийг гараар ажиллуулах боломжтой юм. Аль ч аргыг сонголоо гэсэн ямар нэгэн зүйл буруутсан тохиолдолд сэргээх боломжтойгоор /etc-г нөөцөлж авах нь зүйтэй юм.

#### 25.7.11.1. `mergemaster`

mergemaster(8) хэрэгсэл нь /etc дэх таны тохиргооны файлууд болон /usr/src/etc эх модон дахь тохиргооны файлуудын ялгааг тодорхойлоход танд тусалдаг Bourne скрипт юм. Энэ нь системийн тохиргооны файлуудыг эх модон дахь тохиргооны файлуудаар шинэчлэх зориулалттай бидний зөвлөдөг шийдэл юм.

Эхлэхийн тулд өөрийн тушаал оруулах мөрөнд ердөө л `mergemaster`-г бичиж түүний эхлэхийг нь хараарай. `mergemaster` нь түр зуурын root орчныг /-с доошлуулан бүтээж төрөл бүрийн системийн тохиргооны файлуудаар дамждаг. Тэдгээр файлууд нь таны системд суулгагдсан файлуудтай харьцуулагддаг. Энэ үед хоорондоо ялгаатай файлууд diff(1) хэлбэрээр үзүүлэгддэг бөгөөд `+` тэмдэгтээр нэмэгдсэн эсвэл өөрчлөгдсөн мөрүүдийг `-` тэмдэгтээр устгагдсан эсвэл шинэ мөрөөр солигдсон мөрүүдийг харуулдаг. diff(1)-н синтакс болон файлын өөрчлөлтүүдийг хэрхэн үзүүлдэг талаар дэлгэрэнгүй мэдээллийг diff(1) гарын авлагын хуудаснаас үзнэ үү.

mergemaster(8) нь зөрчилдөөнүүдийг үзүүлсэн файл болгоныг харуулдаг бөгөөд энэ үед танд шинэ файлыг устгах (түр зуурын файл гэгддэг), түр зуурын файлыг өөрчлөлгүйгээр суулгах, суусан байгаа файлтай түр зуурын файлыг нийлүүлэх эсвэл diff(1)-н гаралтыг дахин харах сонголтыг үзүүлэх болно.

Түр зуурын файлыг устгахыг сонгосноор бид одоо байгаа файлаа хэвээр өөрчлөлгүй үлдээж шинэ хувилбарыг устгахыг хүсэж байгаагаа mergemaster(8)-д хэлж байна гэсэн үг юм. Хэрэв та одоо байгаа файлаа өөрчлөх шалтгааныг олж харахгүй байгаагаас бусад тохиолдолд энэ сонголтыг хийхийг зөвлөдөггүй. Та ямар ч үед mergemaster(8) тушаал хүлээх мөрөн дээр `?` гэж бичин тусламж авч болох юм. Хэрэв хэрэглэгч файлыг орхихоор сонгосон бол энэ нь бусад бүх файлуудтай ажилсны дараа дахин үзүүлэгдэн хэрэглэгчээс тушаал хүлээх болно.

Өөрчлөгдөөгүй түр зуурын файлыг суулгахыг сонгосноор одоо байгаа файлыг шинээр сольдог. Ихэнх өөрчлөгдөөгүй файлуудын хувьд энэ нь хамгийн шилдэг сонголт юм.

Файлыг нийлүүлэхийг сонгосноор текст засварлагч болон хоёр файлын агуулгыг танд харуулах болно. Та дэлгэцийн хоёр талд байрласан тэдгээр хоёр файлыг хоёуланг нь шалган аль аль талаас нь хэрэгтэй хэсгүүдийг сонгон эцсийн бүтээгдэхүүн гаргаж аван нийлүүлж болно. Файлууд нь дэлгэцийн хоёр талд байрлан харьцуулагдах явцад `l` түлхүүр таны зүүн талын агуулгыг сонгодог бол `r` түлхүүр нь таны баруун тал дахь агуулгыг сонгох юм. Гарах эцсийн үр дүн нь хоёр файлын хоёулангийн хэсгүүдийг агуулсан файл болох бөгөөд түүнийг дараа нь суулгах боломжтой болох юм. Энэ сонголтыг хэрэглэгчийн тохиргоонуудад хийгдсэн өөрчлөлтүүдтэй файлуудын хувьд хэрэглэх нь зуршил болжээ.

diff(1)-ээс гарах үр дүнг дахин харахыг сонгосноор өмнө нь mergemaster(8) файлын өөрчлөлтүүдийг харуулан таны сонголтыг хүлээсний нэгэн адилыг дахин харуулдаг.

mergemaster(8) системийн файлуудтай ажиллаж дууссаны дараа танаас бусад сонголтуудыг хийхийг хүлээдэг. mergemaster(8) тушаал нууц үгийн файлыг дахин бүтээхийг хүсэж байгаа эсэхийг танаас асууж үлдсэн түр зуурын файлуудыг устгах сонголтыг үзүүлэн дуусдаг.

#### 25.7.11.2. Гараар шинэчлэх

Хэрэв та гараар шинэчлэхийг хүсвэл гэхдээ та /usr/src/etc сангаас /etc сан уруу файлуудыг

зүгээр л дарж хуулж ажиллуулж чадахгүй. Зарим файлуудыг эхлээд "суулгах" хэрэгтэй. Учир нь /usr/src/etc сан таны /etc сангийн хуулбар шиг байхаар *харагддагүй*. Мөн /usr/src/etc санд байдаггүй хэрнээ /etc сан дотор байх шаардлагатай зарим файлууд байдаг.

Хэрэв та mergemaster(8) (зөвлөсний дагуу) ашиглаж байвал та дагаагийн хэсэг уруу орж болно.

Үүнийг гараар хамгийн хялбар аргаар хийхийн тулд файлуудыг шинэ сан уруу суулгаж нэг бүрчлэн өөрчлөлтүүдийг хайн ажиллах хэрэгтэй.

*Өөрт байгаа /etc-г нөөцөл*

Онолоор бол автоматаар энэ санд юу ч хүрдэггүй ч үүнд үргэлж итгэлтэй байх хэрэгтэй. Тэгэхээр өөрийн байгаа /etc санг хаа нэг аюулгүй газар хуулах хэрэгтэй. Доорхтой адилаар:

```
# cp -Rp /etc /etc.old
```

`-R` нь рекурсив хуулбар хийх бөгөөд `-p` нь файлуудын хугацаа, эзэмшигч гэх мэтийг хадгалдаг.

Та шинэ /etc болон бусад файлуудыг суулгахын тулд хоосон сангууд бүтээх хэрэгтэй. /var/tmp/root нь боломжийн сонголт болох бөгөөд энэ сангийн доор хэд хэдэн дэд сангууд бас шаардлагатай болно.

```
# mkdir /var/tmp/root
# cd /usr/src/etc
# make DESTDIR=/var/tmp/root distrib-dirs distribution
```

Энэ нь шаардлагатай сангийн бүтцийг бүтээж файлуудыг суулгадаг. /var/tmp/root дотор үүсгэгдсэн олон дэд сангууд хоосон бөгөөд тэдгээрийг устгах шаардлагатай байдаг. Үүнийг хамгийн хялбараар хийхийн тулд:

```
# cd /var/tmp/root
# find -d . -type d | xargs rmdir 2>/dev/null
```

Энэ нь бүх хоосон сангуудыг устгана. (Хоосон биш сангуудын тухай анхааруулгуудыг гаргахгүйн тулд стандарт алдаа нь /dev/null уруу илгээгддэг.)

Одоо /var/tmp/root нь /-с доор байрлах тохирох байрлалуудад байршуулах ёстой бүх файлуудыг агуулсан байх болно. Та одоо эдгээр файл бүрийг шалгаж танд байгаа файлуудаас хэрхэн ялгаатай болохыг тогтоох хэрэгтэй.

/var/tmp/root дотор суулгагдсан зарим файлуудын нэр урдаа "." тэмдэгттэй байдгийг анхаарна уу. Энэ баримтыг бичиж байх үед ийм файлуудтай адил файлууд /var/tmp/root/ болон /var/tmp/root/root/ сан дахь бүрхүүлийн эхлүүлэх файлууд байсан, гэхдээ (таны хэзээ үүнийг уншиж байгаагаас хамаарч) өөр бусад файлууд байхыг үгүйсгэхгүй. Тэдгээрийг олж

харахын тулд `ls -a` тушаалыг заавал ашиглаарай.

Үүнийг хамгийн хялбар аргаар хийж хоёр файлыг харьцуулахын тулд diff(1) тушаалыг ашиглах явдал юм:

```
# diff /etc/shells /var/tmp/root/etc/shells
```

Энэ нь таны /etc/shells файл болон шинэ /var/tmp/root/etc/shells файлын хоорондын ялгааг харуулна. Эдгээрийг ашиглаж өөрийн хийсэн өөрчлөлтүүдийг нийлүүлэх эсвэл өөрийн хуучин файл дээрээс хуулах эсэхээ шийдээрэй.

*Хувилбаруудын Хооронд Ялгаануудыг Хялбараар Харьцуулахын Тулд Та Шинэ Root Сангаа Тухайн Үеийн Хугацаагаар Нэрлээрэй*

Ертөнцийг байнга дахин бүтээнэ гэдэг нь /etc-г та бас байнга шинэчилнэ гэсэн үг бөгөөд энэ нь ердөө л жижиг хэвшмэл ажил юм.

Та энэ процессийг /etc уруу нийлүүлсэн өөрийн хамгийн сүүлийн өөрчлөгдсөн файлуудыг хадгалснаар хурдасгаж болох юм. Дараах процедур үүнийг хэрхэн хийж болох нэг санааг өгч байна.

1. Ертөнцийг жирийнээр бүтээ. /etc болон бусад сангуудыг шинэчлэхийг хүсэхдээ тухайн цаг дээр тулгуурласан нэр бүхий санг өг. Хэрэв та үүнийг 1998 оны 2 сарын 14-нд хийж байгаа бол дараах байдлаар хийнэ:

   ```
   # mkdir /var/tmp/root-19980214
   # cd /usr/src/etc
   # make DESTDIR=/var/tmp/root-19980214 \
       distrib-dirs distribution
   ```

2. Энэ сангийн өөрчлөлтүүдийг дээр дурдсаны дагуу нийлүүл.

   Та дууссаныхаа дараа /var/tmp/root-19980214 санг *битгий* устгаарай.

3. Та эхийн хамгийн сүүлийн хувилбарыг татан авч дахин бүтээхдээ 1-р алхмыг дага. Энэ нь танд шинэ сан өгөх бөгөөд /var/tmp/root-19980221 гэж нэрлэгдсэн байж болох юм (хэрэв та шинэчлэлтүүдийг хийхдээ долоо хоног хүлээсэн бол).

4. Та одоо diff(1) ашиглан хоёр сангийн хоорондрекурсив diff үүсгэж долоо хоногийн хоорондхийгдсэн өөрчлөлтүүдийг харж болно:

   ```
   # cd /var/tmp
   # diff -r root-19980214 root-19980221
   ```

   Ихэнхдээ энэ нь /var/tmp/root-19980221/etc болон /etc хоёрын

хоорондох өөрчлөлтүүдийг бодох юм бол харьцангуй бага өөрчлөлтүүд байдаг. Өөрчлөлтүүд нь арай бага болохоор тэдгээр өөрчлөлтүүдийг өөрийн /etc сан уруу шилжүүлэх нь илүү хялбар байдаг.

5. Та одоо хоёр /var/tmp/root-* сангуудын аль хуучныг устгаж болно:

```
# rm -rf /var/tmp/root-19980214
```

6. /etc уруу өөрчлөлтүүдийг нийлүүлэх болгондоо энэ процессийг давтах хэрэгтэй.

Та date(1)-г ашиглан сангийн нэрсийг автоматаар үүсгэж болно:

```
# mkdir /var/tmp/root-`date "+%Y%m%d"`
```

## 25.7.12. Дахин ачаалах нь

Та ерөнхийдөө ингээд хийгээд дуусч байна. Та бүх зүйл байх ёстой байрандаа байгаа эсэхийг шалгасныхаа дараа системийг дахин ачаалж болно. Энгийн shutdown(8) үүнийг хийх болно:

```
# shutdown -r now
```

## 25.7.13. Дууслаа

Одоо та өөрийн FreeBSD системийг амжилттайгаар шинэчлээд дууссан байх ёстой. Баяр хүргэе.

Хэрэв юмс шал буруугаар эргэвэл системийн тухайн хэсгийг дахин бүтээхэд амархан байдаг. Жишээ нь хэрэв та шинэчлэлтийн явцад эсвэл /etc-г нийлүүлэх явцад санамсаргүйгээр /etc/magic файлыг устгасан бол file(1) тушаал ажиллахаа больно. Ийм тохиолдолд дараах засварыг ажиллуулж болох юм:

```
# cd /usr/src/usr.bin/file
# make all install
```

## 25.7.14. Асуултууд

### 25.7.14.1. Өөрчлөлт бүрт зориулан ертөнцийг дахин бүтээх хэрэгтэй юу?

Үүнд хялбар хариулт байхгүй, учир нь өөрчлөлтийн цаад утга чанараас хамаарна. Жишээ нь хэрэв та CVSup-г дөнгөж ажиллуулахад дараах файлууд шинэчлэгдэж байгааг үзүүлж байгаа бол:

```
src/games/cribbage/instr.c
src/games/sail/pl_main.c
src/release/sysinstall/config.c
src/release/sysinstall/media.c
src/shared/mk/bsd.port.mk
```

магадгүй бүхэл ертөнцийг дахин бүтээх хэрэггүй байж болох юм. Та тохирох дэд сангууд уруу орж `make all install` гэж тушаалыг өгөөд л болох юм. Хэрэв зарим нэг гол чухал зүйл жишээ нь src/lib/libc/stdlib өөрчлөгдсөн бол та ертөнцийг эсвэл хамгийн багаар бодоход статикаар холбогдсон (statically linked) түүний тэдгээр хэсгүүдийг дахин бүтээх шаардлагатай болно.

Эцсийн эцэст энэ нь танаас л хамаарна. Та жишээ нь хоёр долоо хоног тутам ертөнцийг дахин бүтээж тэр хоёр долоо хоногийн хугацаанд өөрчлөлтүүдийг хуримтлуулж байгаадаа сэтгэл хангалуун байж болно. Эсвэл та зөвхөн өөрчлөгдсөн зүйлсүүдийг дахин бүтээхийг хүсэж магадгүй бөгөөд бүх хамаарлуудыг шийднэ гэдэгтээ итгэлтэй байх хэрэгтэй.

Тэгээд мэдээж энэ бүхэн таны ямар давтамжтайгаар шинэчлэхийг хүсдэг болон FreeBSD-STABLE эсвэл FreeBSD-CURRENT-ийн алийг дагаж байгаагаас хамаарах болно.

#### 25.7.14.2. Миний эмхэтгэл маш олон дохио 11 дохио 11 (эсвэл бусад дохионы дугаар) алдаагаар амжилтгүй болсон. Юу болсон юм бол?

Энэ нь ихэвчлэн тоног төхөөрөмжийн асуудлыг илэрхийлдэг. Ертөнцийг (дахин) бүтээх нь өөрийн тоног төхөөрөмжийг ачаалах тест хийх үр дүнтэй арга бөгөөд удаа дараа санах ойн асуудлууд байвал тэдгээрийг илрүүлдэг. Эмхэтгэгч нь сонин/хачин дохионуудыг хүлээн авч ид шидийн байдлаар амжилтгүй болсноор тэдгээр асуудлууд нь өөрсдийгөө зарлан тунхагладаг.

Хэрэв та бүтээлтийг дахин эхлүүлээд тэр нь процессийн өөр өөр хэсэгт амжилтгүй болж байвал энэ нь үүнийг тодоор зааж байна гэсэн үг юм.

Энэ тохиолдолд та өөрийн машин дахь бүрэлдэхүүн хэсгүүдээ өөрчлөн нэгээс нөгөөд сольж тавин аль нь ажиллахгүй байгааг олохоос өөр зүйл хийж чадахгүй л болов уу.

#### 25.7.14.3. Би дууссаныхаа дараа /usr/obj-г устгаж болох уу?

Товчхондоо бол болно.

/usr/obj нь эмхэтгэх үед бүтээгдсэн бүх обьект файлуудыг агуулдаг. Жирийн үед `make buildworld` процессийн эхний алхмуудын нэг нь энэ санг устгаад цоо шинээр эхлэх явдал юм. Энэ тохиолдолд /usr/obj-г дууссаныхаа дараа байлгаад байх нь ухаалаг биш бөгөөд үүнийг устгаснаар ихээхэн хэмжээний дискний зайг суллах болно (одоогоор 2 GB орчим).

Гэхдээ хэрэв та юу хийж байгаагаа мэдэж байгаа бол `make buildworld` хийхдээ энэ алхмыг алгасаж болно. Энэ нь дараа дараагийн бүтээлтийг илүү хурдасгадаг бөгөөд учир нь ихэнх эхүүд дахин эмхэтгэх шаардлагагүй байдаг. Үүний сул тал нь баригдашгүй хамаарлын асуудлууд илэрч таны бүтээлтийг хачин байдлаар амжилтгүй болгодог. Хэн нэгэн илүү дөтлөх гэснээсээ болоод амжилтгүй болсныг мэдэлгүй өөрийн бүтээлтийг амжилтгүй

болсныг гомдоллосноор FreeBSD-ийн захидлын жагсаалтуудад хий дэмий шуугианыг удаа дараа үүсгэдэг билээ.

#### 25.7.14.4. Тасалдсан бүтээлтүүдийг үргэлжлүүлж болох уу?

Энэ нь асуудлыг олох хүртлээ та хэр хол явснаас хамаарна.

*Ерөнхийдөө* (энэ нь хэцүү бас хурдан дүрэм биш) `make buildworld` процесс нь үндсэн багажуудын (gcc(1), болон make(1) зэрэг) болон системийн сангуудын шинэ хуулбаруудыг бүтээдэг. Тэдгээр багажууд болон сангууд нь дараа нь суулгагддаг. Шинэ багажууд болон сангууд дараа нь өөрсдийгөө дахин бүтээхэд ашиглагддаг бөгөөд дахин суулгагддаг. Бүхэл бүтэн систем (одоо ls(1) эсвэл grep(1) зэрэг ердийн хэрэглэгчийн програмууд) дараа нь шинэ системийн файлуудтайгаар дахин бүтээгддэг.

Хэрэв та сүүлийн шатанд байгаа бөгөөд та үүнийг мэдэж байгаа бол (та хадгалж байгаа гаралтаас харсан болохоор) та дараах тушаалыг ажиллуулж (бараг л аюулгүйгээр) болно:

```
... fix the problem ...
# cd /usr/src
# make -DNO_CLEAN all
```

Энэ нь өмнөх `make buildworld` тушаалын хийснийг буцаахгүй.

Хэрэв та доорх мэдэгдлийг :

```
--------------------------------------------------------------
Building everything..
--------------------------------------------------------------
```

`make buildworld` тушаалын гаралт дээр харсан бол магадгүй тэгж хийх нь аюулгүй байж болох юм.

Хэрэв та тийм мэдэгдэл харахгүй байгаа бол эсвэл та итгэлтэй биш байгаа бол харамсахаасаа өмнө аюулгүй байдлыг бодож бүтээлтийг бүр эхнээс нь дахин эхлүүлсэн нь дээр юм.

#### 25.7.14.5. Би ертөнцийг бүтээхийг хэрхэн хурдасгах вэ?

- Ганц хэрэглэгчийн горимд ажиллуул.
- /usr/src болон /usr/obj сангуудыг тус тусдаа байх дискнүүд дээр тус тусдаа байх файлын системүүд дээр байрлуул. Хэрэв боломжтой бол эдгээр дискнүүдийг тус тусад нь дискний хянагчууд дээр байрлуул.
- ccd(4) (нийлүүлсэн дискний драйвер) төхөөрөмж ашиглан эдгээр файлын системүүдийг олон дискнүүдийн дагуу байрлуулах нь бас арай илүү хурдасгах юм.
- Профиль хийгдэхийг (/etc/make.conf файлд "NO_PROFILE=true" гэж зааж өг) болиул. Танд энэ бараг гарцаагүй хэрэггүй.

- `-jn` тохируулгыг make(1)-д дамжуулж олон процессийг зэрэгцээгээр ажиллуул. Энэ нь танд ганц эсвэл олон процессортой машин аль нь ч байсан ялгаагүйгээр ихэвчлэн тусалдаг.
- /usr/src-г агуулж байгаа файлын систем `noatime` тохируулгаар холболт хийгдэж (эсвэл салгагдаж) болно. Энэ нь файлын систем файл уруу хандах хандалтын хугацааг бүртгэхийг болиулдаг. Танд магадгүй энэ мэдээлэл бараг л хэрэггүй биз ээ.

```
# mount -u -o noatime /usr/src
```

Энэ жишээ /usr/src нь өөрийн файлын систем дээр байгаа гэж тооцож байгаа болно. Хэрэв энэ нь тийм биш бол (хэрэв энэ сан жишээ нь /usr -ийн хэсэг маягаар байгаа бол) та /usr/src-г биш харин тэр файлын системээ холболтын цэг болгон ашиглах хэрэгтэй.

- /usr/obj-г агуулж байгаа файлын систем `async` тохируулгатай холболт хийгдэж (эсвэл салгагдаж) болно. Энэ нь диск уруу хийх бичилтийг асинхроноор буюу зэрэг биш хийлгэдэг. Өөрөөр хэлбэл бичилт нэн даруй хийгдээд өгөгдөл диск уруу цөөн секундын дараа бичигддэг. Энэ нь бичилтүүдийг бүлэглэхийг зөвшөөрч маш их үр дүнтэйгээр ажиллагааг хурдасгаж болох юм.

Энэ тохируулга нь таны файлын системийг илүү эмзэг болгохыг санаарай. Тэжээл тасалдаж машин дахин ачаалах үед файлын систем сэргээж болшгүй төлөвт орох магадлал энэ тохируулгатай байхад илүү байдаг.

Хэрэв /usr/obj нь энэ файлын систем дээрх цорын ганц зүйл бол энэ асуудал биш юм. Хэрэв танд уг файлын систем дээр өөр, үнэтэй өгөгдөл байгаа бол энэ тохируулгыг идэвхжүүлэхээсээ өмнө өөрийн нөөц чинь шинэ эсэхийг шалгаарай.

```
# mount -u -o async /usr/obj
```

Дээр дурдсан шиг хэрэв /usr/obj нь өөрийн файлын систем дээр биш байх юм бол жишээн дээрхийг тохирох холболт хийх цэгийн нэрээр солиорой.

#### 25.7.14.6. Хэрэв ямар нэг юм буруутвал би юу хийх вэ?

Таны орчинд өмнөх бүтээлтүүдийн үеийн илүү үлдэгдлүүд байхгүйд үнэхээр итгэлтэй байх хэрэгтэй. Энэ нь их амархан юм.

```
# chflags -R noschg /usr/obj/usr
# rm -rf /usr/obj/usr
# cd /usr/src
# make cleandir
```

```
# make cleandir
```

Тиймээ, `make cleandir` тушаалыг үнэндээ хоёр удаа ажиллуулах шаардлагатай.

Тэгээд `make buildworld` тушаалыг эхлүүлж бүх процессийг дахин эхлүүл.

Хэрэв та асуудалтай хэвээр байгаа бол алдаа болон `uname -a` тушаалын дүнг FreeBSD ерөнхий асуултууд захидлын жагсаалт уруу явуулаарай. Өөрийн тохиргооныхоо талаар бусад асуултанд хариулахад бэлэн байгаарай!

# 25.8. Хуучин файлууд, хавтаснууд болон сангуудыг устгах

FreeBSD хөгжүүлэлтийн явцад файлууд болон тэдгээрийн агуулга үе үе хуучирдаг.Тэдгээрийн үүрэг болон боломжууд өөр хаа нэгтээ хийгдсэн юм уу эсвэл сангийн хувилбарын дугаар өөрчлөгдсөн юм уу эсвэл системээс бүрмөсөн хасагдсанаас болоод тийм байж болох юм. Эдгээрт хуучин файлууд, сангууд болон хавтаснууд ордог бөгөөд эдгээрийг системийг шинэчлэхдээ устгах ёстой. Энэ нь хэрэглэгчийн хувьд хадгалах (болон нөөц) төхөөрөмж дээр хэрэгцээгүй зай эзлээд байгаа хуучин файлуудаар систем дүүрэхгүй байх ашигтай юм. Үүнээс гадна хуучин сан аюулгүй байдлын болон найдвартай ажиллагааны хувьд асуудалтай байсан бол та өөрийн системийг аюулгүй болгож хуучин сангаас болоод ажиллахаа болиод байсан асуудлаас сэргийлэхийн тулд шинэ сан руу шинэчлэх хэрэгтэй. Хуучин гэгдсэн файлууд, хавтаснууд, сангуудын жагсаалт /usr/src/ObsoleteFiles.inc файлд байдаг. Дараах заавар нь системийг шинэчлэх явцад хуучин файлуудыг устгахад туслах болно.

Таныг Шалгагдсан аргаар өөрийн системийг шинэчлэх нь-д заасны дагуу явж байгаа гэж энд үзнэ. `make installworld` болон дараагийн `mergemaster` тушаал амжилттай хэрэгжсэний дараа дараах маягаар та хуучин файлууд болон сангуудыг шалгах ёстой:

```
# cd /usr/src
# make check-old
```

Хэрэв ямар нэг хуучин файл олдвол дараах тушаал ашиглан тэдгээрийг устгаж болно:

```
# make delete-old
```

Түлхүүр үгсийн талаар дэлгэрэнгүйг сонирхож байгаа бол /usr/src/Makefile файлыг үзнэ үү.

Хуучин файл бүрийг устгахын өмнө асууж хариулах дэлгэц гарна. Та энэ дэлгэцийг өнгөрөөж систем эдгээр файлуудыг автоматаар устгахаар тохируулахын тулд `BATCH_DELETE_OLD_FILES` хувьсагчийг дараах байдлаар ашиглана:

```
# make -DBATCH_DELETE_OLD_FILES delete-old
```

Мөн энэ зорилгод хүрэхийн тулд доорхитой адилаар эдгээр тушаалд `yes` өгч хүрч болно:

```
# yes|make delete-old
```

*Анхааруулга*

Хуучин файлуудыг устгах нь тэдгээр хуучин файлуудаас хамааралтай програмуудыг ажиллахгүй болгоно. Энэ нь ялангуяа хуучин сангуудын хувьд үнэн байдаг. Ихэнх тохиолдолд та `make delete-old-libs` тушаалыг биелүүлэхээсээ өмнө хуучин сан ашиглаж байсан програмууд, портууд эсвэл сангуудыг дахин бүтээх хэрэгтэй.

Хуваалцсан сангуудаас хамааралтай эсэхийг шалгадаг хэрэгслүүд sysutils/libchk эсвэл sysutils/bsdadminscripts зэрэг портын цуглуулгад байдаг.

Хуучин хуваалцсан сангууд нь шинэ сангуудтай зөрчилдөж болох бөгөөд доорх шиг алдаа өгч болно:

```
/usr/bin/ld: warning: libz.so.4, needed by /usr/local/lib/libtiff.so, may conflict
with libz.so.5
/usr/bin/ld: warning: librpcsvc.so.4, needed by /usr/local/lib/libXext.so, may
conflict with librpcsvc.so.5
```

Эдгээр асуудлуудыг шийдэхийн тулд уг санг аль порт суулгасныг олно:

```
# pkg_info -W  /usr/local/lib/libtiff.so
/usr/local/lib/libtiff.so was installed by package tiff-3.9.4
# pkg_info -W /usr/local/lib/libXext.so
/usr/local/lib/libXext.so was installed by package libXext-1.1.1,1
```

Дараа нь уг портыг deinstall хийгээд дахин бүтээж суулгах хэрэгтэй. Энэ явцыг автоматжуулахын тулд ports-mgmt/portmaster болон ports-mgmt/portupgrade хэрэгслийг ашиглаж болно. Бүх портуудыг дахин бүтээсэн гэдэгтээ итгэлтэй болсны дараа хуучин сангуудыг ашиглах хэрэггүй бөгөөд тэдгээрийг дараах тушаал ашиглан устгаж болно:

```
# make delete-old-libs
```

## 25.9. Олон машины хувьд дагах нь

Хэрэв та олон машинуудын хувьд ижил эх модыг дагахыг хүсэж бүгдийн хувьд эхийг татан авахуулж бүгдийг дахин бүтээхийг хүсэж байгаа бол энэ нь дискний зай, сүлжээний зурвасын өргөн болон CPU циклүүд зэрэг эх үүсвэрүүдийг үр ашиггүйгээр ашиглахад

хүргэхээр санагдаж болох юм. Тиймээ, үүний шийдэл нь нэг машинаар ихэнх ажлыг хийлгэж бусад машинууд нь тэр ажлыг NFS-ээр дамжуулан холбох явдал юм. Энэ хэсэгт ингэж хийх аргыг тайлбарсан.

### 25.9.1. Бэлтгэл ажлууд

Эхлээд хоёртын адил файлуудыг ажиллуулах *build set буюу бүтээх олонлог* гэж бидний нэрлэх машинуудыг олох хэрэгтэй. Машин бүр өөрчлөн тохируулсан цөмтэй байж болох бөгөөд гэхдээ тэд ижил хэрэглэгчийн талбарын хоёртын файлуудыг ажиллуулж байх ёстой. Тэр олонлогоос *бүтээх машин*ыг сонгох хэрэгтэй. Энэ нь ертөнц болон цөм бүтээгдэх машин байх юм. Туйлын хүслээр бол энэ нь `make buildworld` болон `make buildkernel` тушаалуудыг ажиллуулахад хангалттай нөөц CPU бүхий хурдан машин байх хэрэгтэй. Та мөн үйлдвэрлэлд ашиглахаас өмнө програм хангамжуудыг тест хийдэг *тест машин* сонгохыг бас хүсэж болох юм. Энэ нь удаан хугацаагаар унтраастай эсвэл зогссон байж болох машин *байх ёстой*. Энэ нь бүтээх машин байж болох юм, гэхдээ заавал биш юм.

Энэ бүтээх олонлог дахь бүх машинууд нь өөр өөрийн машин дээрээсээ ижил цэг дээр /usr/obj болон /usr/src-г холболт хийх хэрэгтэй. Туйлын хүслээр бол энэ нь бүтээх машин дээрх хоёр өөр дискнүүд байж болох бөгөөд гэхдээ эдгээр нь уг машин дээр NFS холболт бас хийгдэж болохоор байж болох юм. Хэрэв танд олон бүтээх олонлогууд байгаа бол /usr/src сан нь нэг бүтээх машин дээр байрлаж бусад дээр нь NFS холболт хийгдсэн байх юм.

Төгсгөлд нь бүтээх олонлогийн бүх машинууд дээрх /etc/make.conf болон /etc/src.conf файлууд бүтээх машиныхтай тохирч байгаа эсэхийг шалгаарай. Энэ нь бүтээх олонлогийн машин бүрийн суулгах үндсэн системийн бүх хэсгүүдийг бүтээх машин хийх ёстой гэсэн үг юм. Мөн бүтээх машин бүр өөрийн цөмийн нэрийг /etc/make.conf файлд `KERNCONF` хувьсагчид заан өгөх ёстой бөгөөд бүтээх машин бүр `KERNCONF` хувьсагчдаа өөрийн цөмийг эхэнд оруулан дараа нь тэдгээрийг жагсаах ёстой байдаг. Бүтээх машин нь машин бүрийн цөмийг бүтээхээр болох юм бол тэдгээрийн тохиргооны файлыг /usr/src/sys/arch/conf санд агуулсан байх шаардлагатай.

### 25.9.2. Үндсэн систем

Одоо бүх юм ингэж хийгдсэний дараа та бүгдийг бүтээхэд бэлэн боллоо. Бүтээх машин дээр Үндсэн системийг эмхэтгэ-д тайлбарласны дагуу цөм болон ертөнцийг бүтээ, гэхдээ юуг ч битгий суулгаарай. Бүтээлт дууссаны дараа тест машин дээр дөнгөж саяхан бүтээсэн цөмөө суулга. Хэрэв энэ машин нь /usr/src болон /usr/obj сангуудыг NFS-ээр холболт хийх гэж байгаа бол та ганц хэрэглэгчийн горимд дахин ачаалахдаа сүлжээг нээж тэдгээрийг холбож өгөх хэрэгтэй. Үүнийг хамгийн хялбараар хийхийн тулд олон хэрэглэгчийн горимд ачаалан `shutdown now` тушаалыг ажиллуулж ганц хэрэглэгчийн горимд орох явдал юм. Тэгэж орсныхоо дараа та шинэ цөм болон ертөнцийг суулгаж жирийн үедээ хийдэг `mergemaster` тушаалыг ажиллуулж болно. Ингэж дууссаныхаа дараа энэ машины хувьд ердийн олон хэрэглэгчийн үйлдлүүдэд дахин ачаалж орно.

Тест машин дээрх бүх зүйлс зөв ажиллаж байгааг мэдсэнийхээ дараа та бүтээх олонлогийн бусад машин бүр дээр шинэ програм хангамж суулгахдаа ижил процедурыг ашиглаарай.

### 25.9.3. Портууд

Үүнтэй адил санааг бас портуудын модонд ашиглаж болно. Эхний чухал алхам бол нөгөө машин дээрх /usr/ports санг бүтээх олонлогийн бусад машинууд дээр холбож өгөх явдал юм. Дараа нь та /etc/make.conf файлыг distfiles буюу түгээлтийн файлуудыг хуваалцахаар зөв тохируулж өгч болно. Та `DISTDIR` хувьсагчийг таны NFS холболтуудад заагдсан аль ч `root` хэрэглэгчийн хувьд бичигдэх боломжтой байх нийтлэг хуваалцсан сангаар тохируулах шаардлагатай. Машин бүр `WRKDIRPREFIX` хувьсагчийг локал бүтээх сангаар зааж өгөх хэрэгтэй. Эцэст нь хэрэв та багцуудыг бүтээж түгээх гэж байгаа бол `PACKAGES` хувьсагчийг `DISTDIR` хувьсагчийн нэгэн адил сангаар зааж өгөх хэрэгтэй.

# Chapter 26. DTrace

## 26.1. Ерөнхий агуулга

DTrace буюу Dynamic Tracing (Динамик Мөрдөлт) нь үйлдвэрлэлд ашиглаж байгаа болон ашиглахын өмнөх системүүдэд үйл ажиллагааны саад тотгоруудыг олоход зориулагдсан Sun™-с хөгжүүлсэн хэрэгсэл юм. Энэ нь ямар ч тохиолдолд дибаг хийгч хэрэгсэл биш боловч үйл ажиллагаа болон бусад асуудлуудыг цаг тухайд нь олох системийн анализ хийхэд зориулагдсан хэрэгсэл юм.

DTrace нь системийн асуудлуудыг оношлоход зориулагдсан сэтгэл хөдлөм боломжуудтай гайхамшигтай профиль хийгч хэрэгсэл юм. Энэ нь өөрийн боломжуудыг ашиглахын тулд өмнө нь бичигдсэн скриптүүдийг ажиллуулж бас болдог. Хэрэглэгчид нь тусгай зориулалтаар өөрсдийн профиль хийлтийг өөрчлөх боломжтой DTrace D хэлийг ашиглан бүр өөрсдийн хэрэгслүүдийг зохиож болох юм.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- DTrace гэж юу болох ямар ямар боломжуудаар хангадаг талаар.
- Solaris™-ийн DTrace шийдэл болон FreeBSD-д байгаа хоёрын хоорондын ялгаанууд.
- FreeBSD дээр DTrace-ийг хэрхэн идэвхжүүлж ашиглах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- UNIX® болон FreeBSD-ийн үндсүүдийг (Юниксийн үндэс) ойлгох.
- Цөмийн тохиргоо/эмхэтгэлтийн (FreeBSD цөмийг тохируулах нь) үндсийг мэддэг байх.
- Аюулгүй байдлын талаар болон энэ нь FreeBSD-д хэрхэн хамааралтай байх талаар зарим ойлголттой байх (Аюулгүй байдал).
- FreeBSD эхийг хэрхэн авч дахин бүтээх талаар ойлгох (FreeBSD-г шинэчилж сайжруулах нь).

Энэ боломж нь туршилтын гэж тооцогддог. Зарим тохиргоонууд ажиллагааны хувьд дутуу, зарим хэсэг нь ерөөсөө ажиллахгүй байж болох юм. Цагийн явцад энэ боломж нь үйлдвэрлэлд ашиглагдахад бэлэн гэж тооцогдох бөгөөд энэ баримт нь тэр тохиолдолд таарахаар өөрчлөгдөх болно.

## 26.2. Шийдлийн ялгаанууд

FreeBSD дээрх DTrace нь Solaris™-д байдагтай маш төстэй боловч цааш үргэлжлүүлэхээсээ өмнө тайлбарлах ёстой ялгаанууд байдаг. Хэрэглэгчдийн мэдрэх үндсэн ялгаа нь FreeBSD дээр DTrace нь тусгайлан идэвхжүүлэх ёстой байдаг. DTrace нь зөв ажиллахын тулд идэвхжүүлсэн байх ёстой цөмийн тохиргоонууд болон модулиуд байдаг. Эдгээрийг дараа тайлбарлах болно.

CTF өгөгдлийг цөмийн модуль болон цөмөөс өөрөөс нь дуудахад зориулсан дэмжлэгийг

идэвхжүүлэх цөмийн DDB_CTF тохиргоо байдаг. CTF нь Solaris™-ийн Compact C Type буюу Шахмал С Төрөл бөгөөд DWARF болон venerable stab-уудтай (хүндтэй хатгалтууд) адил дибаг мэдээллийн багасгасан хэлбэрийг агуулдаг. Энэ CTF өгөгдөл нь ctfconvert болон ctfmerge гэсэн бүтээх хэрэгслүүдээр хоёртын файлуудад нэмэгддэг. ctfconvert хэрэгсэл нь эмхэтгэгчийн үүсгэсэн DWARFELF дибаг хэсгүүдийг задлан ялгаж ctfmerge нь обьектуудын CTFELF хэсгүүдийг ажилладаг файлууд юм уу эсвэл хуваалцсан сангууд руу нийлүүлдэг. Цөм болон FreeBSD бүтээлтэд зориулж үүнийг хэрхэн идэвхжүүлэх талаар дэлгэрэнгүй тайлбар удахгүй бичигдэх болно.

Solaris™-д байдгаас өөр хангагчид FreeBSD-д байдаг. Хамгийн алдартай нь dtmalloc хангагч бөгөөд FreeBSD цөмд malloc()-ийг төрлөөр нь мөрдөх боломжийг бүрдүүлдэг.

Зөвхөн root DTrace-ийг FreeBSD дээр ашиглаж болно. Энэ нь аюулгүй байдлын ялгаатай холбоотой бөгөөд FreeBSD-д байдаггүй хэдэн доод түвшний аюулгүй байдлын шалгалтууд Solaris™-д байдаг. Тийм болохоор /dev/dtrace/dtrace нь зөвхөн root хэрэглэгчээр хатуу хязгаарлагдсан байдаг.

Төгсгөлд нь DTrace програм хангамж нь Sun™-ий CDDL лицензийн доор байдаг. Common Development and Distribution License буюу Нийтлэг Хөгжүүлэлт ба Түгээлтийн Лиценз нь FreeBSD-тэй цуг ирдэг бөгөөд /usr/src/cddl/contrib/opensolaris/OPENSOLARIS.LICENSE юм уу эсвэл онлайн хэлбэрээр http://www.opensolaris.org/os/licensing хаягаас харж болно.

Энэ лиценз нь DTrace тохиргоотой FreeBSD цөм нь BSD лицензтэй хэвээр гэсэн үг боловч модулиуд нь хоёртын хэлбэрээр түгээгдэх үед юм уу эсвэл хоёртын файлууд дуудагдсан үед CDDL нь хүчин төгөлдөр болдог.

## 26.3. DTrace дэмжлэгийг идэвхжүүлэх

DTrace-д зориулсан дэмжлэгийг идэвхжүүлэхийн тулд дараах мөрүүдийг цөмийн тохиргооны файлд нэмнэ:

```
options         KDTRACE_HOOKS
options         DDB_CTF
```

AMD64 архитектурын хэрэглэгчид өөрсдийн цөмийн тохиргооны файлдаа дараах мөрийг нэмэхийг хүсэх болно:

```
options         KDTRACE_FRAME
```

Энэ тохиргоо нь FBT боломжид зориулсан дэмжлэгийг хангадаг. DTrace нь энэ тохиргоогүйгээр ажилладаг боловч функцийн хязгаарын мөрдөлтөд зориулсан дэмжлэг хязгаарлагдмал байх болно.

Бүх эх CTF тохиргоонуудтай дахин бүтээгдэж суулгагдсан байх ёстой. Ингэхийн тул FreeBSD-ийн эхийг доорх тушаалуудыг ашиглан дахин бүтээнэ:

```
# cd /usr/src

# make WITH_CTF=1 kernel
```

Системийг дахин ачаалах хэрэгтэй.

Дахин ачаалж шинэ цөмийг санах ойд дуудагдсаны дараа Korn бүрхүүлийн дэмжлэгийг нэмэх хэрэгтэй. DTrace хэрэгслүүд нь `ksh` дээр бичигдсэн хэд хэдэн хэрэгслүүдтэй болохоор энэ нь хэрэгтэй юм. shells/ksh93-г суулгах хэрэгтэй. Эдгээр хэрэгслүүдийг shells/pdksh эсвэл shells/mksh дээр ажиллуулж бас болно.

Төгсгөлд нь одоогийн DTrace хэрэгслүүдийг олж авах хэрэгтэй. Хэрэв FreeBSD 10 бол DTrace хэрэгслүүдийг /usr/shared/dtrace байрлалаас олж болно. Үгүй бол DTrace хэрэгслүүдийг sysutils/dtrace-toolkit портоос суулгаж болно.

## 26.4. DTrace-г ашиглах

DTrace боломжийг ашиглахаас өмнө DTrace төхөөрөмж байх хэрэгтэй. Төхөөрөмжийг дуудахын тулд дараах тушаалыг ажиллуулна:

```
# kldload dtraceall
```

DTrace дэмжлэг одоо боломжтой болсон байх ёстой. Бүх шалгалтыг харахын тулд администратор дараах тушаалыг одоо ажиллуулж болно:

```
# dtrace -l | more
```

Дэлгэцийн буфферийг түргэн дүүргэх болохоор бүх гаралтыг `more` хэрэгсэл рүү дамжуулдаг. Энд хүрэхэд DTrace-ийг ажиллаж байна гэж үзнэ. Одоо уг хэрэгслийг шалгаж үзэх цаг боллоо.

Уг хэрэгсэл нь системийн мэдээллийг цуглуулахын тулд DTrace-тэй цуг ажиллуулж болох ашиглахад бэлэн скриптүүдийн цуглуулга юм. Нээлттэй файлууд, санах ой,CPU-ийн хэрэглээ болон өөр олон зүйлсийг шалгах скриптүүд байдаг. Скриптүүдийг дараах тушаалаар задалж болно:

```
# gunzip -c DTraceToolkit* | tar xvf -
```

`cd` ашиглан тэр сан руу өөрчилж жижиг үсгээр бичигдсэн нэртэй бүх файлууд дээр ажиллуулах зөвшөөрлүүдийг `755` руу өөрчлөх хэрэгтэй.

Эдгээр бүх скриптүүдийн агуулгад өөрчлөлт хийх хэрэгтэй болно. /usr/bin/ksh гэснийг /usr/local/bin/ksh болгох ёстой бөгөөд /usr/bin/sh-ийг /bin/sh болгон ашиглахаар, төгсгөлд нь /usr/bin/perl-г /usr/local/bin/perl болгон ашиглахаар болгож өөрчлөх хэрэгтэй.

FreeBSD дээрх DTrace дэмжлэг нь *бүрэн бус* бөгөөд *туршилтын* гэдгийг уншигчид сануулах нь ухаалаг болох юм. Эдгээр скриптүүдийн олонхи нь ажиллахгүй бөгөөд тэдгээр нь хэтэрхий Solaris™-д тусгайлан зориулсан эсвэл одоо дэмжигдээгүй байгаа шалгалтуудыг ашигладаг.

Үүнийг бичиж байх үед DTrace хэрэгслийн зөвхөн хоёр скрипт FreeBSD дээр бүрэн дэмжигдсэн: эдгээр нь hotkernel болон procsystime скриптүүд юм. Эдгээр хоёрыг бид энэ бүлгийн дараагийн хэсгүүдэд авч үзэх болно.

hotkernel нь аль функц цөмийн ихэнх хугацааг ашиглаж байгааг илрүүлэхээр хийгдсэн. Хэвийнээр ажиллуулахад энэ нь дараахтай төстэй гаралтыг харуулах болно:

```
# cd /usr/shared/dtrace/toolkit
# ./hotkernel
Sampling... Hit Ctrl-C to end.
```

Системийн администратор процессийг зогсоохын тулд `Ctrl` + `C` товчлуурын хослолыг ашиглах ёстой. Зогсоосны дараа скрипт нь цөмийн функцуудын жагсаалт болон хугацааны мэдээллийг харуулах бөгөөд гаралтыг хугацааны ихэссэн дарааллаар эрэмбэлж харуулна:

```
kernel`_thread_lock_flags                             2   0.0%
0xc1097063                                            2   0.0%
kernel`sched_userret                                  2   0.0%
kernel`kern_select                                    2   0.0%
kernel`generic_copyin                                 3   0.0%
kernel`_mtx_assert                                    3   0.0%
kernel`vm_fault                                       3   0.0%
kernel`sopoll_generic                                 3   0.0%
kernel`fixup_filename                                 4   0.0%
kernel`_isitmyx                                       4   0.0%
kernel`find_instance                                  4   0.0%
kernel`_mtx_unlock_flags                              5   0.0%
kernel`syscall                                        5   0.0%
kernel`DELAY                                          5   0.0%
0xc108a253                                            6   0.0%
kernel`witness_lock                                   7   0.0%
kernel`read_aux_data_no_wait                          7   0.0%
kernel`Xint0x80_syscall                               7   0.0%
kernel`witness_checkorder                             7   0.0%
kernel`sse2_pagezero                                  8   0.0%
kernel`strncmp                                        9   0.0%
kernel`spinlock_exit                                 10   0.0%
kernel`_mtx_lock_flags                               11   0.0%
kernel`witness_unlock                                15   0.0%
kernel`sched_idletd                                 137   0.3%
0xc10981a5                                        42139  99.3%
```

Энэ скрипт нь цөмийн модулиудтай бас ажилладаг. Энэ боломжийг ашиглахын тулд

скриптийг `-m` өгөгдөлтэй ажиллуулах хэрэгтэй:

```
# ./hotkernel -m
Sampling... Hit Ctrl-C to end.
^C
MODULE                                                  COUNT   PCNT
0xc107882e                                                  1   0.0%
0xc10e6aa4                                                  1   0.0%
0xc1076983                                                  1   0.0%
0xc109708a                                                  1   0.0%
0xc1075a5d                                                  1   0.0%
0xc1077325                                                  1   0.0%
0xc108a245                                                  1   0.0%
0xc107730d                                                  1   0.0%
0xc1097063                                                  2   0.0%
0xc108a253                                                 73   0.0%
kernel                                                    874   0.4%
0xc10981a5                                             213781  99.6%
```

procsystime скрипт нь өгөгдсөн PID эсвэл процессийн хувьд системийн дуудлагын цагийн хэрэглээг авч хэвлэн харуулдаг. Дараах жишээн дээр /bin/csh-ийн шинэ хуулбар ажиллуулсан. procsystime-ийг ажиллуулсан бөгөөд өөр нэг `csh` дээр цөөн хэдэн тушаалыг бичтэл энэ нь хүлээх горимд орсон. Энэ тестийн үр дүн энд байна:

```
# ./procsystime -n csh
Tracing... Hit Ctrl-C to end...
^C

Elapsed Times for processes csh,

         SYSCALL          TIME (ns)
          getpid               6131
       sigreturn               8121
           close              19127
           fcntl              19959
             dup              26955
         setpgid              28070
            stat              31899
       setitimer              40938
           wait4              62717
       sigaction              67372
     sigprocmask             119091
    gettimeofday             183710
           write             263242
          execve             492547
           ioctl             770073
           vfork            3258923
      sigsuspend            6985124
```

```
            read          3988049784
```

Энд үзүүлснээр `read()` системийн дуудлага ихэнх цагийг, `getpid()` системийн дуудлага хамгийн бага хугацааг ашигласныг наносекундаар харуулж байна.

## 26.5. D хэл

DTrace хэрэгсэл нь DTrace-ийн тусгай хэл дээр байдаг олон скриптүүдийг агуулдаг. Энэ хэл нь Sun™-ийн баримтаар "D хэл" гэгддэг бөгөөд C++ хэлтэй тун төстэй юм. Энэ хэлний талаарх гүнзгий хэлэлцүүлэг нь энэ баримтаас хүрээнээс гадна юм. Энэ нь http://wikis.oracle.com/display/DTrace/Documentation хаяг дээр дэлгэрэнгүй хэлэлцэгдсэн байдаг.

# Part IV: Сүлжээний Холболт

FreeBSD нь өндөр ажиллагаатай сүлжээний серверүүдэд зориулагдсан хамгийн өргөн хэрэглэгддэг үйлдлийн системүүдийн нэг юм. Энэ хэсэгт дараах сэдвүүдийг хамрах болно:

- Сериал холболт
- PPP болон Ethernet дээгүүрх PPP (PPP over Ethernet)
- Цахим захидал
- Сүлжээний серверүүдийг ажиллуулах
- Галт хананууд
- Бусад дээд түвшний сүлжээний сэдвүүд

Эдгээр бүлгүүд нь танд мэдээлэл хэрэгтэй үед уншигдахаар зориулагдсан юм. Та ямар нэгэн тодорхой дарааллаар унших хэрэггүй бөгөөд FreeBSD ашиглаж эхлэхээсээ өмнө бүгдийг нь унших шаардлагагүй болно.

# Chapter 27. Цуваа холбоонууд

## 27.1. Ерөнхий агуулга

UNIX® нь цуваа холбоонуудад зориулсан дэмжлэгтэй үргэлж байсаар ирсэн. Тухайлбал хамгийн анхны UNIX® машинууд хэрэглэгчийн оролт болон гаралтын хувьд цуваа шугамууд дээр тулгуурладаг байсан. Дундаж терминал нь секундэд 10 тэмдэгт дамжих цуваа хэвлэгч болон гараас тогтдог байсан тэр үеэс хойш юмс асар их өөрчлөгдсөөр ирсэн юм. Энэ бүлэг нь FreeBSD дээр цуваа холбоонуудыг ашиглах зарим аргуудаас дурдах болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- FreeBSD системд терминалуудыг хэрхэн холбох талаар.
- Алсын хостууд уруу хэрхэн модемоор залгаж холбогдох талаар.
- Алсын хэрэглэгчдийг модемоор FreeBSD систем уруу нэвтрэхийг хэрхэн зөвшөөрөх талаар.
- Цуваа консолоос FreeBSD системийг хэрхэн ачаалах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- Шинэ цөмийг хэрхэн тохируулан суулгах талаар мэдэх.
- UNIX® зөвшөөрлүүд болон процессуудыг ойлгох.
- FreeBSD дээр ашиглагддаг цуваа тоног төхөөрөмжийн техникийн гарын авлагад хандаж болохоор байх.

## 27.2. Танилцуулга

### 27.2.1. Ухагдахуунууд

**bps**

Bits per Second bps буюу секундэд дамжуулах бит - өгөгдөл дамжих хурд

**DTE**

Data Terminal Equipment DTE буюу Өгөгдлийн Терминал Төхөөрөмж - жишээ нь таны компьютер

**DCE**

Data Communications Equipment DCE буюу Өгөгдлийн Холбоонуудын Төхөөрөмж - таны модем

**RS-232**

Тоног төхөөрөмжийн цуваа холбоонуудад зориулсан андагч стандарт. Одоо TIA-232 гэгддэг.

Холбоонуудын өгөгдлийн хурдны талаар ярихдаа энэ хэсэг "baud" гэсэн ухагдахууныг

ашигладаггүй. Baud нь тодорхой хугацаанд хийгдсэн цахилгааны төлвийн шилжилтүүдийн тоог хэлдэг бол bps нь хэрэглэгдэх *зөв* ойлголт юм.

## 27.2.2. Кабелиуд болон Портууд

FreeBSD систем уруу модем эсвэл терминалыг холбохын тулд танд компьютер дээр чинь цуваа порт болон өөрийн цуваа төхөөрөмж уруу холбох зөв кабель хэрэгтэй болно. Хэрэв та өөрийн тоног төхөөрөмжтэй болон шаардагдах кабельтай аль хэдийн танил болсон бол энэ хэсгийг айлгүйгээр орхиж болно.

### 27.2.2.1. Кабелиуд

Хэд хэдэн өөр өөр төрлийн цуваа кабелиуд байдаг. Бидний зорилгод нийцэх хоёр хамгийн нийтлэг төрөл бол null-модем болон стандарт RS-232 кабелиуд юм. Таны тоног төхөөрөмжийн баримт шаардлагатай кабелийн төрлийг тайлбарлах ёстой.

#### 27.2.2.1.1. Null-модем кабелиуд

Null-модем кабель нь "Signal Ground" буюу Газар дохио зэрэг зарим дохионуудыг шууд нэвтрүүлдэг боловч зарим дохионуудыг шилжүүлдэг. Жишээ нь нэг талын "Transmitted Data" буюу Дамжигдсан Өгөгдөл зүү нь нөгөө талын "Received Data" буюу Хүлээн авсан Өгөгдөл зүү уруу ордог.

null-модем кабелийг терминалуудтай ажиллахаар болгож болно. Энэ хүснэгт нь RS-232C дохионуудын нэрс болон DB-25 холбогч дээрх зүүний тоонуудыг харуулж байна. Мөн стандарт нь шууд холбогдох 1-р зүүг 1-р зүү *Protective Ground* буюу Хамгаалалтын Газар шугам уруу дууддаг боловч энэ нь ихэвчлэн орхигддог. Зарим терминалууд 2, 3 болон 7-р зүүнүүдийг зөвхөн ашиглахад зүгээр ажилладаг бол бусдууд нь доор дурдсан жишээнүүдээс өөр тохиргоонуудыг шаарддаг.

*Хүснэгт 9. DB-25-аас DB-25 Null-Модем кабель*

| **Дохио** | **Зүү #** | | **Зүү #** | **Дохио** |
|---|---|---|---|---|
| SG | 7 | холбогдох | 7 | SG |
| TD | 2 | холбогдох | 3 | RD |
| RD | 3 | холбогдох | 2 | TD |
| RTS | 4 | холбогдох | 5 | CTS |
| CTS | 5 | холбогдох | 4 | RTS |
| DTR | 20 | холбогдох | 6 | DSR |
| DTR | 20 | холбогдох | 8 | DCD |
| DSR | 6 | холбогдох | 20 | DTR |
| DCD | 8 | холбогдох | 20 | DTR |

Дараагийн хоёр хүснэгт нийтлэг байдаг өөр хоёр схемийг үзүүлэв.

*Хүснэгт 10. DB-9-өөс DB-9 Null-Модем кабель*

| Дохио | Зүү # | | Зүү # | Дохио |
|---|---|---|---|---|
| RD | 2 | холбогдох | 3 | TD |
| TD | 3 | холбогдох | 2 | RD |
| DTR | 4 | холбогдох | 6 | DSR |
| DTR | 4 | холбогдох | 1 | DCD |
| SG | 5 | холбогдох | 5 | SG |
| DSR | 6 | холбогдох | 4 | DTR |
| DCD | 1 | холбогдох | 4 | DTR |
| RTS | 7 | холбогдох | 8 | CTS |
| CTS | 8 | холбогдох | 7 | RTS |

*Хүснэгт 11. DB-9-өөс DB-25 Null-Модем кабель*

| Дохио | Зүү # | | Зүү # | Дохио |
|---|---|---|---|---|
| RD | 2 | холбогдох | 2 | TD |
| TD | 3 | холбогдох | 3 | RD |
| DTR | 4 | холбогдох | 6 | DSR |
| DTR | 4 | холбогдох | 8 | DCD |
| SG | 5 | холбогдох | 7 | SG |
| DSR | 6 | холбогдох | 20 | DTR |
| DCD | 1 | холбогдох | 20 | DTR |
| RTS | 7 | холбогдох | 5 | CTS |
| CTS | 8 | холбогдох | 4 | RTS |

Нэг талд байгаа нэг зүү нь нөгөө талын хос зүү уруу холбогдох бол тэдгээрийн холбогч дээр хос зүүний хооронд нэг богино утсаар болон урт утсаар нөгөө ганц зүү уруу холбон ихэвчлэн шийдсэн байдаг.

Дээрх дизайнууд нь хамгийн нийтлэг юм шиг санагддаг. Өөр хувилбарууд дээр SG нь SG уруу, TD нь RD уруу, RTS болон CTS нь DCD уруу, DTR нь DSR уруу болон эсрэгээр холбогддог.

##### 27.2.2.1.2. Стандарт RS-232C кабелиуд

Стандарт цуваа кабель нь RS-232C дохионуудыг бүгдийг нь шууд нэвтрүүлдэг. Өөрөөр хэлбэл кабелийн нэг талын "Transmitted Data" буюу Дамжигдсан Өгөгдөл зүү нь нөгөө талын "Transmitted Data" буюу Дамжигдсан Өгөгдөл зүү уруу ордог. Энэ нь модемийг таны FreeBSD систем уруу болон зарим нэг тохирох терминалиуд уруу холбоход хэрэглэгдэх кабелийн төрөл юм.

#### 27.2.2.2. Портууд

Цуваа портууд нь өгөгдлийг FreeBSD хост компьютер болон терминалын хооронд

дамжуулах төхөөрөмжүүд юм. Энэ хэсэг нь ямар төрлийн портууд байдаг болон FreeBSD дээр тэдгээрт хэрхэн ханддагийг тайлбарлах болно.

##### 27.2.2.2.1. Портуудын төрлүүд

Хэд хэдэн төрлийн портууд байдаг. Та кабелийг худалдаж авах юм уу эсвэл хийхээсээ өмнө энэ нь таны терминал болон FreeBSD систем дээрх портуудад таарах эсэхийг шалгах хэрэгтэй.

Ихэнх терминалууд DB-25 портуудтай байдаг. Персонал компьютерууд нь DB-25 эсвэл DB-9 портуудтай байдаг. PC дээрээ олон порттой цуваа карттай бол RJ-12 эсвэл RJ-45 портууд байж болох юм.

Тоног төхөөрөмжтэй цуг ирсэн баримтаас ашиглах портын төрлийн тодорхойлолтыг үзнэ үү. Портыг нүдээрээ харж шалгах нь бас үр дүнгээ өгдөг.

##### 27.2.2.2.2. Портын нэрс

FreeBSD дээр цуваа порт бүрт /dev сан дахь оруулгаар та ханддаг. Хоёр өөр төрлийн оруулга байдаг:

- Call-in буюу дуудагдах портууд нь /dev/ttyuN гэгддэг бөгөөд *N* нь тэгээс эхлэх портын дугаар юм. Ерөнхийдөө дуудагдах портыг терминалуудад зориулж ашигладаг. Data carrier detect (DCD) буюу өгөгдлийн зөөгч илрүүлэх дохиог зөв ажиллахыг цуваа шугам батлахыг дуудагдах портууд нь шаарддаг.
- Call-out буюу дуудах портууд нь /dev/cuauN гэгддэг. Дуудах портуудыг терминалуудад зориулж ихэвчлэн ашигладаггүй, зөвхөн модемуудад зориулж ашигладаг. Цуваа кабель эсвэл терминал нь зөөгчийг илрүүлэх дохиог дэмждэггүй бол дуудах портуудыг ашиглаж болох юм.

Хэрэв терминалыг эхний цуваа порт уруу (COM1) холбосон бол терминалд хандахдаа /dev/ttyu0 гэж ашиглах болно. Хэрэв терминал нь хоёр дахь цуваа порт дээр байгаа бол (COM2) /dev/ttyu1 гэх мэтээр ашиглах ёстой.

### 27.2.3. Цөмийн тохиргоо

FreeBSD нь анхдагчаар дөрвөн цуваа портыг дэмждэг. MS-DOS®-ийн ертөнцөд эдгээр нь COM1, COM2, COM3, болон COM4 гэгддэг. FreeBSD нь одоогоор BocaBoard 1008 болон 2016 зэрэг "дүлий" олон порттой цуваа интерфэйс картууд болон Digiboard болон Stallion Technologies-ийн хийдэг илүү ухаалаг олон порттой картуудыг дэмждэг. Гэхдээ анхдагч цөм нь зөвхөн стандарт COM портуудыг хайдаг.

Таны цуваа портуудыг таны цөм таньж байгаа эсэхийг харахын тулд цөмийг ачаалж байх үед мэдэгдлүүдийг үзэх хэрэгтэй эсвэл цөмийн ачаалалтын мэдэгдлүүдийг дахин харуулах `/sbin/dmesg` тушаалыг ашиглах хэрэгтэй. `uart` тэмдэгтүүдээр эхэлсэн мэдэгдлүүдийг хайгаарай:

```
# /sbin/dmesg | grep 'uart'
```

Хэрэв цөм бүх цуваа портуудыг чинь танихгүй байгаа бол /boot/device.hints файлд тохируулах хэрэгтэй байж болох юм. Байхгүй төхөөрөмжүүдийн хувьд тэдгээр мөрүүдийг тайлбар болгох эсвэл бүрмөсөн арилгаж болох юм.

`IO_COM1 порт` нь `0x3f8` хаяг дээрх портод зориулагдсан орлуулалт юм, `IO_COM2` нь `0x2f8`-ийн `IO_COM3` нь `0x3e8`-ийн `IO_COM4` нь `0x2e8`-ийн орлуулалт юм. Эдгээр хаягууд нь цуваа портуудын нэлээн түгээмэл портын хаягууд бөгөөд 4,3, 5 болон 9 тасалдлууд нь нэлээн түгээмэл тасалдал хүсэх шугамнууд юм. Ердийн цуваа портууд нь ISA-bus бүхий PC-үүд дээр тасалдлуудыг хуваалцаж *чаддаггүй* бас санаарай (олон порттой хавтангууд нь өөр дээрээ хавтан дээр бүх 16550A-уудад нэг эсвэл хоёр тасалдал хүсэх шугамуудыг хуваалцахыг зөвшөөрдөг бичил схемүүдтэй байдаг).

### 27.2.4. Төхөөрөмжийн Тусгай файлууд

Цөм дэх ихэнх төхөөрөмжүүдэд "төхөөрөмжийн тусгай файлууд"ын тусламжтайгаар ханддаг бөгөөд эдгээр файлууд нь /dev санд байдаг. sio төхөөрөмжид /dev/ttyuN (dial-in буюу гаднаас залгах) болон /dev/cuauN (call-out буюу дуудах) төхөөрөмжүүдийн тусламжтай ханддаг. FreeBSD нь бас эхлүүлэх төхөөрөмжүүд (/dev/ttyuN.init болон /dev/cuauN.init) болон түгжих төхөөрөмжүүдтэй (/dev/ttyuN.lock болон /dev/cuauN.lock) байдаг. Эхлүүлэх төхөөрөмжүүд нь урсгалын хяналтдаа `RTS/CTS` дохиоллыг ашигладаг модемийн `crtscts` зэрэг холбооны портын параметрүүдийг порт нээгдэх бүрт эхлүүлж тохируулахад хэрэглэгддэг. Түгжих төхөөрөмжүүд нь хэрэглэгчид болон програмуудын зарим нэг параметрүүдийг өөрчлөхөөс сэргийлэхийн тулд портууд дээр тугуудыг түгжихэд хэрэглэгддэг. Терминалын тохиргоонууд, түгжих болон эхлүүлэх төхөөрөмжүүд болон терминалын тохируулгуудыг тохируулах талаар дэлгэрэнгүй мэдээллийг termios(4), sio(4), болон stty(1) гарын авлагын хуудаснуудаас үзнэ үү.

### 27.2.5. Цуваа портын тохиргоо

ttyuN (эсвэл cuauN) төхөөрөмж нь таны өөрийн програмууддаа зориулж нээхийг хүсэх ердийн төхөөрөмж юм. Процесс төхөөрөмжийг онгойлгоход энэ нь терминалын I/O тохиргоонуудын анхдагч олонлогтой байх болно. Та эдгээр тохиргоонуудыг дараах тушаалаар үзэж болно

```
# stty -a -f /dev/ttyu1
```

Энэ төхөөрөмжид тохиргоонуудыг өөрчлөхөд төхөөрөмжийг хаах хүртэл тохиргоонууд нь идэвхтэй байдаг. Дахин онгойлгоход энэ нь анхдагч тохиргоо уруугаа буцдаг. Анхдагч олонлогт өөрчлөлтүүдийг хийхийн тулд та онгойлгоод "эхний төлөвт" төхөөрөмжийн тохиргоонуудыг тааруулж болно. Жишээ нь ttyu5-ийн хувьд `CLOCAL` горим, 8 бит холбоо, болон `XON/XOFF` урсгалын хяналтыг идэвхжүүлэхийн тулд доор дурдсаныг бичнэ:

```
# stty -f /dev/ttyu5.init clocal cs8 ixon ixoff
```

Цуваа төхөөрөмжүүдийн системийн дагуух эхлүүлэлт нь /etc/rc.d/serial файлаар хянагддаг.

Энэ файл нь цуваа төхөрөмжүүдийн анхдагч тохиргоонуудад нөлөөлдөг.

Програм зарим тохиргоонуудыг өөрчлөхөөс сэргийлэхийн тулд "түгжих төлөвт" төхөөрөмжид тохиргоо хийнэ. Жишээ нь ttyu5-ийн хурдыг 57600 bps болгож түгжихийн тулд доор дурдсаныг бичнэ:

```
# stty -f /dev/ttyu5.lock 57600
```

Одоо програм ttyu5-ийг онгойлгож портын хурдыг өөрчлөхийг оролдоход хурд нь 57600 bps-с хөдлөхгүй байх болно.

Эхний төлөвт болон түгжих төлөвт төхөөрөмжүүдийг зөвхөн `root` бүртгэл бичдэгээр болгох хэрэгтэй.

# 27.3. Терминалууд

Терминалууд нь таныг компьютерийнхаа консол дээр байхгүй үед эсвэл сүлжээнд холбогдоогүй байхад амархан, үнэ хямд аргаар FreeBSD систем уруу хандах боломжийг олгодог. Энэ хэсэг нь терминалуудыг FreeBSD-тэй хэрхэн ашиглах талаар тайлбарлана.

## 27.3.1. Терминалуудын хэрэглээ болон төрлүүд

Анхдагч UNIX® системүүдэд консол байгаагүй юм. Харин компьютерийн цуваа портуудад холбогдсон терминалуудаар хүмүүс нэвтрэн ороод програмууд ажиллуулдаг байсан байна.

Цуваа порт дээр нэвтрэх сесс үүсгэх чадвар өнөөгийн бараг бүх UNIX® загварын үйлдлийн системд байдаг бөгөөд FreeBSD ч бас үүний нэгэн адил юм. Ашиглагдаагүй байгаа цуваа портод холбогдсон терминалыг ашиглан та нэвтрэн орж дурын текст програмыг консол эсвэл `xterm` цонхонд ажиллуулдаг байсан шигээр ажиллуулж болох юм.

FreeBSD системд олон терминалуудыг холбож болох юм. Хуучин компьютер нь FreeBSD ажиллуулж байгаа илүү хүчтэй компьютерт холбогдсон терминал болж болох юм. Ингэснээр ганц хэрэглэгчийн байсан компьютерийг хүчирхэг олон хэрэглэгчийн систем болгон хувиргаж болох юм.

Энэ хэсэгт FreeBSD-н дэмждэг дүлий терминал, терминал хэлбэрээр ажиллаж байгаа компьютер, болон X терминал гэсэн гурван төрлийн терминалын талаар тайлбарлана.

### 27.3.1.1. Дүлий терминалууд

Дүлий терминалууд нь таныг компьютерууд уруу цуваа шугамуудаар холбогдохыг зөвшөөрөх тусгайлсан тоног төхөөрөмж юм. Тэдгээрийг "дүлий" гэдэг бөгөөд тэдгээр нь зөвхөн текст харуулах, илгээх, болон хүлээж авах хангалттай тооцооллын хүчин чадалтай учраас тэр юм. Та тэдгээр дээр ямар ч програм ажиллуулж чадахгүй. Текст засварлагчууд, эмхэтгэгчид, цахим захидал, тоглоомууд гэх зэргүүдийг ажиллуулах хүчин чадал бүхий таны компьютерт тэдгээрийг холбодог.

Бараг бүх төрөл FreeBSD-тэй цуг ажиллана. Зарим нэг өндөр чанартай терминалууд график

үзүүлж чаддаг боловч эдгээр дэвшилтэт боломжуудыг зөвхөн цөөн хэдэн програм хангамжийн багцууд ашиглаж чаддаг.

Ажилтнууд нь график програмуудад хандах шаардлагагүй ажлын орчинд дүлий терминалууд нь түгээмэл байдаг.

#### 27.3.1.2. Терминал маягаар ажиллаж байгаа PC-үүд

Хэрэв дүлий терминал нь текст харуулах, илгээж хүлээн авах чадвартай байдаг бол ямар ч нөөц персонал компьютер дүлий терминал болж чадна. Танд зөв кабель болон компьютер дээр ажиллах зарим нэг *терминал эмуляц хийх* програм хангамж хоёр л хэрэгтэй болох юм.

Ийм тохиргоо их ашигтай байдаг. Жишээ нь хэрэв нэг хэрэглэгч FreeBSD системийн консол дээр завгүй ажиллаж байвал өөр хэрэглэгч FreeBSD систем уруу терминал маягаар холбосон арай хүчин чадал багатай персонал компьютер дээрээс зарим нэг текст орчны ажил зэрэг хийж байж болох юм.

FreeBSD-ийн үндсэн системд цуваа холболтоор ажиллахад хэрэглэж болох хамгийн багаар бодоход хоёр хэрэгсэл байдаг: энэ нь cu(1) болон tip(1) юм.

FreeBSD ажиллаж байгаа клиент систем дээрээс өөр системийн цуваа холболт уруу холбогдохын тулд та дараах тушаалыг ашиглаж болно:

```
# cu -l serial-port-device
```

Энд байгаа "serial-port-device" нь таны системийн цуваа портод зориулагдсан тусгай төхөөрөмжийн файлын нэр юм. Эдгээр төхөөрөмжийн файлууд нь /dev/cuauN гэж нэрлэгддэг.

Төхөөрөмжийн "N"-хэсэг нь цуваа портын дугаарыг илэрхийлнэ.

FreeBSD дээр төхөөрөмжийн дугаарууд нь нэгээс биш тэгээс эхэлдэг Энэ нь COM1 гэж нэрлэдэг порт FreeBSD дээр ихэвчлэн /dev/cuau0 гэгддэг.

Зарим хүмүүс портын цуглуулгад байдаг өөр програмууд ашиглахыг илүүд үздэг. comms/minicom зэрэг цөөнгүй хэдэн хэрэгслүүд байдаг.

#### 27.3.1.3. X терминалууд

X терминалууд нь хамгийн төвөгтэй терминалуудын нэг юм. Цуваа порт уруу холбогдохын оронд тэдгээр нь ихэвчлэн Ethernet зэрэг сүлжээнд холбогддог. Зөвхөн текст орчны програмуудад зориулагдсан биш тэдгээр нь харин X програм үзүүлж чаддаг.

Энэ бүлэг X терминалуудыг суулгах, тохируулах болон тэдгээрийн хэрэглээг *тайлбарлахгүй*.

### 27.3.2. Тохиргоо

Энэ хэсэг нь терминал дээр нэвтрэх сесс идэвхжүүлэхийн тулд өөрийн FreeBSD систем дээр

таныг юу тохируулах хэрэгтэйг тайлбарлах болно. Энэ нь цуваа порт дэмжихээр өөрийн цөмийг таныг аль хэдийн тохируулсан байгаа гэж үздэг. Цуваа порт уруу терминал холбогдсон бөгөөд таныг тэр терминал уруу холбогдсон байгаа гэж тооцдог.

`init` процесс нь бүх процессийн хяналт болон системийн эхлүүлэх үе дэх эхний тохиргоог хариуцдаг. `init`-ийн хийдэг үйлдлүүдийн нэг нь /etc/ttys файлыг уншиж байгаа терминалууд дээр `getty` процессийг эхлүүлэх явдал юм. `getty` процесс нь нэвтрэх нэрийг уншиж `login` програмыг ажиллуулахыг хариуцдаг.

FreeBSD системийн хувьд терминалуудыг тохируулахын тулд `root` хэрэглэгчээр дараах алхмуудыг гүйцэтгэх хэрэгтэй:

1. /dev сан дахь оруулгад зориулж /etc/ttys файлд хэрэв байхгүй бол цуваа портын мөрийг нэмнэ.
2. `/usr/libexec/getty` нь порт дээр ажиллахыг зааж /etc/gettytab файлаас тохирох *getty* төрлийг тохируулах хэрэгтэй.
3. Анхдагч терминалын төрлийг заана.
4. Портыг "on" болгоно.
5. Порт "secure" байх ёстой эсэхийг заана.
6. `init` тушаалаар /etc/ttys файлыг дахин уншуулна.

Нэмэлт алхам болгож хоёрдугаар алхам дээр та өөр *getty* төрлийг хэрэглээнд зориулан /etc/gettytab-д оруулан үүсгэж болох юм. Дэлгэрэнгүй мэдээллийг gettytab(5) болон getty(8) гарын авлагын хуудаснуудаас үзэхийг зөвлөж байна.

#### 27.3.2.1. /etc/ttys уруу оруулгууд нэмэх нь

/etc/ttys файл нь нэвтрэлтүүдийг зөвшөөрөхийг хүссэн FreeBSD систем дээрх бүх портуудыг жагсаадаг. Жишээ нь эхний виртуал консол ttyv0 нь энэ файлд өөрийн оруулгатай байна. Та энэ оруулгыг ашиглан консол дээр нэвтрэн орж чадна. Энэ файл нь бас бусад виртуал консолууд, цуваа портууд, болон псевдо tty-уудын оруулгуудыг бас агуулдаг. Хатуугаар холбогдсон терминалын хувьд цуваа портын /dev оруулгыг /dev хэсэггүйгээр (жишээ нь /dev/ttyv0 нь ttyv0 гэж жагсаагдах болно) жагсаадаг.

Анхдагч FreeBSD суулгац нь /etc/ttys файлыг эхний дөрвөн цуваа портын дэмжлэгтэйгээр агуулдаг: ttyu0-с ttyu3 хооронд байна. Терминалыг эдгээр портуудын аль нэгэнд холбож байгаа бол өөр оруулга нэмэх шаардлагагүй юм.

*Жишээ 32. /etc/ttys уруу терминалын оруулгууд нэмэх нь*

Энэ жишээ хоёр терминалыг тохируулна: Wyse-50 болон VT-100 терминалыг эмуляц хийх Procomm терминал програм хангамжийг ажиллуулж байгаа хуучин 286 IBM PC-г терминал маягаар ашиглая. Wyse-ийг хоёр дахь цуваа порт уруу холбож 286-г зургаа дахь цуваа порт уруу холбоё (олон порттой цуваа картын порт). /etc/ttys файл дахь харгалзах оруулгууд нь иймэрхүү байна:

```
ttyu1  "/usr/libexec/getty std.38400"  wy50  on  insecure
ttyu5   "/usr/libexec/getty std.19200"  vt100  on  insecure
```

- Эхний талбар нь ихэвчлэн /dev-д байгаа терминалын тусгай файлын нэрийг заадаг.
- Хоёр дахь талбар нь энэ мөрийн хувьд ажиллах тушаал бөгөөд ихэвчлэн getty(8) байдаг. `getty` нь эхлүүлж шугамыг нээн хурдыг тохируулан хэрэглэгчийн нэрийг асууж login(1) програмыг ажиллуулдаг.`getty` програм нь нэг (нэмэлт) параметр *getty* төрлийг өөрийн тушаалын мөрөнд хүлээн авдаг. *getty* төрөл нь терминалын шугам дахь bps хурд болон parity зэрэг шинжүүдийг тохируулдаг. `getty` програм нь эдгээр шинжүүдийг /etc/gettytab файлаас уншдаг./etc/gettytab файл нь хуучин болон шинэ терминалын шугамнуудад зориулсан олон мөрүүдийг агуулдаг. Бараг бүх тохиолдолд `std` текстээр эхэлсэн оруулгууд нь hardwired буюу хатуугаар холбогдсон терминалуудын хувьд ажилладаг. Эдгээр оруулгууд нь parity-г орхидог. 110-аас 115200 хүртэл bps хурд болгоны хувьд `std` оруулга байдаг. Мэдээж та өөрийнхөө оруулгыг энэ файлд нэмж болно. gettytab(5) гарын авлагын хуудас илүү дэлгэрэнгүй мэдээллийг өгөх болно.*getty* төрлийг /etc/ttys файлд тохируулахдаа терминал дахь холбооны тохируулгууд таарч байгаа эсэхийг шалгаарай.Бидний жишээн дээр Wyse-50 нь parity ашиглахгүй бөгөөд 38400 bps хурдаар холбогдоно. 286 PC нь parity ашиглахгүй бөгөөд 19200 bps хурдаар холбогдоно.
- Гурав дахь талбар нь тэр tty шугам уруу ихэвчлэн холбогдсон байдаг терминалын төрөл юм. Хэрэглэгч нь бараг л дурын терминалын төрөл эсвэл програм хангамжаар утасдан холбогдож болох учир dial-up портуудын хувьд `unknown` эсвэл `dialup` төрөл энэ талбарт ихэвчлэн ашиглагддаг. Хатуугаар холбогдсон терминалуудын хувьд терминалын төрөл өөрчлөгдөхгүй учраас та энэ талбарт termcap(5) мэдээллийн баазаас жинхэнэ терминалын төрлийг тавьж болно.Бидний жишээн дээр Wyse-50 нь жинхэнэ терминалын төрлийг ашиглах бол Procomm ажиллуулж байгаа 286 PC нь VT-100-г эмуляц хийхээр тохируулагдах болно.
- Дөрөв дэх талбар нь порт идэвхтэй байх ёстой эсэхийг заадаг. Энд `on` гэж тавихад `init` процесс хоёр дахь талбар дахь програм `getty`-г ажиллуулах болно. Хэрэв та энэ талбарт `off` гэж тавьбал `getty` үгүй болж порт дээр нэвтрэлт байхгүй болно.
- Сүүлийн талбар нь порт аюулгүй эсэхийг заахад хэрэглэгддэг. Портыг secure буюу аюулгүй болгох нь `root` бүртгэлийг (эсвэл хэрэглэгчийн UID 0 бүхий дурын бүртгэлийг) тэр портоос нэвтрэхийг зөвшөөрч хангалттай итгэнэ гэсэн үг юм. Insecure буюу аюулгүй биш портууд нь `root` нэвтрэлтүүдийг зөвшөөрдөггүй. Insecure буюу аюулгүй биш порт дээр хэрэглэгчид нь эрхгүй бүртгэлүүдээс нэвтрэн орж супер хэрэглэгчийн эрхүүдийг авахын тулд su(1) эсвэл өөр төстэй арга замыг ашиглах ёстой юм.Түгжээтэй хаалганы цаана байгаа терминалын хувьд ч гэсэн `insecure` тохируулгыг ашиглах нь хамгийн зөв зүйтэй юм. Хэрэв танд супер хэрэглэгчийн зөвшөөрлүүд хэрэгтэй бол нэвтрэн ороод `su` тушаалыг ашиглах нь их амархан байдаг.

#### 27.3.2.2. /etc/ttys-г `init` тушаалаар дахин уншуулах

/etc/ttys файлд шаардлагатай өөрчлөлтүүдийг хийсний дараа `init` процесс уруу өөрийн тохиргооных нь файлыг дахин уншуулахын тулд SIGHUP (hangup) дохиог илгээх хэрэгтэй:

```
# kill -HUP 1
```

`init` нь үргэлж систем дээр ажиллаж байгаа эхний процесс байдаг учир энэ нь үргэлж процесс ID 1-тэй байна.

Хэрэв бүгд зөв тохируулагдаж бүх кабелиуд нь байх ёстой газраа байж терминалууд нь ассан байгаа бол терминал бүр дээр `getty` процесс ажиллаж байх ёстой бөгөөд та энэ үед өөрийн терминалууд дээр нэвтрэлт хүлээх мөрүүдийг харах ёстой.

## 27.3.3. Өөрийн холболтод гарсан алдааг олж засварлах нь

Нягт нямбай анхаарал хандуулсан байлаа ч гэсэн терминалыг тохируулж байх явцад алдаа гарч буруутаж болох юм. Энд шинж тэмдгүүдийн жагсаалт болон зарим нэг санал болгосон засваруудыг үзүүлэв.

### 27.3.3.1. Нэвтрэлт хүлээх мөр гарч ирэхгүй байх

Терминал холбогдсон бөгөөд ассан байгаа эсэхийг шалгана. Энэ нь хэрэв терминал маягаар ажиллаж байгаа персонал компьютер бол зөв цуваа порт дээр терминал эмуляц хийх програм хангамж ажиллуулж байгаа эсэхийг шалгаарай.

Кабель терминал болон FreeBSD компьютерт сайн холбогдсон эсэхийг шалгах хэрэгтэй. Энэ нь зөв төрлийн кабель болохыг баталгаажуулж үзээрэй.

Терминал болон FreeBSD нь bps хурд болон parity тохиргоонууд дээр тохирч байгаа эсэхийг шалгаарай. Хэрэв та видео дэлгэцийн терминалтай бол хар цагаан болон тод байдлын хяналтууд таарсан эсэхийг үзээрэй. Хэрэв энэ нь хэвлэх терминал бол цаас болон бэх сайн эсэхийг шалгаарай.

`getty` процесс ажиллаж терминалд үйлчилж байгаа эсэхийг үзнэ. Жишээ нь ажиллаж байгаа `getty` процессуудын жагсаалтыг `ps` тушаалаар авахын тулд доор дурдсаныг бичнэ:

```
# ps -axww|grep getty
```

Та терминалын хувьд түүний оруулгыг харах ёстой. Жишээ нь дараах дэлгэц нь `getty` нь хоёр дахь цуваа порт ttyu1 дээр ажиллаж /etc/gettytab файлд `std.38400` гэсэн оруулгыг ашиглаж байгааг харуулж байна:

```
22189  d1  Is+    0:00.03 /usr/libexec/getty std.38400 ttyu1
```

Хэрэв ямар ч `getty` процесс ажиллахгүй байгаа бол /etc/ttys файлд та портыг идэвхжүүлсэн эсэхээ шалгаарай. Мөн ttys файлыг өөрчилснийхөө дараа `kill -HUP 1` тушаалыг ажиллуулахаа мартуузай.

Хэрэв `getty` процесс ажиллаж байгаа боловч терминал нь нэвтрэлт хүлээх мөрийг харуулахгүй байгаа, эсвэл энэ нь хүлээх мөрийг үзүүлж байгаа боловч танд бичихийг

зөвшөөрөхгүй байгаа бол таны терминал эсвэл кабель тоног төхөөрөмжийн handshaking буюу харилцан зохицож холбогдох горимыг дэмжихгүй байна гэсэн үг юм. /etc/ttys файл дахь оруулгыг `std.38400`-с `3wire.38400` уруу болгож үзээрэй (/etc/ttys файлыг өөрчилснийхөө дараа `kill -HUP 1` тушаалыг ажиллуулахаа мартуузай). `3wire` оруулга нь `std`-тэй төстэй боловч тоног төхөөрөмжийн handshaking хийдэггүй. Магадгүй та хурдыг багасгаж эсвэл `3wire`-г ашиглаж байхдаа буфер дүүрэхээс сэргийлж програм хангамжийн урсгалын хяналтыг идэвхжүүлэх хэрэгтэй байж болох юм.

#### 27.3.3.2. Нэвтрэлт хүлээх мөрийн оронд хог гарч ирвэл

Терминал болон FreeBSD нь bps хурд болон parity тохиргоонууд дээр тохирч байгаа эсэхийг шалгана. Зөв *getty* төрлийг ашиглаж байгаа эсэхийг баталгаажуулж `getty` процессуудыг шалгах хэрэгтэй. Хэрэв үгүй бол /etc/ttys-г засварлаж `kill -HUP 1` тушаалыг ажиллуулна.

#### 27.3.3.3. Тэмдэгтүүд хоёр удаа гарах; бичих үед нууц үг гарах

Терминалыг (эсвэл терминал эмуляц хийх програм хангамжийг) "half duplex" эсвэл "local echo" горимоос "full duplex" уруу шилжүүлнэ.

# 27.4. Dial-in буюу гаднаас залгах үйлчилгээ

FreeBSD системийг dial-in буюу гаднаас залгах үйлчилгээнд зориулж тохируулах нь терминалууд холбохтой маш төстэй бөгөөд ялгаатай тал нь та терминалуудын оронд модемуудтай ажиллах юм.

## 27.4.1. Гадаад, Дотоод модемууд

Гадаад модемууд нь dial-up буюу гадагш залгахад илүү тохиромжтой байдаг, учир нь гадаад модемууд нь хувирдаггүй RAM-д хадгалагдах параметрүүдээр хагас тогтмолоор ихэвчлэн тохируулагдсан байж болдог бөгөөд тэдгээр нь ихэвчлэн чухал RS-232 дохионуудын төлөвийг үзүүлдэг гэрлийн заагчтай байдаг.

Дотоод модемуудад ихэвчлэн хувирдаггүй RAM байдаггүй болохоор тэдгээрийн тохиргоо зөвхөн DIP шилжүүлэгчдийг тохируулснаар хязгаарлагдаж болох юм. Хэрэв таны дотоод модем ямар нэг дохио заагч гэрэлтэй бол системийн гадар байгаа нөхцөлд харахад хүнд байх байсан болов уу.

#### 27.4.1.1. Модемууд болон Кабелууд

Хэрэв та гадаад модем ашиглаж байгаа бол танд мэдээж зөв кабель хэрэгтэй болно. Хэвийн бүх дохионууд холбогдсон байхад стандарт RS-232C цуваа кабель хангалттай байх ёстой:

*Хүснэгт 12. Дохионы нэрс*

| Товчлолууд | Нэрс |
|---|---|
| RD | Received Data буюу Хүлээн авсан өгөгдөл |
| TD | Transmitted Data буюу Дамжуулсан өгөгдөл |

| Товчлолууд | Нэрс |
| --- | --- |
| DTR | Data Terminal Ready буюу Өгөгдлийн терминал бэлэн |
| DSR | Data Set Ready буюу Өгөгдлийн олонлог бэлэн |
| DCD | Data Carrier Detect буюу Өгөгдлийн зөөгч илрүүлэх (RS-232-ийн Хүлээн авсан Шугамын Дохио Илрүүлэгч) |
| SG | Signal Ground буюу Дохио газар |
| RTS | Request to Send буюу Илгээх хүсэлт |
| CTS | Clear to Send буюу Илгээхэд цэвэр |

2400 bps хурдаас дээш урсгалын хяналтад RTS болон CTS дохионууд FreeBSD-д хэрэгтэй байдаг. Мөн хэзээ дуудлага хүлээн авсан эсвэл шугам таслагдсаныг мэдэх CD дохио болон сесс бүрэн гүйцэд болсны дараа модемийг дахин тохируулах DTR дохионууд хэрэгтэй байдаг. Зарим кабелиуд нь бүх хэрэгцээтэй дохионуудгүйгээр холбогдсон байдаг болохоор шугам таслагдах үед нэвтрэх сесс байхгүй болохгүй байх зэрэг асуудлууд хэрэв танд байгаа бол таны кабель асуудалтай байж болох юм.

Бусад UNIX® төст үйлдлийн системүүдийн адил FreeBSD нь хэзээ дуудлагыг хүлээн авсан эсвэл шугам хэзээ таслагдсан болон дуудлагын дараа модемийг тасалж дахин тохируулахыг мэдэх тоног төхөөрөмжийн дохионуудыг ашигладаг. FreeBSD нь модем уруу тушаалуудыг илгээх эсвэл модемоос төлвийн тайлангуудыг харахаас зайлсхийдэг.

### 27.4.2. Цуваа интерфэйсийн эргэцүүллүүд

FreeBSD нь NS8250-, NS16450-, NS16550-, болон NS16550A-дээр тулгуурласан EIA RS-232C (CCITT V.24) холбооны интерфэйсүүдийг дэмждэг. 8250 болон 16450 интерфэйсүүд нь ганц тэмдэгтийн буфертай байдаг. 16550 төхөөрөмж нь системийн ажиллагааг илүү сайн болгох 16 тэмдэгтийн буфертай байдаг. Цэвэр 16550 дахь алдаанууд нь 16 тэмдэгтийн буферийн хэрэглээнээс зайлсхийдэг, тиймээс хэрэв боломжтой бол 16550A-г ашиглаарай. Ганц тэмдэгтийн буфер бүхий төхөөрөмжүүд нь 16 тэмдэгтийн буфер бүхий төхөөрөмжүүдээс илүү ажил үйлдлийн системээс шаарддаг учраас 16550A-дээр тулгуурласан цуваа интерфэйс картуудыг эрхэмлэх нь зүйтэй юм. Хэрэв систем олон идэвхтэй цуваа портуудтай бол эсвэл их ачаалалтай бол 16550A-дээр тулгуурласан картууд нь бага алдааны түвшин бүхий холбоонуудад илүү зохицдог.

### 27.4.3. Түргэн дүгнэлт

Терминалуудтай адил `init` нь гаднаас залгах холболтуудын хувьд тохируулсан цуваа порт бүрийн хувьд `getty` процессийг ажиллуулдаг. Жишээ нь хэрэв модем /dev/ttyu0 уруу залгагдсан бол `ps ax` тушаал ингэж харуулж болох юм:

```
4850 ??  I      0:00.09 /usr/libexec/getty V19200 ttyu0
```

Хэрэглэгч модемийн шугам залгаад модем холбогдох үед CD (Carrier Detect буюу Зөөгч

Илрүүлэгч) шугамыг модем гаргаж харуулна. Цөм зөөгч илрүүлэгдсэнийг мэдэж getty-ийн порт нээхийг гүйцээнэ. getty нь login: хүлээх мөрийг заагдсан эхний шугамын хурдаар илгээдэг. getty нь зөвшөөрөгдсөн тэмдэгтүүд хүлээн авагдсан эсэхийг мэдэхийн тулд харж байдаг бөгөөд ердийн тохиргоонд хэрэв энэ нь хог олох юм бол (модемийн холболтын хурд getty-ийн хурднаас өөр байгаагаас магадгүй болоод) getty нь боломжийн тэмдэгтүүд хүлээж автлаа шугамын хурдуудыг тааруулахыг оролдох болно.

Хэрэглэгч өөрийн нэвтрэх нэрийг оруулсны дараа getty тушаал хэрэглэгчийн нууц үгийг асууж нэвтрэлтийг гүйцээн хэрэглэгчийн бүрхүүлийг эхлүүлдэг /usr/bin/login тушаалыг ажиллуулдаг.

### 27.4.4. Тохиргооны файлууд

/etc санд гурван системийн тохиргооны файл байдаг бөгөөд өөрийн FreeBSD системд dial-up буюу гадагш залгах боломжийг олгохын тулд та магадгүй засварлах хэрэгтэй болно. Эхний /etc/gettytab нь /usr/libexec/getty демоны тохиргооны мэдээллийг агуулдаг. Хоёр дахь /etc/ttys нь getty процессуудын ажиллах tty төхөөрөмжүүдэд юу байх ёстойг /sbin/init тушаалд хэлж өгөх мэдээллийг агуулдаг. Сүүлд нь порт эхлүүлэх тушаалуудыг та /etc/rc.d/serial скриптэд байрлуулж болно.

UNIX® дээр dial-up модемуудын талаар санаа бодлын хоёр сургууль байдаг. Нэг бүлэг нь алсын хэрэглэгч ямар ч хурдаар залгасан байсан гэсэн локал компьютераас модемд холбогдсон RS-232 интерфэйс түгжигдсэн хурдаар ажиллахаар өөрсдийн модемууд болон системүүдийг тохируулах дуртай байдаг. Энэ тохиргооны ашигтай тал нь алсын хэрэглэгч системийн нэвтрэлт хүлээх мөрийг үргэлж нэн даруй хардаг явдал юм. Сул тал нь систем хэрэглэгчийн өгөгдлийн жинхэнэ хурдыг мэддэггүй явдал бөгөөд Emacs зэрэг бүрэн дэлгэцийн програмууд нь удаан холболтуудын хувьд хариултыг сайжруулахын тулд өөрийн дэлгэц будах аргуудаа тааруулахгүй байх болно.

Нөгөө нэг сургууль нь өөрсдийн модемуудынхаа RS-232 интерфэйсийн хурдыг алсын компьютерийн холболтын хурд дээр тулгуурлан өөрчилдгөөр тохируулдаг. Жишээ нь модем уруу хийгдэх V.32bis (14.4 Kbps) холболтууд нь модемийг өөрийн RS-232 интерфэйсийг 19.2 Kbps хурдаар ажиллуулах бол 2400 bps холболтууд нь модемийн RS-232 интерфэйсийг 2400 bps хурд дээр ажиллуулдаг. getty нь тухайн ямар нэг модемийн холболтын хурдны тайлагналтыг ойлгодоггүй учраас getty тушаал нь login: мэдэгдлийг эхний хурдан дээр өгдөг бөгөөд хариултаас ирэх тэмдэгтүүдийг харж байдаг. Хэрэв хэрэглэгч хог харвал тэдгээр нь танигдах хүлээх мөрийг хартлаа `Enter` товчлуурыг дарах ёстой гэдгээ мэдэж байгаа гэж тооцдог. Хэрэв өгөгдлийн хурд таарахгүй бол getty нь хэрэглэгчийн бичсэн болгоныг "хог" гэж харах бөгөөд дараагийн хурдыг оролдож login: хүлээх мөрийг дахин өгдөг. Энэ процедур дотор муухайрах түвшинд хүртэл үргэлжилж болох боловч хэрэглэгч зөв хүлээх мөрийг харах хүртэл ихэвчлэн ганц хоёр товчлуур даруулдаг. Мэдээж энэ нэвтрэлтийн дараалал нь түрүүчийн "locked-speed буюу түгжигдсэн хурд"тай адил цэвэр байдаггүй боловч бага хурдтай холболтон дээр байх хэрэглэгчийн хувьд бүрэн дэлгэцийн програмуудаас илүү сайн интерактив хариултыг хүлээн авах ёстой юм.

Энэ хэсэг нь тэнцвэртэй тохиргооны мэдээллийг өгөхийг оролдох боловч модемийн өгөгдлийн хурд холболтын хурдыг дагах тал уруу илүү хазайсан байгаа болно.

#### 27.4.4.1. /etc/gettytab

/etc/gettytab нь getty(8)-ийн тохиргооны мэдээллийн termcap(5) загварын файл юм. Файлын хэлбэр болон боломжуудын жагсаалтын тухай бүрэн мэдээллийг gettytab(5) гарын авлагын хуудаснаас үзнэ үү.

##### 27.4.4.1.1. Locked-speed буюу түгжигдсэн хурд тохиргоо

Хэрэв та өөрийн модемийн өгөгдлийн холбоонуудын хурдыг тухайн нэг хурдан дээр түгжиж байгаа бол магадгүй /etc/gettytab файлд ямар нэг өөрчлөлт хийх шаардлагагүй юм.

##### 27.4.4.1.2. Matching-speed буюу таарах хурд тохиргоо

Өөрийн модемд зориулж ашиглахыг хүссэн хурдны талаар `getty`-д мэдээлэл өгөхийн тулд /etc/gettytab файлд та оруулга тохируулах хэрэгтэй болно. Хэрэв та 2400 bps модемтой бол байгаа `D2400` оруулгыг магадгүй ашиглаж болох юм.

```
#
# Fast dialup terminals, 2400/1200/300 rotary (can start either way)
#
D2400|d2400|Fast-Dial-2400:\
        :nx=D1200:tc=2400-baud:
3|D1200|Fast-Dial-1200:\
        :nx=D300:tc=1200-baud:
5|D300|Fast-Dial-300:\
        :nx=D2400:tc=300-baud:
```

Хэрэв та илүү хурдтай модемтой бол /etc/gettytab файлд магадгүй оруулга нэмэх хэрэгтэй болж болох юм; энд интерфэйсийн дээд хурд нь 19.2 Kbps бүхий 14.4 Kbps модемд зориулан таны ашиглах оруулга байна:

```
#
# Additions for a V.32bis Modem
#
um|V300|High Speed Modem at 300,8-bit:\
        :nx=V19200:tc=std.300:
un|V1200|High Speed Modem at 1200,8-bit:\
        :nx=V300:tc=std.1200:
uo|V2400|High Speed Modem at 2400,8-bit:\
        :nx=V1200:tc=std.2400:
up|V9600|High Speed Modem at 9600,8-bit:\
        :nx=V2400:tc=std.9600:
uq|V19200|High Speed Modem at 19200,8-bit:\
        :nx=V9600:tc=std.19200:
```

Энэ нь 8 бит, parity байхгүй холболтуудад хүргэнэ.

Дээрх жишээ нь холболтын хурдаа 19.2 Kbps (V.32bis холболтын хувьд) гэж эхлүүлж дараа нь 9600 bps (V.32-ийн хувьд), 2400 bps, 1200 bps, 300 bps, гэж оролдоод 19.2 Kbps хурд уруу буцах

болно. Холбооны хурдны тойролт нь `nx=` ("next table" буюу дараагийн хүснэгт) боломжтой цуг хийгдсэн байдаг. Мөр бүр нь өгөгдлийн тухайн хурдны хувьд "стандарт" тохиргоонуудын үлдсэнийг авахын тулд `tc=` ("table continuation" буюу хүснэгтийн үргэлжлэл) оруулгыг ашигладаг.

Хэрэв та 28.8 Kbps модемтой ба/эсвэл та 14.4 Kbps модем дээр шахалтын давуу талыг авахыг хүсвэл 19.2 Kbps-ээс илүү холбооны хурдыг ашиглах хэрэгтэй юм. Энд 57.6 Kbps-ээс эхлэх gettytab оруулгын жишээ байна:

```
#
# Additions for a V.32bis or V.34 Modem
# Starting at 57.6 Kbps
#
vm|VH300|Very High Speed Modem at 300,8-bit:\
        :nx=VH57600:tc=std.300:
vn|VH1200|Very High Speed Modem at 1200,8-bit:\
        :nx=VH300:tc=std.1200:
vo|VH2400|Very High Speed Modem at 2400,8-bit:\
        :nx=VH1200:tc=std.2400:
vp|VH9600|Very High Speed Modem at 9600,8-bit:\
        :nx=VH2400:tc=std.9600:
vq|VH57600|Very High Speed Modem at 57600,8-bit:\
        :nx=VH9600:tc=std.57600:
```

Хэрэв та удаан CPU-тэй эсвэл ачаалал ихтэй системтэй бөгөөд 16550A дээр тулгуурласан цуваа портуудтай бол 57.6 Kbps хурд дээр та `sio` "silo" гэсэн алдаанууд хүлээн авч болох юм.

#### 27.4.4.2. /etc/ttys

/etc/ttys файлын тохиргоо /etc/ttys уруу терминалын оруулгууд нэмэх нь хэсэгт яригдсан байгаа. Модемуудын тохиргоо төстэй боловч бид өөр терминалын төрлийг заахын тулд `getty` уруу өөр нэмэлт өгөгдлүүдийг дамжуулах ёстой. Түгжигдсэн хурдтай болон таарах хурдтай тохиргоонуудын ерөнхий хэлбэр нь иймэрхүү байна:

```
ttyu0   "/usr/libexec/getty xxx"   dialup on
```

Дээрх мөрний эхнийх нь энэ оруулгад зориулсан төхөөрөмжийн тусгай файл юм - ttyu0 нь /dev/ttyu0 гэсэн үг бөгөөд үүнийг `getty` харж байдаг. Хоёр дахь `"/usr/libexec/getty xxx"` (*xxx* нь эхний gettytab боломжоор солигдоно) нь энэ төхөөрөмж дээр `init`-ийн ажиллуулах процесс юм. Гурав дахь `dialup` гэдэг нь терминалын анхдагч төрөл юм. Дөрөв дэх параметр `on` гэдэг нь `init`-д шугам ажиллагаатай гэдгийг заана. Тав дахь параметр `secure` байж болох боловч энэ нь зөвхөн физикийн хувьд аюулгүй терминалуудад (системийн консолд) ашиглагдах ёстой юм.

Анхдагч терминалын төрөл нь (дээрх жишээний `dialup`) локал тохируулгуудаас хамаарна. `dialup` нь dial-up шугамууд дээрх уламжлалт терминалын анхдагч төрөл бөгөөд хэрэглэгчид нь өөрсдийн нэвтрэх скриптүүдийг терминал `dialup` байгаа үед мэдэхээр өөрчилж өөрсдийн терминалын төрлийг автоматаар тааруулж болох юм. Гэхдээ зохиогч өөрийн сайт дээр

анхдагч терминалын төрлөө vt102 гэх нь хялбар болохыг олсон бөгөөд хэрэглэгчид өөрсдийн алсын системүүд дээрээ ердөө л VT102 эмуляцыг ашигладаг болохоор тэр юм.

Та /etc/ttys файлд өөрчлөлтүүд хийснийхээ дараа init процессод файлыг дахин уншуулахын тулд HUP дохиог илгээж болох юм. Та дараах тушаалыг ашиглаж

```
# kill -HUP 1
```

дохио илгээж болно. Хэрэв та анх удаагаа системийг тохируулж байгаа бол өөрийн модем(үүд)ийг зөв тохируулж init уруу дохиолохоос өмнө холбогдох хүртэл хүлээхийг хүсэж болох юм.

##### 27.4.4.2.1. Locked-speed буюу түгжигдсэн хурд тохиргоо

Түгжигдсэн хурд тохиргооны хувьд таны ttys оруулга getty уруу өгөгдөх тогтмол хурд бүхий оруулгатай байх хэрэгтэй. Портын хурд нь 19.2 Kbps гэж түгжигдсэн модемийн хувьд ttys оруулга нь иймэрхүү харагдаж болох юм:

```
ttyu0   "/usr/libexec/getty std.19200"   dialup on
```

Хэрэв таны модем өөр хурдан дээр түгжигдсэн бол std.19200-ийн оронд std.speed-ийн хувьд тохирох утгыг тавьж солиорой. Та /etc/gettytab-д жагсаагдсан зөв төрлийг ашигласан эсэхээ шалгаарай.

##### 27.4.4.2.2. Matching-speed буюу таарах хурд тохиргоо

Таарах хурд тохиргоон дээр таны ttys оруулга нь /etc/gettytab дахь тохирох зөв "auto-baud" (sic) оруулгаас авах хэрэгтэй болдог. Жишээ нь хэрэв та дээрх санал болгосон оруулгыг 19.2 Kbps-ээс (V19200 эхлэх цэгийг агуулах gettytab оруулга) эхлэх таарах хурд бүхий модемийн хувьд нэмсэн бол таны ttys оруулга иймэрхүү харагдах болно:

```
ttyu0   "/usr/libexec/getty V19200"   dialup on
```

#### 27.4.4.3. /etc/rc.d/serial

V.32, V.32bis, болон V.34 зэрэг өндөр хурдтай модемууд тоног төхөөрөмжийн (RTS/CTS) урсгал хяналтыг ашиглах хэрэгтэй. Та модемийн портуудын хувьд FreeBSD цөмд тоног төхөөрөмжийн урсгалын хяналтын тугийг тохируулахын тулд stty тушаалуудыг /etc/rc.d/serial файлд нэмж болно.

Жишээ нь гаднаас залгах болон гадагш залгах эхний төхөөрөмжүүдийн хувьд цуваа порт #1 дээр crtscts гэсэн termios-ийн тугийг тохируулахын тулд дараах мөрүүдийг /etc/rc.d/serial файлд нэмж болох юм:

```
# Serial port initial configuration
stty -f /dev/ttyu1.init crtscts
```

```
stty -f /dev/cuau1.init crtscts
```

### 27.4.5. Модемийн тохиргоонууд

Хэрэв та хувирдаггүй RAM-д параметрүүд нь тогтмолоор тохируулагдсан байж болох модемтой бол параметрүүдийг тохируулахын тулд та терминал програм `tip` гэх мэт) ашиглах хэрэгтэй болно. Эхний хурдтай адил холболтын хурдыг ашиглан модем уруу холбогдохын тулд `getty` нь доор дурдсан шаардлагуудтай таарахын тулд модемийн хувирдаггүй RAM-ийг хэрэглэж тохируулах болно:

- Холбогдох үед CD шалгагдана
- Үйлдлийн хувьд DTR шалгагдана; DTR-ийг хаях нь шугамыг салгаж модемийг дахин тохируулахад хүргэнэ
- CTS нь өгөгдлийн урсгалын хяналтыг дамжуулсан
- XON/XOFF урсгалын хяналтыг хаана
- RTS нь өгөгдлийн урсгалын хяналтыг хүлээн авсан
- Чимээгүй горим (үр дүнгийн кодууд байхгүй)
- Тушаалын цуурай байхгүй

Аль тушаалууд ба/эсвэл DIP шилжүүлэгчийн тохиргоонуудыг та өгөх ёстой болохыг олохын тулд өөрийн модемийн баримтыг уншина уу.

Жишээ нь U.S. Robotics® Sportster® 14,400 хурдтай гадаад модем дээр дээрх параметрүүдийг тохируулахын тулд эдгээр тушаалуудыг модемд өгч болох юм:

```
ATZ
AT&C1&D2&H1&I0&R2&W
```

V.42bis ба/эсвэл MNP5 шахалтыг ашиглах зэрэг модемийн бусад тохиргоонуудыг тааруулахын тулд та бас энэ боломжийг ашиглахыг хүсэж болох юм.

U.S. Robotics® Sportster® 14,400 хурдтай гадаад модем нь тохируулах хэрэгтэй DIP шилжүүлэгчүүдтэй бас байдаг; бусад модемуудын хувьд та эдгээр тохируулгуудыг жишээ болгон ашиглаж болох юм:

- Шилжүүлэгч 1: ДЭЭШ - DTR Хэвийн
- Шилжүүлэгч 2: N/A буюу Байхгүй (Үр дүнгийн үгэн кодууд/Үр дүнгийн тоон кодууд)
- Шилжүүлэгч 3: ДЭЭШ - Үр дүнгийн дарах/хориглох кодууд
- Шилжүүлэгч 4: ДООШ - Цуурай байхгүй, холбогдоогүй үеийн тушаалууд
- Шилжүүлэгч 5: ДЭЭШ - Автомат хариулна
- Шилжүүлэгч 6: ДЭЭШ - Зөөгч илрүүлэлт хэвийн
- Шилжүүлэгч 7: ДЭЭШ - NVRAM анхдагчуудыг ачаалах
- Шилжүүлэгч 8: N/A буюу Байхгүй (Ухаалаг Горим/Дүлий Горим)

Хэрэв `getty` тушаал нь `login:` хүлээх мөрийг тушаалын горимд байгаа модем уруу андуурч өгсөн бөгөөд модем нь тушаалыг цуурайтуулан буцааж эсвэл үр дүнгийн кодыг буцаахад үүсч болох асуудлуудаас сэргийлэхийн тулд үр дүнгийн кодууд нь dial-up модемуудын хувьд хаагдсан/дарагдсан байх ёстой. Энэ дараалал нь `getty` болон модемийн хооронд сунасан, тэнэг ярианд хүргэж болох юм.

#### 27.4.5.1. Locked-speed буюу түгжигдсэн хурд тохиргоо

Түгжигдсэн хурд тохиргооны хувьд та модемийг холболтын хурдаас хамааралгүйгээр модемоос компьютер хүртэлх өгөгдлийн хурдыг тогтмолоор байлгахаар тохируурах хэрэгтэй. U.S. Robotics® Sportster® 14,400 хурдтай гадаад модем дээр эдгээр тушаалууд нь модемоос компьютер хүртэлх өгөгдлийн хурдыг тушаалуудыг ажиллуулах хурд дээр түгжих болно:

```
ATZ
AT&B1&W
```

#### 27.4.5.2. Matching-speed буюу таарах хурд тохиргоо

Хувьсах хурд тохиргооны хувьд ирж байгаа дуудлагын хурдтай ижил байлгаж өөрийн цуваа портын өгөгдлийн хурдыг тааруулахын тулд та өөрийн модемийг тохируулах хэрэгтэй болно. U.S. Robotics® Sportster® 14,400 хурдтай гадаад модем дээр эдгээр тушаалууд нь модемийн алдаа засварлах өгөгдлийн хурдыг тушаалуудыг ажиллуулах хурд дээр түгжих боловч цуваа портын хурдыг алдаа засварлахүй холболтуудын хувьд хувьсах байхаар зөвшөөрөх болно:

```
ATZ
AT&B2&W
```

#### 27.4.5.3. Модемийн тохиргоог шалгах нь

Ихэнх өндөр хурдтай модемууд нь модемийн тухайн үеийн үйлдлийн параметрүүдийг хүн уншиж болохоор үзүүлэх тушаалуудтай байдаг. U.S. Robotics® Sportster® 14,400 хурдтай гадаад модемууд дээр `ATI5` тушаал хувирдаггүй RAM-д хадгалагдаж байгаа тохиргоонуудыг харуулдаг. Модемийн үйлдлийн жинхэнэ параметрүүдийг харахын тулд (модемийн DIP шилжүүлэгчийн тохиргоонууд нөлөөлсөн) `ATZ` болон дараа нь `ATI4` тушаалуудыг ашиглана.

Хэрэв та өөр загварын модемтой бол өөрийн модемийн тохиргооны параметрүүдийг хэрхэн давхар шалгах талаар өөрийн модемийн гарын авлагаас шалгаарай.

### 27.4.6. Алдааг олж засварлах нь

Өөрийн систем дээр dial-up модемийг шалгахын тулд таны дагах цөөн хэдэн алхмууд энд байна.

#### 27.4.6.1. FreeBSD системийг шалгах

Өөрийн модемийг өөрийн FreeBSD системд залгаад системээ ачаалах бөгөөд хэрэв таны модем төлөв заах гэрлүүдтэй бол системийн консол дээр `login:` хүлээх мөр гарч ирэх үед модемийн DTR заагч асаж байгаа эсэхийг шалгах хэрэгтэй - хэрэв асаж байгаа бол FreeBSD нь тохирох холболтуудын порт дээр `getty`-г эхлүүлж модемийг дуудлага хүлээж авахыг хүлээж байна гэсэн үг юм.

Хэрэв DTR заагч асахгүй бол FreeBSD систем уруу консолоор нэвтрээд FreeBSD нь зөв порт дээр `getty` процессийг ажиллуулахаар оролдож байгаа эсэхийг харахын тулд `ps ax` тушаалыг ажиллуулна. Та эдгээртэй төстэй мөрүүдийг процессуудын хамтаар харах ёстой:

```
114 ??  I      0:00.10 /usr/libexec/getty V19200 ttyu0
115 ??  I      0:00.10 /usr/libexec/getty V19200 ttyu1
```

Хэрэв та үүнтэй адил өөр:

```
114 d0  I      0:00.10 /usr/libexec/getty V19200 ttyu0
```

мөрийг харсан бөгөөд модем нь дуудлага хүлээж авч амжаагүй байгаа бол `getty` нь холболтуудын порт дээр өөрийн онгойлголтыг бүрэн хийж гүйцсэн гэсэн үг юм. Энэ нь кабелийн асуудал эсвэл буруу тохируулсан модемтой холбоотой асуудлыг зааж болох юм. Яагаад гэвэл `getty` нь CD-г (carrier detect буюу зөөгч илрүүлэх) модемоор шалгагдтал холболтуудын портуудыг онгойлгож чадахгүй байх ёстой юм.

Хэрэв та хүссэн ttyuN портыг онгойлгохыг хүлээж байгаа ямар нэг `getty` процессуудыг харахгүй байгаа бол /etc/ttys файл дахь өөрийн оруулгуудад алдаа байгаа эсэхийг давхар шалгаарай. Мөн `init` эсвэл `getty` тушаалаас гарсан бүртгэлийн ямар нэг мэдэгдлүүд асуудлууд байгаа эсэхийг мэдээлсэн эсэхийг харахын тулд бүртгэлийн файл /var/log/messages -г бас шалгаарай. Хэрэв ямар нэг мэдэгдлүүд байгаа бол /etc/ttys болон /etc/gettytab тохиргооны файлуудыг гурав дахин шалгаарай. Мөн /dev/ttyuN зэрэг тохирох төхөөрөмжийн тусгай файлуудад алдаанууд, оруулгууд дутагдаж байгаа эсэх, эсвэл төхөөрөмжийн тусгай файлууд дутагдаж байгаа эсэхийг бас шалгаарай.

#### 27.4.6.2. Гаднаас залгаж үзэх

Систем уруу гаднаас залгаж үзээрэй; 8 битийг parity байхгүйгээр болон алсын систем дээр 1 зогсох битийг ашигласан эсэхээ шалгаарай. Хэрэв та хүлээх мөрийг тэр дор нь хүлээж авахгүй бол эсвэл хог хүлээн авсан бол секунд тутам нэг `Enter` дараад оролдоод үзээрэй. Тэгсэн ч гэсэн та `login:` хүлээх мөрийг хэсэг хугацаа өнгөрсөн ч гэсэн харахгүй байгаа бол `BREAK` илгээж оролдоод үзээрэй. Та залгахдаа өндөр хурдтай модем ашиглаж байгаа бол модемийн интерфэйсийн хурдыг (жишээ нь U.S. Robotics® Sportster® модем дээр `AT&B1` тушаал ашиглан) түгжсэнийхээ дараа дахин залгаад оролдоод үзээрэй.

Хэрэв та тэгсэн ч гэсэн `login:` хүлээх мөрийг хүлээн авахгүй байгаа бол /etc/gettytab файлыг дахин шалгаад доор дурдсаныг давхар шалгаарай

- Шугамын хувьд /etc/ttys файлд заагдсан боломжийн эхний нэр /etc/gettytab дахь

боломжийн нэртэй тааруулж байгаа эсэх

- `nx=` оруулга бүр gettytab боломжийн өөр нэртэй тааруулж байгаа эсэх
- `tc=` оруулга бүр gettytab боломжийн өөр нэртэй тааруулж байгаа эсэх

Хэрэв таныг залгахад FreeBSD систем дээрх модем хариулахгүй байгаа бол DTR шалгагдах үед модем нь утсанд хариулахаар тохируулагдсан эсэхийг шалгаарай. Хэрэв модем зөв тохируулагдсан юм шиг байвал DTR шугам шалгагдсан эсэхийг модемийн заагч гэрлүүдийг (гэрлүүдтэй бол) шалган нягтлаарай.

Хэрэв та бүгдийг хэд хэдэн удаа шалгасан бөгөөд ажиллахгүй хэвээр байгаа бол завсарлага авааД дараа нь үзээрэй. Хэрэв тэгсэн ч гэсэн ажиллахгүй байгаа бол та FreeBSD ерөнхий асуултууд захидлын жагсаалт уруу өөрийн модем болон асуудлынхаа талаар цахим захидал илгээж болох бөгөөд жагсаалтан дахь сайн хүмүүс танд туслахыг оролдох болно.

# 27.5. Dial-out буюу гадагш залгах үйлчилгээ

Дараах нь таны хостыг модемоор өөр компьютер уруу холбох зааврууд юм. Энэ нь алсын хосттой терминал сесс үүсгэхэд тохирох юм.

Хэрэв та PPP дээр асуудалтай байгаа бол Интернэт дэх файлыг авахад энэ төрлийн холболт нь туйлын ашигтай байж болох юм. Хэрэв та ямар нэг зүйл FTP хийх хэрэгтэй бөгөөд PPP эвдэрсэн бол терминал сесс ашиглан FTP хийж болно. Дараа нь түүнийг өөрийн машин уруу дамжуулахын тулд zmodem ашиглана.

## 27.5.1. Hayes модем ашиглах нь

Энгийн Hayes залгагч `tip`-д цуг бүтээгдсэн байгаа болно. /etc/remote файлд `at=hayes`-г ашиглаарай.

Hayes драйвер нь шинэ модемуудын дэвшилтэт боломжуудын заримыг таних хэмжээний ухаалаг байдаггүй бөгөөд `BUSY`, `NO DIALTONE`, эсвэл `CONNECT 115200` зэрэг мэдэгдлүүд нь түүнийг самууруулах болно. Та `tip`-ийг хэрэглэхдээ тэдгээр мэдэгдлүүдийг (`ATX0&W`-г ашиглаж) хаах хэрэгтэй.

Бас `tip`-ийн хувьд залгах хугацаа нь 60 секунд байдаг. Таны модем нь үүнээс багыг ашиглах ёстой бөгөөд тэгэхгүй бол tip холбоонд асуудал байна гэж үзэх болно. `ATS7=45&W` гэж үзээрэй.

## 27.5.2. `AT` тушаалуудыг ашиглах нь

Өөрийн /etc/remote файлдаа "шууд" гэгддэг оруулгыг хийх хэрэгтэй. Жишээ нь хэрэв таны модем эхний цуваа порт /dev/cuau0-д холбогдсон бол дараах мөрийг тавьна:

```
cuau0:dv=/dev/cuau0:br#19200:pa=none
```

`br` боломжид таны модемийн дэмждэг хамгийн их bps хурдыг ашиглаарай. Дараа нь `tip cuau0` гэж бичихэд та өөрийн модемд холбогдох болно.

`root` хэрэглэгчээр `cu`-г дараах маягаар ашиглаарай:

```
# cu -lline -sspeed
```

*line* нь цуваа порт (өөрөөр хэлбэл /dev/cuau0) бөгөөд *speed* нь хурд (өөрөөр хэлбэл `57600`) юм. Та AT тушаалуудыг оруулж дуусаад гарахын тулд `~.` дарна.

### 27.5.3. `@` тэмдэг ажиллахгүй байна

Утасны дугаарын боломжийн `@` тэмдэг нь `tip`-ийг /etc/phones файлаас утасны дугаар хайхыг заадаг. Гэхдээ `@` тэмдэг нь бас /etc/remote зэрэг боломжийн файлуудын тусгай тэмдэгт юм. Үүнийг урагш ташуу зураас ашиглан оруулж болно:

```
pn=\@
```

### 27.5.4. Тушаалын мөрнөөс залгах

Өөрийн /etc/remote файлд "ердийн" гэгддэг оруулгыг оруулна. Жишээ нь:

```
tip115200|Dial any phone number at 115200 bps:\
        :dv=/dev/cuau0:br#115200:at=hayes:pa=none:du:
tip57600|Dial any phone number at 57600 bps:\
        :dv=/dev/cuau0:br#57600:at=hayes:pa=none:du:
```

Дараа нь та иймэрхүү зүйлс хийж болно:

```
# tip -115200 5551234
```

Хэрэв та `cu`-г `tip`-ээс илүүд үздэг бол ердийн `cu` оруулгыг ашиглаж:

```
cu115200|Use cu to dial any number at 115200bps:\
        :dv=/dev/cuau1:br#57600:at=hayes:pa=none:du:
```

дараа нь ингэж бичнэ:

```
# cu 5551234 -s 115200
```

### 27.5.5. bps хурдыг тохируулах

`tip1200` эсвэл `cu1200`-ийн хувьд оруулгыг хийнэ, гэхдээ `br` боломжтой тохирох ямар ч bps хурдыг ашиглаж болно. `tip` нь боломжийн анхдагч утга бол 1200 bps гэж үздэг. `tip1200` оруулгыг хайдгийн учир нь ийм юм. Гэхдээ та заавал 1200 bps хурдыг ашиглах шаардлагагүй юм.

### 27.5.6. Би терминал серверээр хэд хэдэн хостууд уруу ханддаг

Та холбогдох хүртлээ хүлээж `CONNECT host` гэж тухай бүрт бичиж байхын оронд `tip`-ийн `cm` боломжийг ашиглаж болно. Жишээ нь /etc/remote файл дахь эдгээр оруулгууд нь `pain` эсвэл `muffin` хостууд уруу холбогдохын тулд танд `tip pain` эсвэл `tip muffin` гэж бичих боломжийг олгохоос гадна терминал сервер уруу орохын тулд `tip deep13` гэж бичих боломжийг бас олгоно:

```
pain|pain.deep13.com|Forrester's machine:\
        :cm=CONNECT pain\n:tc=deep13:
muffin|muffin.deep13.com|Frank's machine:\
        :cm=CONNECT muffin\n:tc=deep13:
deep13:Gizmonics Institute terminal server:\
        :dv=/dev/cuau2:br#38400:at=hayes:du:pa=none:pn=5551234:
```

### 27.5.7. `tip`-д нэгээс илүү олон мөрийг ашиглах

Энэ нь их сургууль хэд хэдэн модемийн шугамуудтай бөгөөд хэдэн мянган оюутнууд тэдгээрийг ашиглахыг оролдох үед ихэвчлэн гардаг асуудал юм.

Өөрийн их сургуулийн оруулгыг /etc/remote файлд хийгээд `pn` боломжийн хувьд `@`-ийг ашиглаарай:

```
big-university:\
        :pn=\@:tc=dialout
dialout:\
        :dv=/dev/cuau3:br#9600:at=courier:du:pa=none:
```

Дараа нь их сургуулиудын хувьд утасны дугааруудыг /etc/phones файлд жагсаана:

```
big-university 5551111
big-university 5551112
big-university 5551113
big-university 5551114
```

`tip` нь жагсаагдсан дарааллаар утас болгоныг оролдож дараа нь зогсоно. Та оролдлогоо үргэлжлүүлсээр байхын тулд `tip`-ийг while давталтад ажиллуулах хэрэгтэй.

### 27.5.8. Хүчлэх тэмдэгтийг хэрэглэх

`Ctrl` + `P` нь анхдагч "force буюу хүчлэх" тэмдэгт бөгөөд `tip` тушаалд дараагийн тэмдэгт нь үсгэн өгөгдөл гэдгийг хэлэхэд хэрэглэгддэг. Та хүчлэх тэмдэгтийг дурын тэмдэгт болгон `~s` тушаалаар зааж өгч болно. `~s` тушаал нь "хувьсагчийг тохируулах" гэсэн үг юм.

`~sforce=single-char` гэж бичээд шинэ мөр авах хэрэгтэй. *single-char* нь дурын ганц тэмдэгт юм. Хэрэв та *single-char*-г орхих юм бол хүчлэх тэмдэгт нь хоосон тэмдэгт болох бөгөөд та үүнийг

`Ctrl` + `2` эсвэл `Ctrl` + `Space` гэж бичин авч болно. *single-char*-ийн хувьд боломжийн утга `Shift` + `Ctrl` + `6` байж болох бөгөөд энэ нь зөвхөн зарим нэг терминал серверүүдийн хувьд ашиглагддаг.

Та хүчлэх тэмдэгтийг өөрийн хүссэнээр $HOME/.tiprc файлд доор дурдсаныг зааж тохируулж болно:

```
force=single-char
```

### 27.5.9. Том үсэг ашиглах

Та эвдэгдсэн caps-lock товчлуур бүхий хэрэглэгчдэд тусгайлан зориулагдсан `tip`-ийн "өсгөх тэмдэгт" болох `Ctrl` + `A` хослолыг дарсан байна. Дээр дурдсаны адил `~s` тушаалыг ашиглан `raisechar` хувьсагчийг боломжийн утгаар тохируулах хэрэгтэй. Та үүнийг хүчлэх тэмдэгттэй адилаар тохируулж болох юм. Өөрөөр хэлбэл хэрэв та эдгээр боломжуудын аль нэгийг ашиглах бол ингэж болох юм.

`Ctrl` + `2` болон `Ctrl` + `A` хослолуудыг байнга бичих хэрэгтэй болдог Emacs хэрэглэгчдэд яг зориулагдсан жишээ .tiprc файл энд байна:

```
force=^^
raisechar=^^
```

`^^` нь `Shift` + `Ctrl` + `6` юм.

### 27.5.10. `tip` ашиглан файл дамжуулах

Хэрэв та өөр UNIX® системтэй харьцаж байвал та `~p` (put буюу тавих) болон `~t` (take буюу авах) тушаалуудыг ашиглан файлуудыг илгээж хүлээн авч болно. Эдгээр тушаалууд нь файлуудыг хүлээн авч илгээхийн тулд алсын систем дээр `cat` болон `echo` тушаалуудыг ажиллуулдаг. Синтакс нь ийм байна:

`~p` local-file [ remote-file ]

`~t` remote-file [ local-file ]

Алдаа шалгалт байдаггүй учраас магадгүй та zmodem-той адил өөр протокол ашиглах хэрэгтэй юм.

### 27.5.11. `tip`-тэй zmodem-ийг ашиглах

Файлуудыг хүлээн авахын тулд алсын машин дээр програмыг илгээж эхэлнэ. Дараа нь тэдгээрийг локалаар хүлээн авч эхлэхийн тулд `~C rz` гэж бичнэ.

Файлуудыг илгээхийн тулд алсын машин дээр програмыг хүлээн авч эхэлнэ. Дараа нь тэдгээрийг алсын систем уруу илгээхийн тулд `~C sz files` гэж бичнэ.

# 27.6. Цуваа консолыг тохируулах нь

## 27.6.1. Танилцуулга

FreeBSD нь цуваа порт дээр консол маягаар ашиглах дүлий терминал бүхий систем дээр ачаалах чадвартай байдаг. Ийм тохиргоо нь хоёр ангиллын хүмүүст ашигтай байх ёстой бөгөөд эдгээр нь гар эсвэл монитор байхгүй машинууд дээр FreeBSD суулгахыг хүссэн системийн администраторууд болон цөм эсвэл төхөөрөмжийн драйверуудыг дибаг хийхийг хүссэн хөгжүүлэгчид юм.

FreeBSD-ийн Ачаалах процесс-т тайлбарласнаар FreeBSD нь гурван шаттай эхлүүлэгчийг хэрэглэдэг. Эхний хоёр шат нь ачаалах диск дээрх FreeBSD-ийн зүсмэлийн эхлэл дээр хадгалагддаг ачаалах блокийн кодод байдаг. Ачаалах блок нь ачаалагч дуудагчийг (/boot/loader) гурав дахь шатны код болгон дуудаж ажиллуулдаг.

Цуваа консолыг тохируулахын тулд та ачаалах блокийн код, ачаалагч дуудах код болон цөмөө тохируулах ёстой болно.

## 27.6.2. Цуваа консолын тохиргоо, Товч хувилбар

Энэ хэсэг нь таныг анхдагч тохируулга ашиглаж байгаа бөгөөд цуваа консолыг тохируулах хурдан тоймыг авахыг хүсэж байгаа гэж үздэг.

1. Цуваа кабелийг COM1 болон хянаж байгаа терминал уруу холбоно.
2. Цуваа консол дээр бүх ачаалалтын мэдэгдлүүдийг харахын тулд супер хэрэглэгчээр нэвтэрснийхээ дараа дараах тушаалыг ажиллуулна:

   ```
   # echo 'console="comconsole"' >> /boot/loader.conf
   ```

3. /etc/ttys-г засварлаж ttyu0 оруулгын хувьд `off` гэснийг `on`, `dialup` гэснийг `vt100` болгоно. Ингэхгүй бол цуваа консолоор холбогдоход нууц үг шаардахгүй бөгөөд боломжит аюулгүй байдлын цоорхой үүсгэх юм.
4. Өөрчлөлтүүд идэвхтэй болсон эсэхийг харахын тулд системийг дахин ачаална.

Хэрэв өөр тохиргоо шаардлагатай бол дараагийн хэсэгт илүү гүнзгий тохиргооны тайлбар байгаа болно.

## 27.6.3. Цуваа консолын гүнзгий тохиргоо

1. Цуваа кабель бэлдэнэ.

   Танд null-модем кабель эсвэл стандарт цуваа кабель болон null-модем хувиргагч хэрэгтэй болно. Цуваа кабелиудын тухай хэлэлцүүлгийг Кабелиуд болон Портууд хэсгээс үзнэ үү.

2. Гарыг салгана.

   Ихэнх PC системүүд нь Power-On Self-Test (POST) буюу асаах үеийн тестийн үеэр гарыг байгаа эсэхийг шалгадаг бөгөөд гар олдохгүй бол алдаа үүсгэдэг. Зарим машинууд нь гар байхгүйг чангаар мэдээлж залгагдтал нь цааш ачаалдаггүй.

   Хэрэв таны компьютер алдааг мэдээлсэн боловч цааш ачаалах юм бол та ямар нэг тусгай зүйл хийх шаардлагагүй юм.

   Хэрэв таны компьютер гар залгалгүйгээр ачаалахаас татгалзвал энэ алдааг авч үзэлгүйгээр орхихоор болгож та BIOS-оо тохируулах хэрэгтэй болно (хэрэв болдог бол). Үүнийг хэрхэн хийх талаар дэлгэрэнгүйг өөрийн эх хавтангийн гарын авлагаас лавлана уу.

   BIOS-ийн тохиргоон дээр гараа "Not installed" буюу суулгагдаагүй гэж тохируулна. Тэгсэн ч гэсэн та өөрийн гарыг ашиглаж чадах болно. Үүний хийж байгаа зүйл нь BIOS-д асах үед гарыг шалгахгүй гэж хэлж өгч байгаа юм. Таны BIOS гар байхгүй үед гомдоллохгүй байх ёстой. Та энэ тугийг "Not installed" гэж тохируулсан байлаа ч гэсэн гараа залгаатай хэвээр орхиж болох бөгөөд гар ажилласан хэвээр байх болно. Хэрэв дээрх тохиргоо BIOS-д байхгүй бол оронд нь "Halt on Error" тохиргоог хайх хэрэгтэй. Үүнийг "All but Keyboard" гэж эсвэл "No Errors" гэж тохируулах нь ижил үйлчилгээтэй байдаг.

   Хэрэв таны систем PS/2® хулганатай бол та бас өөрийн гарны адил хулганаа бас салгах шаардлагатай байж болох юм. Учир нь юу вэ гэвэл PS/2® хулгана нь гартай цуг зарим тоног төхөөрөмжийг хуваалцдаг бөгөөд хулганаа залгаатай орхих нь гар шалгагчийг гар байсаар байгаа гэж бодоход хүргэн хуурч болох юм.

3. Дүлий терминалыг COM1 уруу залгана (sio0).

   Хэрэв танд дүлий терминал байхгүй бол та хуучин компьютер модемийн програмтай цуг, эсвэл өөр UNIX® хайрцаг дээр цуваа портыг ашиглаж болох юм. Хэрэв танд COM1 (sio0) байхгүй бол түүнтэй болох хэрэгтэй. Одоогоор ачаалах блокуудыг дахин эмхэтгэлгүйгээр ачаалах блокуудын хувьд COM1-с өөр портыг сонгох арга байхгүй. Хэрэв та COM1-г өөр төхөөрөмжид ашиглаж байгаа бол та тэр төхөөрөмжөө түр зуур салгаад FreeBSD эхлүүлж ажиллуулсныхаа дараа шинэ ачаалах блок болон цөмийг суулгах хэрэгтэй болно.

4. Таны цөмийн тохиргооны файлд COM1-ийн (sio0) хувьд тохирох тугууд тохируулагдсан эсэхийг шалгаарай.

   Холбоотой тугууд нь:

   `0x10`

   Энэ нэгжид зориулан консолын дэмжлэгийг идэвхжүүлнэ. Энэ туг тохируулагдаагүй бол бусад тугууд нь орхигддог. Одоогоор хамгийн ихдээ нэг нэгж консолын дэмжлэгтэй байж болдог; Энэ туг тохируулагдсан эхнийхийг

(тохиргооны файлын дарааллаар) эрхэмлэдэг. Энэ тохируулга нь ганцаараа цуваа портыг консол болгодоггүй. Дараах тугийг тохируулах юм уу эсвэл энэ тугтай цуг доор тайлбарласан `-h` тохируулгыг ашиглана.

`0x20`

Доор тайлбарласан `-h` тохируулга байгаа эсэхээс үл хамаарч энэ нэгжийг консол болгоно (өөр илүү эрхтэй консол байхгүй тохиолдолд). `0x20` туг нь `0x10` тугтай цуг ашиглагдах ёстой.

`0x40`

Энэ нэгжийг (`0x10`-тай цуг) нөөцөлж хэвийн хандалтад энэ нэгжийг байхгүй болгоно. Та энэ тугийг цуваа консол маягаар ашиглахыг хүсэж байгаа цуваа портын нэгждээ тохируулах ёсгүй юм. Энэ тугны цорын ганц хэрэглээ нь алсаас цөмийг дибаг хийхэд нэгжийг зориулах явдал юм. Алсаас дибаг хийх талаар илүү мэдээллийг Хөгжүүлэгчийн Гарын Авлагаас үзнэ үү.

Жишээ:

```
device sio0 flags 0x10
```

Илүү дэлгэрэнгүйг sio(4) гарын авлагын хуудаснаас үзнэ үү.

Хэрэв тугнууд тохируулагдаагүй бол UserConfig-г (өөр консол дээр) ажиллуулах юм уу эсвэл цөмийг дахин эмхэтгэх хэрэгтэй.

5. Ачаалах төхөөрөмж дээр `a` хуваалтын root санд boot.config файлыг үүсгэнэ.

Энэ файл нь системийг та хэрхэн ачаалахыг хүсэж байгааг ачаалах блокийн кодод зааварлах болно. Цуваа консолыг идэвхжүүлэхийн тулд танд дараах нэг буюу хэд хэдэн тохируулгууд хэрэгтэй болно-хэрэв та олон тохируулгуудыг хүсэж байгаа бол бүгдийг нэг мөрт оруулах хэрэгтэй:

`-h`

Дотоод болон цуваа консолуудыг харилцан сольно. Та үүнийг ашиглан консол төхөөрөмжүүдийг сольж шилжүүлж болно. Жишээ нь хэрэв та дотоод (видео) консолоос ачаалах бол ачаалагч дуудагч болон цөмд өөрийн консолын төхөөрөмждөө цуваа портыг ашиглуулахаар чиглүүлэх `-h` тохируулгыг ашиглаж болох юм. Мөн өөрөөр хэрэв та цуваа портоос ачаалах бол ачаалагч дуудагч болон цөмд видео дэлгэцийг консол маягаар харин хэрэглүүлэхийн тулд `-h` тохируулгыг ашиглаж болох юм.

`-D`

Ганц болон хоёр консолын тохиргоонуудыг сольно. Ганц тохиргоон дээр дээрх `-h` тохиргооны төлвөөс хамаарч консол нь дотоод консол (видео дэлгэц) байх юм уу эсвэл цуваа порт байх болно. Хоёр консолын тохиргоон дээр видео дэлгэц болон цуваа порт нь хоёулаа `-h` тохиргооны төлвөөс үл хамааран консол болох юм. Гэхдээ хоёр консолын тохиргоо нь зөвхөн ачаалах блок ажиллаж байх үед идэвхтэй байхыг санаарай. Ачаалагч дуудагч хяналтыг авсны дараа `-h`

тохируулгаар заагдсан консол нь цорын ганц консол болох юм.

`-P`

Ачаалах блокоор гарыг шалгуулна. Хэрэв гар олдоогүй бол `-D` болон `-h` тохируулгууд автоматаар тохируулагдана.

Зайны шаардлагаас болоод ачаалах блокуудын одоогийн хувилбарт `-P` тохируулга нь зөвхөн өргөтгөсөн гарнуудыг олох чадвартай байдаг. 101 товчлуураас бага товчлууруудтай (ба F11 болон F12 товчлууруудгүй) гарнууд нь олдохгүй байж болох юм. Зарим нэг зөөврийн компьютерууд дээрх гарнууд нь энэ хязгаарлалтаас болоод зөвөөр олдохгүй байж болох юм. Хэрэв та ийм системтэй бол `-P` тохируулга ашиглахаа болих хэрэгтэй юм. Харамсалтай нь энэ асуудлыг тойрон гарах арга зам байдаггүй.

Консолио автоматаар сонгохын тулд `-P` тохируулгыг ашиглах юм уу эсвэл цуваа консолыг идэвхжүүлэхийн тулд `-h` тохируулгыг ашиглана.

Та boot(8)-д тайлбарласны адил бас бусад тохируулгуудыг оруулж болох юм.

`-P`-с бусад тохируулгууд нь ачаалагч дуудагчид (/boot/loader) дамжуулагддаг. Ачаалагч дуудагч нь зөвхөн `-h` тохируулгын төлөвийг шалгаж дотоод видео эсвэл цуваа портын аль нь консол болох ёстойг тодорхойлох болно. Энэ нь хэрэв та /boot.config файлд `-D` тохируулгыг тохируулаад гэхдээ `-h` тохируулгыг тохируулаагүй бол та цуваа портыг зөвхөн ачаалах блокийн үед консол болгон ашиглаж болно гэсэн үг юм; ачаалагч дуудагч нь дотоод видео дэлгэцийг консол маягаар ашиглах болно.

6. Машиныг ачаална.

Та өөрийн FreeBSD хайрцгийг эхлүүлэх үед ачаалах блокууд нь /boot.config файлын агуулгуудыг консол уруу гаргаж харуулах болно. Жишээ нь:

```
/boot.config: -P
Keyboard: no
```

Хоёр дахь мөр нь /boot.config файлд `-P` тохируулгыг зөвхөн тавьсан тохиолдолд гарах бөгөөд гар байгаа/байхгүйг заадаг. Эдгээр мэдэгдлүүд нь /boot.config файл дахь тохируулгаас хамаараад цуваа эсвэл дотоод консол уруу, эсвэл хоёулан уруу гардаг.

| **Тохируулгууд** | **Мэдэгдлүүд гарах газар** |
|---|---|
| байхгүй | дотоод консол |
| `-h` | цуваа консол |
| `-D` | цуваа болон дотоод консолууд |
| `-Dh` | цуваа болон дотоод консолууд |
| `-P`, гар байна | дотоод консол |

| Тохируулгууд | Мэдэгдлүүд гарах газар |
| --- | --- |
| `-P`, гар байхгүй | цуваа консол |

Дээрх мэдэгдлүүдийн дараа ачаалах блокууд нь ачаалагч дуудагчийг дуудахаас өмнө болон бусад дараа дараагийн мэдэгдлүүд консол дээр хэвлэгдэхээс өмнө бага хэмжээний завсарлага (пауз) авах болно. Хэвийн нөхцөлд танд ачаалах блокуудыг тасалдуулах хэрэггүй боловч та бүх зүйлс зөв тохируулагдсаныг баталгаажуулахын тулд ингэхийг хүсэж болох юм.

Ачаалах процессийг тасалдуулахын тулд консол дээр `Enter`-с өөр дурын товчлуурыг дарах хэрэгтэй. Ачаалах блокууд нь дараа нь танаас дараачийн чинь үйлдлийг асуух болно:

```
>> FreeBSD/i386 BOOT
Default: 0:ad(0,a)/boot/loader
boot:
```

Дээрх мэдэгдэл цуваа эсвэл дотоод консол дээр эсвэл хоёулан дээр гарч байгаа эсэхийг /boot.config файлд хийсэн тохируулгуудынхаа дагуу шалгаарай. Хэрэв мэдэгдэл зөв консол дээр гарч байвал ачаалах процессийг үргэлжлүүлэхийн тулд `Enter` дараарай.

Хэрэв та цуваа консолыг хүсэж байгаа боловч цуваа терминал дээр хүлээх мөрийг харахгүй байгаа бол таны тохиргоонд ямар нэг зүйл буруу байна. Тэр хооронд та ачаалах блокт (ба дараа нь ачаалагч дуудагч болон цөмд) цуваа портыг консолд зориулан сонгуулахыг хэлэхийн тулд `-h` гэж бичээд `Enter` эсвэл `Return` (боломжтой бол) дараарай. Систем ачаалсны дараа юу буруу болсныг буцаж шалгаарай.

Ачаалагч дуудагч дуудагдсаны дараа та ачаалах процессийн гурав дахь шатанд байгаа бол ачаалагч дуудагчид зохих орчны хувьсагчуудыг тохируулж дотоод консол болон цуваа консолын хооронд та шилжиж чадсан хэвээр байх болно. Ачаалагч дуудагчаас консолыг өөрчлөх-г үзнэ үү.

## 27.6.4. Дүгнэлт

Энэ хэсэгт хэлэлцсэн төрөл бүрийн тохиргоонуудын дүгнэлт болон эцсийн дүнд консол сонгогдсон тухай мэдээлэл энд байна.

### 27.6.4.1. Тохиолдол 1: Та sio0-д зориулж тугуудыг 0x10 болгож тохируулна

```
device sio0 flags 0x10
```

| /boot.config дахь тохируулгууд | Ачаалах блокуудын үеийн консол | Ачаалагч дуудагчийн үеийн консол | Цөм дэх консол |
|---|---|---|---|
| юу ч байхгүй | дотоод | дотоод | дотоод |
| -h | цуваа | цуваа | цуваа |
| -D | цуваа болон дотоод | дотоод | дотоод |
| -Dh | цуваа болон дотоод | цуваа | цуваа |
| -P, гар байна | дотоод | дотоод | дотоод |
| -P, гар байхгүй | цуваа болон дотоод | цуваа | цуваа |

#### 27.6.4.2. Тохиолдол 2: Та sio0-д зориулж тугуудыг 0x30 болгож тохируулна

```
device sio0 flags 0x30
```

| /boot.config дахь тохируулгууд | Ачаалах блокуудын үеийн консол | Ачаалагч дуудагчийн үеийн консол | Цөм дэх консол |
|---|---|---|---|
| юу ч байхгүй | дотоод | дотоод | цуваа |
| -h | цуваа | цуваа | цуваа |
| -D | цуваа болон дотоод | дотоод | цуваа |
| -Dh | цуваа болон дотоод | цуваа | цуваа |
| -P, гар байна | дотоод | дотоод | цуваа |
| -P, гар байхгүй | цуваа болон дотоод | цуваа | цуваа |

### 27.6.5. Цуваа консолд зориулсан зааврууд

#### 27.6.5.1. Цуваа портын хурдыг илүү хурдан болгох

Анхдагчаар цуваа портын тохируулгууд нь 9600 baud, 8 бит, parity байхгүй, ба 1 зогсох бит бүхий байна. Хэрэв та анхдагч консолын хурдыг өөрчлөхийг хүсвэл танд дараах сонголтууд байна:

- Ачаалах блокуудыг консолын шинэ хурд заагдсан `BOOT_COMCONSOLE_SPEED` хувьсагчтай цуг дахин эмхэтгэнэ. Шинэ ачаалах блокуудыг бүтээж суулгах талаар дэлгэрэнгүй зааврууд Консолд зориулж sio0-с өөр цуваа портыг ашиглах-с үзнэ үү.

  Хэрэв цуваа консол нь `-h` тохируулгатайгаар ачаалахаас өөр аргаар тохируулагдсан бол эсвэл цөмийн ашиглаж байгаа цуваа консол нь ачаалах блокуудын ашиглаж байгаагаас өөр бол та дараах тохируулгыг цөмийн тохиргооны файлдаа бас нэмж шинэ цөмөө эмхэтгэх ёстой юм:

```
options CONSPEED=19200
```

- Цөмийн `-S` ачаалах тохируулгыг ашиглана. `-S` тушаалын мөрийн тохируулга нь /boot.config файлд нэмэгдэж болно. /boot.config файлд хэрхэн тохируулгууд нэмэх тухай тайлбар болон дэмжигдсэн тохируулгуудын талаар boot(8) гарын авлагын хуудаснаас үзнэ үү.
- Өөрийн /boot/loader.conf файлд `comconsole_speed` тохируулгыг идэвхжүүлнэ. Энэ тохируулга нь /boot/loader.conf файлд бас `console`, `boot_serial`, болон `boot_multicons` тохируулагдсан эсэхээс хамаардаг. Цуваа консолын хурдыг өөрчлөх `comconsole_speed` тохируулгын хэрэглээний жишээ энд байна:

```
boot_multicons="YES"
boot_serial="YES"
comconsole_speed="115200"
console="comconsole,vidconsole"
```

#### 27.6.5.2. Консолд зориулж sio0-с өөр цуваа портыг ашиглах

sio0-с өөр портыг консолд ашиглах нь зарим нэг дахин эмхэтгэлтийг шаарддаг. Та ямар ч шалтгаанаар байлаа гэсэн өөр цуваа портыг ашиглахыг хүсвэл доор дурдсан шигээр ачаалах блокууд, ачаалагч дуудагч болон цөмийг дахин эмхэтгэх хэрэгтэй.

1. Цөмийн эхийг авна. (FreeBSD-г шинэчилж сайжруулах нь бүлгийг үзнэ үү)
2. /etc/make.conf файлыг засварлаж `BOOT_COMCONSOLE_PORT` хувьсагчийг ашиглахыг хүсэж байгаа портынхоо хаягаар (0x3F8, 0x2F8, 0x3E8 эсвэл 0x2E8) тохируулна. Зөвхөн sio0-с sio3 (COM1-с COM4) хүртэлх портууд ашиглагдаж болно; олон порттой цуваа картууд ажиллахгүй. Тасалдлын тохируулга шаардлагагүй.
3. Тусгайлсан цөмийн тохиргооны файл үүсгээд ашиглахыг хүсэж байгаа портынхоо хувьд тохирох тугуудыг нэмээрэй. Жишээ нь хэрэв та sio1 (COM2) портыг консол болгохыг хүсэж байгаа бол:

   ```
   device sio1 flags 0x10
   ```

   эсвэл

   ```
   device sio1 flags 0x30
   ```

   Бусад цуваа портуудын хувьд консолын тугууд тохируулагдах ёсгүй.
4. Ачаалах блокууд болон ачаалагч дуудагчийг дахин эмхэтгээд суулгана:

   ```
   # cd /sys/boot
   # make clean
   ```

```
# make
# make install
```

5. Цөмийг дахин бүтээж суулгана.
6. Ачаалах блокуудыг ачаалах диск уруу bsdlabel(8)-ийн тусламжтай бичээд шинэ цөмөөс ачаална.

#### 27.6.5.3. Цуваа шугамнаас DDB дибаг хийгч уруу орох

Хэрэв та цуваа консолоос цөмийг дибаг хийгч уруу орохыг хүсвэл (алсаас хийх шинжилгээнд ашиг тустай байдаг боловч хэрэв та цуваа порт дээр алдаатай BREAK үүсгэсэн бол энэ нь аюултай байдаг!) та өөрийн цөмийг дараах тохируулгуудтай эмхэтгэх шаардлагатай:

```
options BREAK_TO_DEBUGGER
options DDB
```

#### 27.6.5.4. Цуваа консол дээр нэвтрэлт хүлээх мөр гаргаж авах

Энэ нь шаардлагатай биш боловч та одоо цуваа шугамаар ачаалалтын мэдэгдлүүдийг харж цөмийн дибаг хийх сесс уруу орж чадаж байгаа юм чинь та цуваа шугамаар *нэвтрэлт* хүлээх мөрийг авахыг хүсэж болох юм. Энд хэрхэн үүнийг хийх талаар байна.

/etc/ttys файлыг засварлагчаар онгойлгоод дараах мөрүүдийг олно:

```
ttyu0 "/usr/libexec/getty std.9600" unknown off secure
ttyu1 "/usr/libexec/getty std.9600" unknown off secure
ttyu2 "/usr/libexec/getty std.9600" unknown off secure
ttyu3 "/usr/libexec/getty std.9600" unknown off secure
```

ttyu0-с ttyu3 хүртэлх нь COM1-с COM4 хүртэлхтэй тохирно. Хүссэн порт дээрээ `off` гэснийг `on` болгож өөрчилнө. Хэрэв та цуваа портын хурдыг өөрчилсөн бол тухайн үеийн тохиргоотой тааруулахын тулд `std.9600`-г өөрчлөх хэрэгтэй болно, өөрөөр хэлбэл `std.19200` гэх мэт.

Та бас терминалын төрлийг `unknown` гэдгээс өөрийн цуваа терминалын жинхэнэ төрлөөр солихыг хүсэж болох юм.

Файлыг засварласны дараа та энэ өөрчлөлтийг идэвхжүүлэхийн тулд `kill -HUP 1` хийх ёстой.

### 27.6.6. Ачаалагч дуудагчаас консолыг өөрчлөх

Өмнөх хэсгүүд нь ачаалах блокуудыг тохируулан цуваа консолыг хэрхэн тохируулах талаар тайлбарлалаа. Энэ хэсэг нь ачаалагч дуудагч дээр зарим нэг тушаалууд болон орчны хувьсагчуудыг оруулж консолыг тохируулах талаар үзүүлэх болно. Ачаалах блокийн дараа ачаалах процессийн гурав дахь шатан дээр ачаалагч дуудагч ажиллах үед ачаалагч дуудагч дахь тохиргоонууд нь ачаалах блок дахь тохиргоонуудыг дарж ажиллах болно.

#### 27.6.6.1. Цуваа консол тохируулах

Та ачаалагч дуудагч болон цөмд цуваа консолыг ашиглуулахын тулд /boot/loader.conf файлд зөвхөн нэг мөр бичээд амархан зааж өгч болно:

```
console="comconsole"
```

Энэ нь өмнөх хэсэгт хэлэлцсэн ачаалах блок дахь тохиргоонуудаас үл хамааран идэвхжих болно.

Та цуваа консол дээр ачаалалтын мэдэгдлүүдийг аль болох эрт харахын тулд дээрх мөрийг /boot/loader.conf файлынхаа хамгийн эхний мөр болгон оруулах нь зүйтэй юм.

Үүнтэй адилаар та дотоод консолоо ингэж зааж өгч болно:

```
console="vidconsole"
```

Хэрэв та ачаалагч дуудагчийн орчны хувьсагч `console`-г тохируулахгүй бол ачаалагч болон дараа нь цөм ачаалах блок дахь `-h` тохируулгаар заагдсан тэр консолыг ашиглах болно.

Консолыг /boot/loader.conf.local юм уу эсвэл /boot/loader.conf файлд зааж өгч болно.

Дэлгэрэнгүй мэдээллийг loader.conf(5)-с үзнэ үү.

Одоогоор ачаалагч дуудагчид ачаалах блок дахь `-P` тохируулгатай адил тохируулга байдаггүй бөгөөд гар байгаа эсэхээс хамаарч дотоод консол ба цуваа консолыг автоматаар сонгох бэлтгэл байдаггүй юм.

#### 27.6.6.2. Консолд зориулж sio0-с өөр цуваа портыг ашиглах

Та цуваа консолд зориулж sio0-с өөр цуваа портыг ашиглахын тулд ачаалагч дуудагчийг дахин эмхэтгэх хэрэгтэй. Консолд зориулж sio0-с өөр цуваа портыг ашиглах хэсэгт тайлбарласан процедурыг дагаарай.

### 27.6.7. Анхаарах зүйлс

Гол санаа нь график тоног төхөөрөмж эсвэл гар залгагдсан байхыг шаарддаггүй тийм зориулагдсан серверүүдийг тохируулах боломжийг хүмүүст зөвшөөрөх явдал юм. AMI BIOS-уудтай машинуудыг график хувиргагч суулгалгүйгээр ачаалдаг болгож тохируулж болдог. Ингэхийн тулд ердөө л CMOS тохиргоон дахь "graphics adapter" тохируулгыг "Not installed" болгож өөрчилнө.

Гэхдээ олон машинууд энэ тохируулгыг дэмждэггүй бөгөөд та систем дээрээ ямар ч дэлгэцийн тоног төхөөрөмж байхгүй бол машин ачаалахаас татгалзах болно. Ийм машинуудад та ямар нэг төрлийн график карт (хогийн хар цагаан хавтан байсан ч гэсэн болно) залгаж орхих хэрэгтэй болох боловч заавал дэлгэц залгах шаардлагагүй юм. Та бас AMI BIOS суулгахыг оролдож болох юм.

# Chapter 28. PPP болон SLIP

## 28.1. Ерөнхий агуулга

FreeBSD нь нэг компьютерийг нөгөөтэй холбох хэд хэдэн аргуудтай байдаг. Dial-up модемоор сүлжээнд эсвэл Интернэтэд холболт хийх юм уу эсвэл бусдыг өөрөөр чинь дамжихыг зөвшөөрөхийн тулд PPP эсвэл SLIP-г хэрэглэхийг шаарддаг. Энэ бүлэг эдгээр модем дээр тулгуурласан холбооны үйлчилгээнүүдийг тохируулах талаар дэлгэрэнгүй тайлбарлах болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Хэрэглэгчийн PPP-г хэрхэн тохируулах талаар.
- Цөмийн PPP-г хэрхэн тохируулах талаар (зөвхөн FreeBSD 7.X-ийн хувьд).
- PPPoE-г (PPP over Ethernet буюу Ethernet дээгүүрх PPP) хэрхэн тохируулах талаар.
- PPPoA-г (PPP over ATM буюу ATM дээгүүрх PPP) хэрхэн тохируулах талаар.
- SLIP клиент болон серверийг хэрхэн тохируулж суулгах талаар (зөвхөн FreeBSD 7.X-ийн хувьд).

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- Сүлжээний үндсэн ухагдахуунуудыг мэддэг байх.
- Гадагш залгах (dialup) холболт болон PPP ба/эсвэл SLIP-ийн үндэс болон зорилгыг ойлгосон байх.

Та хэрэглэгчийн PPP болон цөмийн PPP хоёрын хоорондох гол ялгаа нь юу юм бол гэж гайхаж байж магадгүй юм. Хариулт нь хялбархан: хэрэглэгчийн PPP нь ирж байгаа болон гарч байгаа өгөгдлийг цөмийн талбарт биш хэрэглэгчийн талбарт процесс хийдэг. Энэ нь өгөгдлийг цөм болон хэрэглэгчийн талбар хоёрын хооронд хуулдгаараа зардалтай боловч хамаагүй илүү боломжуудаар баялаг PPP шийдлийг зөвшөөрдөг. Хэрэглэгчийн PPP нь гаднах ертөнцтэй холбогдохдоо tun төхөөрөмжийг ашигладаг бол цөмийн PPP ppp төхөөрөмжийг ашигладаг.

Энэ бүлэгт хэрэглэгчийн PPP-г pppd (зөвхөн FreeBSD 7.X-ийн хувьд) зэрэг өөр бусад PPP програм хангамжаас ялгах шаардлага гарахгүй тохиолдолд ердөө л ppp гэх болно. Тусгайлан тайлбарлаагүй л бол энэ бүлэгт тайлбарласан бүх тушаалууд нь `root` эрхээр ажиллуулагдах ёстой.

## 28.2. Хэрэглэгчийн PPP ашиглах

### 28.2.1. Хэрэглэгчийн PPP

#### 28.2.1.1. Таамаглалууд

Энэ баримт нь таныг дараах шаардлагуудыг хангасан гэж тооцдог:

- PPP ашиглан таны холбогдох Интернэтийн үйлчилгээ үзүүлэгчээс (ISP) өгсөн бүртгэл.
- Модем эсвэл өөр төхөөрөмж таны системд холбогдсон байгаа бөгөөд тэр нь танд өөрийн ISP-тай холбогдохыг зөвшөөрөхөөр зөв тохируулагдсан байгаа.
- Таны ISP-ийн хандах dial-up дугаар(ууд).
- Таны нэвтрэх нэр болон нууц үг. (ердийн UNIX® загварын нэвтрэлт болон нууц үгийн хослол юм уу эсвэл PAP эсвэл CHAP нэвтрэлт болон нууц үгийн хослол).
- Нэг буюу хэд хэдэн нэрийн серверүүдийн IP хаяг. Хэвийн үед үүнд зориулан ашиглахаар хоёр IP хаягийг таны ISP танд өгдөг. Хэрэв тэд танд ядаж ганцыг өгөөгүй бол та `enable dns` тушаалыг ppp.conf файлд ашиглавал ppp нэрийн серверүүдийг танд тохируулж өгөх болно. Энэ боломж нь DNS-тай тохиролцоог дэмждэг таны ISP-ийн PPP шийдлээс хамаарах юм.

Дараах мэдээллийг таны ISP өгсөн байж болох боловч энэ нь заавал шаардлагагүй юм:

- Таны ISP-ийн гарцын IP хаяг. Гарц нь таны холбогдох машин бөгөөд таны *анхдагч чиглүүлэлт* гэж тохируулагдах болно. Хэрэв танд энэ мэдээлэл байхгүй бол бид нэгийг орлуулж болох бөгөөд таны ISP-ийн PPP сервер биднийг холбогдох үед зөв утгыг хэлж өгөх болно.

  Энэ IP дугаар нь ppp-ээр `HISADDR` гэгдэнэ.

- Таны ашиглах ёстой сүлжээний баг. Хэрэв таны ISP үүнийг өгөөгүй бол та аюулгүйгээр `255.255.255.255` гэж ашиглаж болно.
- Хэрэв таны ISP танд статик IP хаяг болон хостын нэр өгсөн бол та тэдгээрийг оруулж болно. Үгүй бол бид нөгөө талын өгөх тохирох IP хаягийг ердөө л зөвшөөрөх болно.

Хэрэв танд шаардлагатай мэдээллийн аль нь ч байхгүй бол өөрийн ISP уруугаа хандана уу.

Энэ хэсэгт жишээнүүдийн олонхийн үзүүлж байгаа тохиргооны файлуудын агуулгуудад байгаа мөр бүр дугаарлагдсан байгаа болно. Эдгээр дугаарууд нь танилцуулга болон хэлэлцүүлэгт туслах зорилгоор зөвхөн ашиглагддаг бөгөөд яг үнэндээ жинхэнэ файл дээрээ тавигдах ёстой гэсэн үг биш юм. Tab ашиглан зөв догол хийх болон зайн тэмдэгтүүд нь бас чухал юм.

#### 28.2.1.2. Автомат PPP тохиргоо

`ppp` болон `pppd` нь (PPP-ийн цөмийн түвшний шийдэл, зөвхөн FreeBSD 7.X-ийн хувьд) /etc/ppp сан дахь тохиргооны файлуудыг ашигладаг. Хэрэглэгчийн ppp-д зориулсан жишээнүүдийг /usr/shared/examples/ppp/ сангаас олж болно.

`ppp`-г тохируулах нь таны өөрийн шаардлагуудаас хамаарч хэд хэдэн файлуудыг засварлахыг танаас шаарддаг. Тэдгээрт юу оруулах нь таны ISP IP хаягуудыг статикаар өгдөг үү (өөрөөр хэлбэл танд нэг IP хаяг өгөх бөгөөд зөвхөн тэрийг дандаа ашиглана) эсвэл динамикаар (өөрөөр хэлбэл таны IP хаяг таныг ISP-даа холбогдох бүрт өөрчлөгддөг) өгдөг үү гэдгээс зарим талаараа хамаардаг.

#### 28.2.1.2.1. PPP болон статик IP хаягууд

Та /etc/ppp/ppp.conf тохиргооны файлыг засварлах хэрэгтэй. Энэ нь доорх жишээтэй төстэй харагдах ёстой.

:-ээр төгссөн мөрүүд эхний баганаас эхэлнэ (мөрийн эхлэл)- бусад бүх мөрүүдэд доор үзүүлсэн шиг зай эсвэл tab-аар догол гаргасан байх ёстой.

```
1     default:
2       set log Phase Chat LCP IPCP CCP tun command
3       ident user-ppp VERSION (built COMPILATIONDATE)
4       set device /dev/cuau0
5       set speed 115200
6       set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \
7                 \"\" AT OK-AT-OK ATE1Q0 OK \\dATDT\\T TIMEOUT 40 CONNECT"
8       set timeout 180
9       enable dns
10
11    provider:
12      set phone "(123) 456 7890"
13      set authname foo
14      set authkey bar
15      set login "TIMEOUT 10 \"\" \"\" gin:--gin: \\U word: \\P col: ppp"
16      set timeout 300
17      set ifaddr x.x.x.x y.y.y.y 255.255.255.255 0.0.0.0
18      add default HISADDR
```

**Мөр 1**

Анхдагч оруулгыг тэмдэглэдэг. Энэ оруулга дахь тушаалууд нь ppp ажиллах үед автоматаар ажилладаг.

**Мөр 2**

Параметрүүдийг бүртгэл хийхийг идэвхжүүлнэ. Тохиргоо хангалттайгаар ажиллаж байгаа үед бүртгэлийн файлын их хэмжээнээс сэргийлэхийн тулд энэ мөрийг:

```
set log phase tun
```

гэж болгох ёстой.

**Мөр 3**

PPP-д нөгөө талдаа өөрийгөө хэрхэн таниулахыг хэлж өгдөг. PPP нь тохиролцож холбоосыг үүсгэх үедээ ямар нэгэн асуудалтай байгаа эсэхээ таниулсан мэдээллээр нөгөө талаа хангадаг бөгөөд нөгөө талын администратор иймэрхүү асуудлуудыг судалж байгаа үед энэ нь ашигтай байж болох юм.

**Мөр 4**

Модемийн холбогдсон төхөөрөмжийг тодорхойлдог. COM1 нь /dev/cuau0 бөгөөд COM2 нь

/dev/cuau1 юм.

**Мөр 5**

Таны холбогдох хурдыг заадаг. Хэрэв 115200 ажиллахгүй бол (ямар ч шинэ модемтой энэ нь болох ёстой) 38400-гаар оролдож үзээрэй.

**Мөр 6 & 7**

Залгах мөр. Хэрэглэгчийн PPP нь chat(8) програмын адил хүлээгээд илгээх зарчмыг ашигладаг. Энэ хэлний боломжуудын тухай мэдээллийг гарын авлагын хуудаснаас лавлана уу.

Энэ тушаал нь уншигдахад хялбар байх зорилгоор дараагийн мөрөнд үргэлжилж байгааг анзаараарай. Хэрэв мөрийн сүүлийн тэмдэгт \ байвал ppp.conf файл дахь ямар ч тушаал ийм байж болох юм.

**Мөр 8**

Энэ холбоосонд зориулж сул зогсох хугацааг тохируулна. 180 секунд нь анхдагч байна, тэгэхээр энэ мөр нь цэвэр гоо сайхны зүйл юм.

**Мөр 9**

Локал танигчийн тохиргоог бататгахын тулд нөгөө талаасаа асуухыг PPP-д хэлнэ. Хэрэв та локал нэрийн серверийг ажиллуулах бол энэ мөрийг тайлбар болгох юм уу эсвэл арилгах ёстой.

**Мөр 10**

Уншихад хялбар байх зорилгоор хоосон мөр байна. PPP нь хоосон мөрүүдийг орхидог.

**Мөр 11**

"provider" гэж нэрлэгдсэн үзүүлэгчид зориулсан оруулгыг тодорхойлдог. Үүнийг өөрийн ISP-ийн нэрээр сольж болох юм. Ингэсний дараа та холболтыг эхлүүлэхийн тулд `load ISP` гэж ашиглаж болох юм.

**Мөр 12**

Энэ үзүүлэгчийн хувьд утасны дугаарыг тохируулдаг. Олон утасны дугаарыг тодорхойлох хоёр цэг (:) юм уу эсвэл хоолой тэмдэгтийг (|) тусгаарлагч болгон ашиглан зааж өгч болно. Хоёр тусгаарлагчийн ялгаа ppp(8)-д тайлбарлагдсан байдаг. Дүгнэж хэлэхэд хэрэв та дугааруудыг ээлжлэн ашиглах бол тодорхойлох цэгийг ашиглана. Хэрэв та эхний дугаар уруу эхлээд үргэлж залгахыг хүсэж байгаа бөгөөд зөвхөн эхний дугаар нь амжилтгүй болсон тохиолдолд бусад дугааруудыг ашиглахыг хүсэж байгаа бол хоолой тэмдэгтийг ашиглаарай. Утасны дугааруудыг тэр чигээр нь үзүүлсэн шиг үргэлж хаалтанд хийх хэрэгтэй.

Хэрэв та утасны дугаар дээр зай ашиглахаар бол утасны дугаарыг хаалтанд (") хийх ёстой. Ингэхгүй бол энэ нь энгийн боловч баригдашгүй алдаанд хүргэж болох юм.

**Мөр 13 & 14**

Хэрэглэгчийн нэр болон нууц үгийг тодорхойлно. UNIX® загварын нэвтрэлт хүлээх мөрийг ашиглан холбогдох үед эдгээр уттууд нь `set login` тушаалаар \U болон \P

хувьсагчуудыг ашиглан хийгддэг. PAP эсвэл CHAP ашиглан холбогдож байгаа үед эдгээр утгууд нь нэвтрэлт танилт хийгдэхэд хэрэглэгддэг.

**Мөр 15**

Хэрэв та PAP эсвэл CHAP ашиглаж байгаа бол нэвтрэлт энэ үед байхгүй байх бөгөөд энэ мөр нь тайлбар болгогдох юм уу эсвэл арилгагдсан байх ёстой. Илүү дэлгэрэнгүй мэдээллийг PAP болон CHAP нэвтрэлт танилт холбоосоос үзнэ үү.

Нэвтрэлтийн мөр нь залгах мөр шиг чалчихтай төсөөтэй зарчимтай байдаг. Энэ жишээн дээр нэвтрэх сесс нь доор дурдсантай адил үйлчилгээний хувьд уг мөр ажиллаж байна:

```
J. Random Provider
login: foo
password: bar
protocol: ppp
```

Та энэ скриптийг өөрийн хэрэгцээндээ тааруулан өөрчлөх хэрэгтэй болно. Энэ скриптийг эхний удаа бичиж байхдаа та яриа хүссэнээр үргэлжилж байгаа эсэхийг тодорхойлж чадахаар байхын тулд "chat" буюу чалчих бүртгэлийг идэвхжүүлсэн эсэхээ баталгаажуулах хэрэгтэй.

**Мөр 16**

Анхдагч сул зогсох хугацааг (секундээр) холболтод зориулж тохируулдаг. Энд холболт нь 300 секундын хугацаанд идэвхгүй байвал автоматаар хаагдах болно. Хэрэв та хугацааны хувьд хэзээ ч дуусахгүй байхыг хүсвэл энэ утгыг тэг болгох юм уу эсвэл `-ddial` тушаалын мөрийн тохируулгыг ашиглах хэрэгтэй.

**Мөр 17**

Интерфэйсийн хаягуудыг тохируулна. *x.x.x.x* мөрийг таны интернэтийн үйлчилгээ үзүүлэгчийн танд хуваарилсан IP хаягаар солих шаардлагатай. *y.y.y.y* мөрийг таны ISP өөрийн гарцыг (таны холбогдох машин) заасан тэр IP хаягаар солих шаардлагатай. Хэрэв таны ISP гарцын хаягийг танд өгөөгүй бол `10.0.0.2/0` гэж ашиглаарай. Хэрэв та "таасан" хаягийг ашиглах хэрэгтэй бол PPP болон динамик IP хаягууд хэсэгт зориулсан зааврууудын дагуу /etc/ppp/ppp.linkup файлдаа оруулга үүсгэсэн эсэхээ шалгаарай. Хэрэв энэ мөрийг орхивол `ppp` нь `-auto` горимд ажиллаж чадахгүй.

**Мөр 18**

Өөрийн ISP-ийн гарц уруу анхдагч чиглүүлэлтийг нэмнэ. `HISADDR` тусгай үгийг мөр 17 дээр заасан гарцын хаягаар сольдог. Энэ мөр нь 17-р мөрийн дараа байх нь чухал бөгөөд тэгэхгүй бол `HISADDR` нь эхлэн тохируулагдаж амжихгүй байх юм.

Хэрэв та ppp-г `-auto` горимд ажиллуулахыг хүсэхгүй бол энэ мөрийг ppp.linkup файлд шилжүүлэх ёстой.

Та статик IP хаягтай бөгөөд ppp-г `-auto` горимоор ажиллуулж байгаа бол ppp.linkup файлд оруулга нэмэх шаардлагагүй бөгөөд учир нь таныг холбогдохоос өмнө таны чиглүүлэлтийн хүснэгтийн оруулгууд нь аль хэдийн зөв байх учраас тэр юм. Гэхдээ та холболтын дараа програмуудыг ажиллуулах оруулга үүсгэхийг хүсэж болох юм. Энэ нь sendmail-ийн жишээн

дээр сүүлд тайлбарлагдах болно.

Жишээ тохиргооны файлууд нь /usr/shared/examples/ppp/ санд байрлана.

#### 28.2.1.2.2. PPP болон динамик IP хаягууд

Хэрэв таны үйлчилгээ үзүүлэгч статик IP хаягуудыг олгодоггүй бол локал болон алсын хаягууд дээр тохирдог байхаар `ppp` нь тохируулагдаж болно. IP хаягийг "таах" болон холболтын дараа IP Configuration Protocol (IPCP) буюу IP тохиргооны протокол ашиглан зөвөөр тохируулахыг `ppp`-д зөвшөөрөх замаар үүнийг хийнэ. ppp.conf тохиргоо нь дараах өөрчлөлтийн хамтаар PPP болон статик IP хаягууд хэсэгтэй адил байна:

```
17      set ifaddr 10.0.0.1/0 10.0.0.2/0 255.255.255.255 0.0.0.0
```

Дахин хэлэхэд мөрийн дугаарыг битгий оруулаарай, энэ нь зөвхөн лавлагааны зорилгоор байгаа юм. Ядаж нэг хоосон зайгаар догол гаргах шаардлагатай.

**Мөр 17**

`/` тэмдэгтийн дараах дугаар нь ppp-ийн шаардах хаягийн битийн тоо юм. Та өөрийн нөхцөлдөө тохируулан IP дугааруудыг ашиглахыг хүсэж болох юм, гэхдээ дээрх жишээнүүд нь үргэлж ажиллах болно.

Сүүлийн нэмэлт өгөгдөл (`0.0.0.0`) нь PPP-д `10.0.0.1`-ийн оронд `0.0.0.0` хаяг ашиглан тохиролцоог эхлүүлэхийг хэлж байгаа бөгөөд зарим нэгэн ISP-уудын хувьд энэ нь шаардлагатай байдаг. Эхний чиглүүлэлтийг `-auto` горимд тохируулахад PPP-д саад болдог учраас `set ifaddr` тушаалд `0.0.0.0`-г эхний нэмэлт өгөгдөл болгон битгий ашиглаарай.

Хэрэв та `-auto` горимд ажиллахгүй байгаа бол /etc/ppp/ppp.linkup файлд оруулга үүсгэх хэрэгтэй болно. ppp.linkup нь холболт тогтсоны дараа ашиглагддаг. Энэ үед `ppp` нь интерфэйсийн хаягуудыг олгосон байх бөгөөд одоо чиглүүлэлтийн хүснэгтийн оруулгуудыг нэмэх боломжтой болсон байх болно:

```
1     provider:
2      add default HISADDR
```

**Мөр 1**

Холболт тогтоохдоо `ppp` нь дараах дүрмүүдийн дагуу ppp.linkup файлд оруулгыг хайх болно: Эхлээд бидний ppp.conf файлд ашигласантай адил хаяг/шошготой таарахыг оролдоно. Хэрэв амжилттүй болвол бидний гарцын IP хаягийн оруулгыг хайна. Энэ оруулга нь дөрвөн ширхэг найман битээс тогтох IP загварын хаяг/шошго юм. Хэрэв бид тэгсэн ч гэсэн оруулга олоогүй байгаа бол `MYADDR` оруулгыг хайна.

**Мөр 2**

Энэ мөр нь `HISADDR` уруу заасан анхдагч чиглүүлэлт нэмэхийг `ppp`-д хэлж байна. `HISADDR` нь IPCP-ээр тохиролцсоны дагуу гарцын IP хаягаар солигдох болно.

Нарийвчилсан жишээний хувьд /usr/shared/examples/ppp/ppp.conf.sample болон

/usr/shared/examples/ppp/ppp.linkup.sample файлууд дахь pmdemand оруулгыг харна уу.

#### 28.2.1.2.3. Ирж байгаа дуудлагуудыг хүлээн авах

LAN-д холбогдсон машин дээр ирж байгаа дуудлагуудыг хүлээн авахаар болгож ppp-г тохируулахдаа LAN уруу пакетуудыг дамжуулахыг хүсэж байгаа эсэхээсээ хамаарч үүнийг шийдэх ёстой. Хэрэв та ингэхийг хүсэж байгаа бол өөрийн LAN-ий дэд сүлжээнээс IP хаягийг нөгөө талдаа хуваарилж өгөх хэрэгтэй бөгөөд өөрийн /etc/ppp/ppp.conf файлд enable proxy тушаалыг ашиглах хэрэгтэй. /etc/rc.conf файл доор дурдсаныг агуулж байгаа эсэхийг бас баталгаажуулах хэрэгтэй:

```
gateway_enable="YES"
```

#### 28.2.1.2.4. Аль getty?

getty(8) ашиглан dial-up буюу гадагш залгах үйлчилгээнүүдийг идэвхжүүлэх тухай сайн тайлбарыг Dial-up буюу гадагш залгах үйлчилгээнүүдэд зориулж FreeBSD-г тохируулах хэсэг өгдөг.

getty-тэй төстэй програм нь dial-up шугамуудыг бодолцож хийсэн getty-ийн илүү ухаалаг хувилбар болох mgetty ( comms/mgetty+sendfax порт) юм.

mgetty-г ашиглахын давуу тал нь тэрээр модемуудтай идэвхтэйгээр *ярилцдаг* явдал бөгөөд энэ нь юу гэсэн үг вэ гэхээр хэрэв таны порт хаагдсан бол модем чинь утсанд хариулахгүй гэсэн үг юм.

mgetty-ийн сүүлийн хувилбарууд нь (0.99beta-аас эхлэн) таны клиентүүдэд скриптгүйгээр таны серверт хандах хандалтыг зөвшөөрч PPP урсгалуудын автомат илрүүлэлтийг бас дэмждэг.

mgetty-ийн тухай дэлгэрэнгүй мэдээллийг Mgetty ба AutoPPP хаягаас лавлана уу.

#### 28.2.1.2.5. PPP зөвшөөрлүүд

ppp тушаалыг root хэрэглэгчээр хэвийн үед ажиллуулах ёстой. Гэхдээ хэрэв та ppp-г доор тайлбарласны адилаар энгийн хэрэглэгчээр серверийн горимд ажиллуулах боломжтой болгохыг хүсэж байгаа бол хэрэглэгчийг ppp ажиллуулах зөвшөөрөлтэй болгож тэдгээрийг /etc/group файлд network бүлэгт нэмэх ёстой.

Та allow тушаалыг ашиглан тохиргооны файлын нэг буюу хэд хэдэн хэсгүүдэд хандах боломжийг бас өгөх шаардлагатай болно:

```
allow users fred mary
```

Хэрэв энэ тушаал нь default хэсэгт хэрэглэгдсэн бол заасан хэрэглэгчдэд бүх зүйл уруу хандах боломжийг олгоно.

#### 28.2.1.2.6. Динамик IP-тай хэрэглэгчдэд зориулсан PPP бүрхүүлүүд

доор дурдсаныг агуулсан /etc/ppp/ppp-shell гэгдэх файлыг үүсгэнэ:

```
#!/bin/sh
IDENT=`echo $0 | sed -e 's/^.*-\(.*\)$/\1/'`
CALLEDAS="$IDENT"
TTY=`tty`

if [ x$IDENT = xdialup ]; then
        IDENT=`basename $TTY`
fi

echo "PPP for $CALLEDAS on $TTY"
echo "Starting PPP for $IDENT"

exec /usr/sbin/ppp -direct $IDENT
```

Энэ скрипт нь ажиллахаар болсон байх ёстой. Одоо энэ скрипт уруу ppp-dialup гэгдсэн симболын холбоосыг дараах тушаалуудыг ашиглан үүсгэх хэрэгтэй:

```
# ln -s ppp-shell /etc/ppp/ppp-dialup
```

Та энэ скриптийг өөрийн бүх dialup хэрэглэгчдийн хувьд *бүрхүүл* маягаар ашиглах ёстой. Энэ нь `pchilds` гэсэн хэрэглэгчийн нэртэй dialup PPP хэрэглэгчид зориулсан /etc/passwd файлд байгаа жишээ юм (нууц үгийн файлыг шууд битгий засаарай, vipw(8)-г ашиглаарай).

```
pchilds:*:1011:300:Peter Childs PPP:/home/ppp:/etc/ppp/ppp-dialup
```

Дараах 0 байт файлуудыг агуулсан бүгд бичих боломжтой /home/ppp санг үүсгэнэ:

```
-r--r--r--   1 root     wheel           0 May 27 02:23 .hushlogin
-r--r--r--   1 root     wheel           0 May 27 02:22 .rhosts
```

Энэ нь /etc/motd файлыг үзүүлэхээс сэргийлдэг.

#### 28.2.1.2.7. Статик IP-тай хэрэглэгчдэд зориулсан PPP бүрхүүлүүд

Дээр дурдсаны адил ppp-shell файл үүсгээд статикаар өгөгдсөн IP-тай бүртгэл бүрийн хувьд ppp-shell уруу симболын холбоос үүсгэнэ.

Жишээ нь хэрэв та `fred`, `sam`, болон `mary` гэсэн гурван dialup хэрэглэгчтэй бөгөөд тэдгээрт зориулж /24 CIDR сүлжээнүүдийг чиглүүлж байгаа бол дараах тушаалыг ажиллуулж болох юм:

```
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-fred
```

```
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-sam
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-mary
```

Эдгээр хэрэглэгчдийн dialup бүртгэл бүр өөрсдийн бүрхүүлийг дээр үүсгэсэн симболын холбоос уруу заасан байх ёстой (жишээ нь `mary`-ийн бүрхүүл /etc/ppp/ppp-mary байх ёстой).

#### 28.2.1.2.8. Динамик IP-тай хэрэглэгчдэд зориулж ppp.conf-г тохируулах

/etc/ppp/ppp.conf файл доор дурдсантай төстэй мөрүүдээс тогтох ёстой:

```
default:
  set debug phase lcp chat
  set timeout 0

ttyu0:
  set ifaddr 203.14.100.1 203.14.100.20 255.255.255.255
  enable proxy

ttyu1:
  set ifaddr 203.14.100.1 203.14.100.21 255.255.255.255
  enable proxy
```

Догол үүсгэх нь чухал юм.

`default:` хэсэг нь хэсэг бүрийн хувьд дуудагддаг. /etc/ttys файлд идэвхжүүлсэн dialup мөр бүрийн хувьд дээрх `ttyu0:`-д зориулсантай төстэй оруулга үүсгэнэ. Динамик хэрэглэгчдэд зориулсан IP хаягийн цөөрмөөс мөр болгон өөр өөр IP хаяг авах ёстой.

#### 28.2.1.2.9. Статик IP-тай хэрэглэгчдэд зориулж ppp.conf-г тохируулах

Дээрх жишээ /usr/shared/examples/ppp/ppp.conf файлын агуулгатай цуг статикаар өгөгдсөн dialup хэрэглэгч бүрийн хувьд зохих хэсэг нэмэх шаардлагатай. Бид өөрсдийн `fred`, `sam`, болон `mary` хэрэглэгчидтэй жишээгээ үргэлжлүүлэх болно.

```
fred:
  set ifaddr 203.14.100.1 203.14.101.1 255.255.255.255

sam:
  set ifaddr 203.14.100.1 203.14.102.1 255.255.255.255

mary:
  set ifaddr 203.14.100.1 203.14.103.1 255.255.255.255
```

/etc/ppp/ppp.linkup файл хэрэв шаардлагатай бол статик IP-тай хэрэглэгч бүрийн хувьд чиглүүлэлтийн мэдээллийг бас агуулсан байх ёстой. Доорх мөр нь клиентийн ppp холбоосоор `203.14.101.0/24` сүлжээнд зориулж чиглүүлэлт нэмэх болно.

```
fred:
  add 203.14.101.0 netmask 255.255.255.0 HISADDR

sam:
  add 203.14.102.0 netmask 255.255.255.0 HISADDR

mary:
  add 203.14.103.0 netmask 255.255.255.0 HISADDR
```

#### 28.2.1.2.10. `mgetty` болон AutoPPP

Анхдагчаар comms/mgetty+sendfax порт `AUTO_PPP` тохируулга идэвхжүүлсэн байдлаар ирдэг нь `mgetty`-д PPP холболтуудын LCP шатыг илрүүлж автоматаар ppp бүрхүүл ажиллуулах боломжийг олгодог. Гэхдээ анхдагч нэвтрэлт/нууц үгийн дараалал болдоггүй болохоор PAP аль эсвэл CHAP ашиглан хэрэглэгчдийн нэвтрэлтийг таних шаардлагатай юм.

Энэ хэсэг нь хэрэглэгч comms/mgetty+sendfax портыг амжилттайгаар эмхэтгэж өөрийн систем дээрээ суулгасан гэж үздэг.

Таны /usr/local/etc/mgetty+sendfax/login.config файл доор дурдсаныг агуулсан эсэхийг шалгаарай:

```
/AutoPPP/ -     -            /etc/ppp/ppp-pap-dialup
```

Энэ нь илрүүлсэн PPP холболтууддаа ppp-pap-dialup скриптийг ажиллуулахыг `mgetty`-д хэлнэ.

доор дурдсаныг агуулсан /etc/ppp/ppp-pap-dialup файлыг үүсгэнэ (файл ажиллах боломжтой байх ёстой):

```
#!/bin/sh
exec /usr/sbin/ppp -direct pap$IDENT
```

/etc/ttys-д идэвхжүүлсэн dialup мөр бүрийн хувьд тохирох оруулгыг /etc/ppp/ppp.conf файлд үүсгэнэ. Энэ нь бидний үүсгэсэн тодорхойлолтуудтай аз жаргалтайгаар цуг байх болно.

```
pap:
  enable pap
  set ifaddr 203.14.100.1 203.14.100.20-203.14.100.40
  enable proxy
```

Энэ аргаар нэвтэрч байгаа хэрэглэгч бүр /etc/ppp/ppp.secret файлд хэрэглэгчийн нэр/нууц үгтэй байх шаардлагатай, эсвэл өөр нэг арга нь /etc/passwd файлаас хэрэглэгчдийг PAP-аар таньж нэвтрүүлэхийн тулд дараах тохируулгыг хийх явдал юм.

```
enable passwdauth
```

Хэрэв та зарим хэрэглэгчдэд статик IP хаяг өгөхийг хүсвэл та хаягийг /etc/ppp/ppp.secret файлд гурав дахь нэмэлт өгөгдөл болгон зааж өгч болно. Жишээнүүдийг /usr/shared/examples/ppp/ppp.secret.sample-с үзнэ үү.

#### 28.2.1.2.11. MS өргөтгөлүүд

DNS болон NetBIOS нэрийн серверийн хаягуудыг шаардлагын дагуу хангахаар PPP-г тохируулах боломжтой байдаг.

PPP хувилбар 1.x дээр эдгээр өргөтгөлүүдийг идэвхжүүлэхийн тулд дараах мөрүүдийг /etc/ppp/ppp.conf файлын тохирох хэсэгт нэмэх ёстой.

```
enable msext
set ns 203.14.100.1 203.14.100.2
set nbns 203.14.100.5
```

PPP хувилбар 2 болон түүнээс хойно гарсан хувилбарын хувьд:

```
accept dns
set dns 203.14.100.1 203.14.100.2
set nbns 203.14.100.5
```

Энэ нь анхдагч болон хоёрдох нэрийн серверийн хаягууд болон NetBIOS нэрийн серверийн хостыг клиентүүдэд хэлнэ.

Хувилбар 2 болон түүнээс дээшхид `set dns` мөрийг орхигдуулсан бол PPP нь /etc/resolv.conf файлд байгаа утгуудыг ашиглах болно.

#### 28.2.1.2.12. PAP болон CHAP нэвтрэлт шалгалт

Таны холболтын нэвтрэлт танилтын хэсгийг PAP юм уу эсвэл CHAP нэвтрэлт танилтын аргуудыг ашиглан хийдэг байхаар зарим ISP-ууд өөрсдийн системийг тохируулдаг. Хэрэв ийм бол таны ISP нь `login:` хүлээх мөрийг таныг холбогдох үед харуулдаггүй бөгөөд PPP нэн даруй ярьж эхлэх болно.

PAP нь CHAP-аас аюулгүй байдлын хувьд дутуу байдаг бөгөөд нууц үг нь PAP-аар цэвэр текст маягаар илгээгддэг боловч цуваа шугамаар зөвхөн дамжуулагддаг болохоор аюулгүй байдал нь нууц үгүүдэд байдаг асуудал шиг биш байдаг. Эвдлэгчдэд зориулсан "сэм чагнах" зай бараг байдаггүй гэсэн үг юм.

PPP болон статик IP хаягууд эсвэл PPP болон динамик IP хаягууд хэсгүүдийн хувьд дараах өөрчлөлтүүдийг хийх ёстой:

```
13      set authname MyUserName
14      set authkey MyPassword
15      set login
```

**Мөр 13**

Энэ мөр нь таны PAP/CHAP хэрэглэгчийн нэрийг заана. Та *MyUserName*-д зөв утгыг оруулах хэрэгтэй болно.

**Мөр 14**

Энэ мөр нь таны PAP/CHAP нууц үгийг заана. Та *MyPassword*-д зөв утгыг оруулах хэрэгтэй болно. Та дараах шиг нэмэлт мөр нэмэхийг хүсэх юм уу:

```
16      accept PAP
```

эсвэл

```
16      accept CHAP
```

гэж болно. Ингэснээр үүнийг зориуд тодорхой болгох юм, гэхдээ PAP болон CHAP-ийг хоюуланг анхдагчаар хүлээн авдаг.

**Мөр 15**

Хэрэв та PAP эсвэл CHAP ашиглаж байгаа бол таны ISP хэвийн үед сервер уруу нэвтрэн орохыг шаарддаггүй. Тийм болохоор та өөрийн "set login" мөрийг хаах ёстой.

##### 28.2.1.2.13. Өөрийн ppp тохиргоог явцын дунд шууд өөрчлөх нь

ppp-г ар талд далд ажиллаж байхад түүнтэй зөвхөн тохирох оношлогооны порт тохируулагдсан тохиолдолд харилцаж болдог. Үүнийг хийхийн тулд өөрийн тохиргоондоо дараах мөрийг нэмнэ:

```
set server /var/run/ppp-tun%d DiagnosticPassword 0177
```

Энэ нь клиентүүдийн хандалтыг зөвшөөрөхийн өмнө тэднээс нууц үгийг асууж заасан UNIX® домэйн сокет дээр сонсохыг PPP-д хэлнэ. Нэрэн дэх %d нь ашиглагдаж байгаа tun төхөөрөмжийн дугаараар солигдоно.

Сокет тохируулагдсаны дараа pppctl(8) програм нь ажиллаж байгаа програмыг удирдахыг хүсэж байгаа скриптүүдэд ашиглагдаж болно.

#### 28.2.1.3. PPP-ийн Network Address Translation буюу Сүлжээний Хаягийн Хөрвүүлэлт боломжийг ашиглах нь

PPP нь цөмийн divert буюу өөрчлөн дамжуулах боломжуудыг ашиглалгүйгээр дотоод NAT-ийг ашиглах чадвартай байдаг. /etc/ppp/ppp.conf файл дахь дараах мөрөөр энэ боломжийг идэвхжүүлж болно:

```
nat enable yes
```

Өөрөөр, PPP NAT-ийг тушаалын мөрийн `-nat` тохируулгаар идэвхжүүлж болох юм. Мөн /etc/rc.conf файлын хувьсагч `ppp_nat` гэж байх бөгөөд энэ нь анхдагчаар идэвхжүүлэгдсэн байна.

Хэрэв та энэ боломжийг ашиглаж байгаа бол ирж байгаа холболтуудын цааш дамжуулалтыг идэвхжүүлэх дараах /etc/ppp/ppp.conf файлын тохируулгуудын ашигтайг харж болох юм:

```
nat port tcp 10.0.0.2:ftp ftp
nat port tcp 10.0.0.2:http http
```

эсвэл гадна талд ерөөсөө битгий итгээрэй

```
nat deny_incoming yes
```

#### 28.2.1.4. Системийн сүүлийн тохиргоо

Та одоо `ppp`-г тохируулчихлаа, гэхдээ яг ажиллахад бэлэн болохын өмнө хийх цөөн хэдэн зүйлс бий. Тэдгээр нь бүгд /etc/rc.conf файлыг засварлуулах болно.

Энэ файлтай дээрээс доош ажиллахдаа `hostname=` мөр тохируулагдсан эсэхийг шалгаарай, өөрөөр хэлбэл:

```
hostname="foo.example.com"
```

Хэрэв таны ISP танд статик IP хаяг болон нэр өгсөн бол та энэ нэрийг хостын нэрэндээ ашиглах нь магадгүй зүйтэй юм.

`network_interfaces` хувьсагчийг хайгаарай. Хэрэв та өөрийн системийг ISP уруугаа шаардлага хүсэлтээр залгахаар тохируулахыг хүсвэл tun0 төхөөрөмжийг жагсаалтад нэмсэн эсэхээ баталгаажуулаарай, хэрэв тэгж хүсээгүй бол түүнийг устгаарай.

```
network_interfaces="lo0 tun0"
ifconfig_tun0=
```

`ifconfig_tun0` хувьсагч хоосон байх ёстой бөгөөд /etc/start_if.tun0 гэж нэрлэгдсэн файлыг үүсгэх ёстой. Энэ файл нь дараах мөрийг агуулсан байх ёстой:

```
ppp -auto mysystem
```

Энэ скрипт нь сүлжээ тохируулах үед таны ppp дэмонг автомат горимд эхлүүлэн ажилладаг. Хэрэв та энэ машин гарц болж байгаа LAN-тай бол `-alias` сонголтыг ашиглахыг хүсэж болох юм. Илүү дэлгэрэнгүй мэдээллийг гарын авлагын хуудаснаас лавлана уу.

Таны /etc/rc.conf файлд чиглүүлэлтийн програм NO гэж дараах мөрөөр хийгдсэн эсэхийг шалгаарай:

```
router_enable="NO"
```

routed дэмон нь эхлэхгүй байх нь чухал юм. Учир нь routed нь ppp-ийн үүсгэсэн анхдагч чиглүүлэлтийн хүснэгтийн оруулгуудыг устгадаг юм.

sendmail_flags мөр -q тохируулгыг агуулаагүй эсэхийг шалгах нь магадгүй зүйтэй юм. Тэгэхгүй бол sendmail нь таны машиныг гадагш залгуулан үргэлж сүлжээний хайлт хийхийг оролдох болно. Та доор дурдсаныг тохируулж болох юм:

```
sendmail_flags="-bd"
```

Үүний сул тал нь ppp холбоос тогтсон үед та доор дурдсаныг бичиж sendmail-ээр захидлын дарааллыг дахин шалгуулж байх явдал юм:

```
# /usr/sbin/sendmail -q
```

Та үүнийг хийхийн тулд ppp.linkup файлд !bg тушаалыг ашиглахыг хүсэж болох юм:

```
1     provider:
2       delete ALL
3       add 0 0 HISADDR
4       !bg sendmail -bd -q30m
```

Хэрэв танд энэ таалагдахгүй байгаа бол SMTP урсгалыг хаахын тулд "dfilter"-г тохируулах боломжтой байдаг. Илүү дэлгэрэнгүй мэдээллийг жишээ файлуудаас лавлана уу.

Одоо машиныг дахин ачаалах л үлдлээ. Дахин ачаалсны дараа та доор дурдсаныг бичиж:

```
# ppp
```

дараа нь PPP сессийг эхлүүлэхийн тулд dial provider тушаалыг ажиллуулах юм уу эсвэл гадагшаа урсгал байгаа (бөгөөд та start_if.tun0 скрипт үүсгээгүй) бол сессүүдийг автоматаар тогтоодгоор ppp-г байлгахыг хүсэж байгаа бол доор дурдсаныг бичээрэй:

```
# ppp -auto provider
```

#### 28.2.1.5. Дүгнэлт

Дүгнэхэд ppp-г эхний удаа тохируулахад дараах алхмууд шаардлагатай:

Клиент тал:

1. Таны цөмд tun төхөөрөмж бүтээгдсэн эсэхийг шалгана.
2. /dev санд tunN төхөөрөмжийн файл байгаа эсэхийг шалгана.
3. /etc/ppp/ppp.conf файлд оруулга үүсгэнэ. pmdemand жишээ нь ихэнх ISP-уудын хувьд хангалттай байх ёстой.
4. Хэрэв та динамик IP хаягтай бол /etc/ppp/ppp.linkup файлд оруулга үүсгээрэй.
5. Өөрийн /etc/rc.conf файлыг шинэчилнэ.
6. Хэрэв танд хэрэгцээгээр залгах шаардлага байгаа бол start_if.tun0 скрипт үүсгээрэй.

Сервер тал:

1. Таны цөмд tun төхөөрөмж бүтээгдсэн эсэхийг шалгана.
2. /dev санд tunN төхөөрөмжийн файл байгаа эсэхийг шалгана.
3. /etc/passwd файлд (vipw(8) програмыг ашиглан) оруулга үүсгэнэ.
4. Энэ хэрэглэгчдийн гэрийн санд `ppp -direct direct-server` юм уу эсвэл үүнтэй адилыг ажиллуулах хувийн тохиргоог (profile) үүсгэнэ.
5. /etc/ppp/ppp.conf файлд оруулга үүсгэнэ. direct-server жишээ хангалттай байх ёстой.
6. /etc/ppp/ppp.linkup файлд оруулга үүсгэнэ.
7. Өөрийн /etc/rc.conf файлыг шинэчилнэ.

# 28.3. Цөмийн PPP-г ашиглах

Энэ хэсэг нь зөвхөн FreeBSD 7.X хувилбаруудад зориулагдсан.

## 28.3.1. Цөмийн PPP тохируулах

Та өөрийн машин дээр PPP-г тохируулж эхлэхээсээ өмнө `pppd` нь /usr/sbin санд байгаа болон /etc/ppp сан байгаа эсэхийг шалгаарай.

`pppd` нь хоёр горимд ажиллах чаддаг:

1. "Клиент" маягаар - та өөрийн машиныг гаднах ертөнц уруу PPP цуваа холболтоор эсвэл модемийн шугамаар холбохыг хүсдэг.
2. "Сервер" маягаар - Таны машин сүлжээн дээр байгаа бөгөөд бусад компьютеруудыг PPP ашиглан холбоход хэрэглэгдэнэ.

Аль ч тохиолдол байлаа гэсэн та тохируулгуудын файлыг (/etc/ppp/options эсвэл хэрэв та PPP ашиглаж байгаа өөрийн машин дээрээ нэгээс илүү олон хэрэглэгчидтэй бол ~/.ppprc) тохируулах хэрэгтэй болно.

Та залгаж алсын хосттой холболт тогтоохын тулд танд бас зарим модем/цуваа програм хангамж (comms/kermit-г аль болох эрхэмлэх) хэрэгтэй болно.

### 28.3.2. pppd-г клиент маягаар ашиглах

Дараах /etc/ppp/options файл нь Cisco терминал серверийн PPP шугам уруу холбогдоход хэрэглэгдэж болох юм.

```
crtscts         # enable hardware flow control
modem           # modem control line
noipdefault     # remote PPP server must supply your IP address
                # if the remote host does not send your IP during IPCP
                # negotiation, remove this option
passive         # wait for LCP packets
domain ppp.foo.com      # put your domain name here

:remote_ip    # put the IP of remote PPP host here
                # it will be used to route packets via PPP link
                # if you didn't specified the noipdefault option
                # change this line to local_ip:remote_ip

defaultroute    # put this if you want that PPP server will be your
                # default router
```

Холбогдохын тулд:

1. Алсын хост уруу Kermit (эсвэл өөр бусад модемийн програм) ашиглан залгаж өөрийн хэрэглэгчийн нэр болон нууц үгийг (эсвэл алсын хост дээр PPP-г идэвхжүүлэхэд шаардлагатай тэр зүйлийг) оруулна.
2. Kermit-с гарна (шугамыг таслалгүйгээр).
3. Доор дурдсаныг оруулна:

   ```
   # /usr/sbin/pppd /dev/tty01 19200
   ```

   Тохирох хурд болон төхөөрөмжийн нэрийг ашиглахаа мартуузай.

Одоо таны компьютер PPP-тэй холбогдлоо. Хэрэв холболт амжилтгүй болбол та /etc/ppp/options файлд debug тохируулгыг нэмж асуудлыг олохын тулд консолын мэдэгдлүүдийг шалгана.

Дараах /etc/ppp/pppup скрипт нь бүх 3 алхмыг автомат болгоно:

```
#!/bin/sh
pgrep -l pppd
pid=`pgrep pppd`
```

```
if [ "X${pid}" != "X" ] ; then
        echo 'killing pppd, PID=' ${pid}
        kill ${pid}
fi
pgrep -l kermit
pid=`pgrep kermit`
if [ "X${pid}" != "X" ] ; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi

ifconfig ppp0 down
ifconfig ppp0 delete

kermit -y /etc/ppp/kermit.dial
pppd /dev/tty01 19200
```

/etc/ppp/kermit.dial нь Kermit скрипт бөгөөд гадагш залгаж бүх шаардлагатай нэвтрэлт таниулалтыг алсын хост дээр хийдэг (энэ баримтын төгсгөлд ийм скриптийн жишээ хавсаргагдсан болно).

Дараах /etc/ppp/pppdown скриптийг ашиглаж PPP шугамыг салгаж болно:

```
#!/bin/sh
pid=`pgrep pppd`
if [ X${pid} != "X" ] ; then
        echo 'killing pppd, PID=' ${pid}
        kill -TERM ${pid}
fi

pgrep -l kermit
pid=`pgrep kermit`
if [ "X${pid}" != "X" ] ; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi

/sbin/ifconfig ppp0 down
/sbin/ifconfig ppp0 delete
kermit -y /etc/ppp/kermit.hup
/etc/ppp/ppptest
```

pppd ажиллаж байгаа эсэхийг шалгахын тулд /usr/etc/ppp/ppptest-г ажиллуулна. Энэ нь иймэрхүү харагдах ёстой:

```
#!/bin/sh
pid=`pgrep pppd`
if [ X${pid} != "X" ] ; then
        echo 'pppd running: PID=' ${pid-NONE}
```

```
else
        echo 'No pppd running.'
fi
set -x
netstat -n -I ppp0
ifconfig ppp0
```

Модемийг салгаж буцааж залгахын тулд /etc/ppp/kermit.hup-г ажиллуулна. Энэ нь доор дурдсаныг агуулах ёстой:

```
set line /dev/tty01 ; put your modem device here
set speed 19200
set file type binary
set file names literal
set win 8
set rec pack 1024
set send pack 1024
set block 3
set term bytesize 8
set command bytesize 8
set flow none

pau 1
out +++
inp 5 OK
out ATH0\13
echo \13
exit
```

Энд `kermit` тушаалын оронд `chat` тушаал хэрэглэх өөр арга байна:

Дараах хоёр мөр нь `pppd` холболтыг хийхэд хангалттай байдаг.

/etc/ppp/options:

```
/dev/cuad1 115200

crtscts     # enable hardware flow control
modem       # modem control line
connect "/usr/bin/chat -f /etc/ppp/login.chat.script"
noipdefault # remote PPP serve must supply your IP address
            # if the remote host doesn't send your IP during
                # IPCP negotiation, remove this option
passive         # wait for LCP packets
domain your.domain  # put your domain name here

:       # put the IP of remote PPP host here
            # it will be used to route packets via PPP link
                # if you didn't specified the noipdefault option
```

```
                # change this line to local_ip:remote_ip

defaultroute    # put this if you want that PPP server will be
            # your default router
```

/etc/ppp/login.chat.script:

Дараах нь нэг мөрөнд байх ёстой.

```
ABORT BUSY ABORT 'NO CARRIER' "" AT OK ATDTphone.number
  CONNECT "" TIMEOUT 10 ogin:-\\r-ogin: login-id
  TIMEOUT 5 sword: password
```

Эдгээр нь суулгагдаж зөв болж өөрчлөгдсөний дараа таны хийх ганц зүйл бол pppd-г иймэрхүү маягаар ажиллуулах явдал юм:

```
# pppd
```

### 28.3.3. pppd-г сервер маягаар ашиглах

/etc/ppp/options файл нь доор дурдсантай төстэйг агуулах ёстой:

```
crtscts                         # Hardware flow control
netmask 255.255.255.0           # netmask (not required)
192.114.208.20:192.114.208.165  # IP's of local and remote hosts
                                # local ip must be different from one
                                # you assigned to the Ethernet (or other)
                                # interface on your machine.
                                # remote IP is IP address that will be
                                # assigned to the remote machine
domain ppp.foo.com              # your domain
passive                         # wait for LCP
modem                           # modem line
```

Дараах /etc/ppp/pppserv скрипт нь pppd-г сервер маягаар ажиллахыг заах болно:

```
#!/bin/sh
pgrep -l pppd
pid=`pgrep pppd`
if [ "X${pid}" != "X" ] ; then
        echo 'killing pppd, PID=' ${pid}
        kill ${pid}
fi
pgrep -l kermit
pid=`pgrep kermit`
```

```
if [ "X${pid}" != "X" ] ; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi

# reset ppp interface
ifconfig ppp0 down
ifconfig ppp0 delete

# enable autoanswer mode
kermit -y /etc/ppp/kermit.ans

# run ppp
pppd /dev/tty01 19200
```

Серверийг зогсоохын тулд энэ /etc/ppp/pppservdown скриптийг ашиглана:

```
#!/bin/sh
pgrep -l pppd
pid=`pgrep pppd`
if [ "X${pid}" != "X" ] ; then
        echo 'killing pppd, PID=' ${pid}
        kill ${pid}
fi
pgrep -l kermit
pid=`pgrep kermit`
if [ "X${pid}" != "X" ] ; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi
ifconfig ppp0 down
ifconfig ppp0 delete

kermit -y /etc/ppp/kermit.noans
```

Дараах Kermit скрипт нь (/etc/ppp/kermit.ans) таны модем дээр автомат хариулагчийг нээх/хаах болно. Энэ нь иймэрхүү харагдах ёстой:

```
set line /dev/tty01
set speed 19200
set file type binary
set file names literal
set win 8
set rec pack 1024
set send pack 1024
set block 3
set term bytesize 8
set command bytesize 8
set flow none
```

```
pau 1
out +++
inp 5 OK
out ATH0\13
inp 5 OK
echo \13
out ATS0=1\13   ; change this to out ATS0=0\13 if you want to disable
                ; autoanswer mode
inp 5 OK
echo \13
exit
```

/etc/ppp/kermit.dial гэсэн нэртэй скрипт нь гадагш залгаж алсын хост дээр нэвтрэлт таниулахад хэрэглэгддэг. Та үүнийг өөрийн хэрэгцээнд зориулан өөрчлөх хэрэгтэй болно. Өөрийн нэвтрэх нэр болон нууц үгийг энэ скриптэд оруулах хэрэгтэй; та бас оролтын илэрхийллийг таны модем болон алсын хостоос ирэх хариунуудаас хамааран өөрчлөх хэрэгтэй болно.

```
;
; put the com line attached to the modem here:
;
set line /dev/tty01
;
; put the modem speed here:
;
set speed 19200
set file type binary            ; full 8 bit file xfer
set file names literal
set win 8
set rec pack 1024
set send pack 1024
set block 3
set term bytesize 8
set command bytesize 8
set flow none
set modem hayes
set dial hangup off
set carrier auto                ; Then SET CARRIER if necessary,
set dial display on             ; Then SET DIAL if necessary,
set input echo on
set input timeout proceed
set input case ignore
def \%x 0                       ; login prompt counter
goto slhup

:slcmd                          ; put the modem in command mode
echo Put the modem in command mode.
clear                           ; Clear unread characters from input buffer
pause 1
```

```
output +++                      ; hayes escape sequence
input 1 OK\13\10                ; wait for OK
if success goto slhup
output \13
pause 1
output at\13
input 1 OK\13\10
if fail goto slcmd              ; if modem doesn't answer OK, try again

:slhup                          ; hang up the phone
clear                           ; Clear unread characters from input buffer
pause 1
echo Hanging up the phone.
output ath0\13                  ; hayes command for on hook
input 2 OK\13\10
if fail goto slcmd              ; if no OK answer, put modem in command mode

:sldial                         ; dial the number
pause 1
echo Dialing.
output atdt9,550311\13\10               ; put phone number here
assign \%x 0                    ; zero the time counter

:look
clear                           ; Clear unread characters from input buffer
increment \%x                   ; Count the seconds
input 1 {CONNECT }
if success goto sllogin
reinput 1 {NO CARRIER\13\10}
if success goto sldial
reinput 1 {NO DIALTONE\13\10}
if success goto slnodial
reinput 1 {\255}
if success goto slhup
reinput 1 {\127}
if success goto slhup
if < \%x 60 goto look
else goto slhup

:sllogin                        ; login
assign \%x 0                    ; zero the time counter
pause 1
echo Looking for login prompt.

:slloop
increment \%x                   ; Count the seconds
clear                           ; Clear unread characters from input buffer
output \13
;
; put your expected login prompt here:
;
```

```
input 1 {Username: }
if success goto sluid
reinput 1 {\255}
if success goto slhup
reinput 1 {\127}
if success goto slhup
if < \%x 10 goto slloop         ; try 10 times to get a login prompt
else goto slhup                 ; hang up and start again if 10 failures

:sluid
;
; put your userid here:
;
output ppp-login\13
input 1 {Password: }
;
; put your password here:
;
output ppp-password\13
input 1 {Entering SLIP mode.}
echo
quit

:slnodial
echo \7No dialtone.  Check the telephone line!\7
exit 1

; local variables:
; mode: csh
; comment-start: "; "
; comment-start-skip: "; "
; end:
```

# 28.4. PPP холболтуудын алдааг олж засварлах

Энэ хэсэг нь PPP-г модемийн холболтоор ашиглах үед гарч болох цөөн асуудлуудыг хамрах болно. Жишээ нь магадгүй танд залгаж холбогдох гэж байгаа систем чинь яг ямар хүлээх мөрүүдийг үзүүлэхийг мэдэх шаардлагатай байж болох юм. Зарим ISP-ууд `ssword` хүлээх мөрийг үзүүлдэг бол зарим нь `password` гэж үзүүлэх болно; хэрэв `ppp` скрипт нь түүний дагуу бичигдээгүй бол нэвтрэх оролдлого амжилттүй болох болно. `ppp` холболтуудыг дибаг хийх хамгийн түгээмэл арга бол гараар холбогдох явдал юм. Дараах мэдээлэл нь алхам алхмаар гар холболтыг танд тайлбарлах болно.

## 28.4.1. Төхөөрөмжийн цэгүүдийг шалгана

Хэрэв өөрчлөн тохируулсан цөм ашиглаж байгаа бол дараах мөрийг өөрийн цөмийн тохиргооны файлдаа нэмсэн эсэхээ шалгаарай:

```
device   uart
```

uart төхөөрөмж GENERIC цөмд аль хэдийн орсон байдаг учир энэ тохиолдолд нэмэлт алхмууд хэрэггүй юм. Модем төхөөрөмжийн хувьд dmesg гаралтыг шалгаарай:

```
# dmesg | grep uart
```

Та uart төхөөрөмжүүдийн талаар тохирох мэдээллүүдийг авах ёстой. Эдгээр нь бидэнд хэрэгтэй COM портууд юм. Хэрэв таны модем стандарт цуваа порт шиг ажиллаж байвал та түүнийг uart1, эсвэл COM2 дээр жагсаагдсан байхыг харах ёстой. Хэрэв тийм бол та цөмийг дахин бүтээх шаардлагагүй юм. sio модем uart1 дээр эсвэл хэрэв та DOS дээр байгаа бол COM2 дээр таарч байвал таны модемийн төхөөрөмж /dev/cuau1 байх болно.

### 28.4.2. Гараар холбогдох нь

Гараар ppp-г хянаж Интернэтэд холбогдох нь холболтыг дибаг хийх юм уу эсвэл ердөө л таны ISP чинь ppp клиент холболтуудад хэрхэн ханддаг талаар мэдээлэл авахад хурдан, хялбар, агуу арга байдаг. Тушаалын мөрөөс PPP-г эхлүүлье. Бидний бүх жишээнүүд дээр бид PPP ажиллуулж байгаа машины хостын нэрийг *example* гэж авч ашиглах болно. Та ppp-г ppp гэж бичин эхлүүлнэ:

```
# ppp
```

Бид одоо ppp-г эхлүүллээ.

```
ppp ON example> set device /dev/cuau1
```

Бид өөрсдийн модемний төхөөрөмжийг тохируулна. Энэ тохиолдолд cuau1 болно.

```
ppp ON example> set speed 115200
```

Холболтын хурдыг тохируулна. Энэ тохиолдолд бид 115,200 kbps-г ашиглаж байна.

```
ppp ON example> enable dns
```

Бидний нэр танигчийг тохируулж /etc/resolv.conf файлд нэрийн серверүүдийн мөрүүдийг нэмэхийг ppp-д хэлнэ. ppp бидний хостын нэрийг тодорхойлж чадахгүй байгаа бол бид дараа нь гараараа тохируулж өгч болно.

```
ppp ON example> term
```

Модемийг гараар хянаж чадаж байхын тулд "Терминал" горимд шилжих хэрэгтэй.

```
deflink: Entering terminal mode on /dev/cuau1
type '~h' for help
```

```
at
OK
atdt123456789
```

Модемийг эхлүүлэхийн тулд `at`-г ашиглаж дараа нь `atdt` дээр өөрийн ISP-ийн дугаараа нэмээд залгах процессоо эхлэх хэрэгтэй.

```
CONNECT
```

Холболтын баталгаа гарах ёстой, хэрэв бид тоног төхөөрөмжтэй холбоогүй ямар нэг холболтын асуудлуудтай байх бол энд бид тэдгээрийг шийдэхийг оролдох болно.

```
ISP Login:myusername
```

Энд танаас хэрэглэгчийн нэрийг асуух болно, ISP-ээс өгсөн хэрэглэгчийн нэрийг оруулаад буцахыг дарна.

```
ISP Pass:mypassword
```

Энэ удаад нууц үгийг асуух болно, ISP-ээс өгсөн нууц үгээр хариулах хэрэгтэй. Яг л FreeBSD уруу нэвтрэн ордог шиг нууц үг нь буцаж дэлгэцэд харагдахгүй.

```
Shell or PPP:ppp
```

Таны ISP-ээс хамаараад энэ хүлээх мөр нь хэзээ ч гарахгүй байж болох юм. Энд биднээс үйлчилгээ үзүүлэгч дээр бүрхүүл ашиглах эсэх эсвэл `ppp` эхлүүлэх эсэхийг асуудаг. Энэ жишээн дээр бид Интернэт холболтыг хүсэж байгаа болохоор `ppp`-г ашиглахаар сонгосон.

```
Ppp ON example>
```

Энэ жишээн дээр эхний `p` тохируулга томоор бичигдсэнийг хараарай. Энэ нь бид ISP-д амжилттайгаар холбогдсоныг харуулж байна.

```
PPp ON example>
```

Бид ISP-д амжилттайгаар өөрсдийгөө таниулан нэвтэрч заагдсан IP хаягийг хүлээж байна.

```
PPP ON example>
```

Бид IP хаяг дээрээ зөвшилцөж холболтоо амжилттайгаар хийж дуусгалаа.

```
PPP ON example>add default HISADDR
```

Энд бид өөрсдийн анхдагч чиглүүлэлтээ нэмж байна. Бид үүнийг гаднах ертөнцтэй ярьж эхлэхээсээ өмнө хийх ёстой бөгөөд одоогоор зөвхөн хийгдсэн холболт нь нөгөө талтайгаа хийгдсэн холболт байгаа болно. Хэрэв энэ нь байгаа чиглүүлэлтүүдээсээ болоод амжилтгүй болбол та `add` тохируулгын өмнө анхаарлын тэмдэг `!` тавьж болно. Өөрөөр та үүнийг жинхэнэ холболт хийхээсээ өмнө тохируулж болох бөгөөд энэ нь шинэ чиглүүлэлтийг зохих ёсоор тохиролцоно.

Хэрэв бүх зүйл сайн болж өнгөрсөн бол бид одоо идэвхтэй Интернэтийн холболттой болсон байх бөгөөд үүнийг ар талд `CTRL` + `z` хослолоор оруулж болох юм. Хэрэв та `PPP`-г `ppp` болохыг анзаарах юм бол бид өөрсдийн холболтоо алдсан гэсэн үг юм. Энэ нь бидний холболтын төлөвийг үзүүлдэг болохоор мэдэхэд илүүдэхгүй юм. Том P үсэгнүүд нь бид ISP уруу холболттой байгааг үзүүлэх бөгөөд жижиг p үсэгнүүд нь холболт ямар нэг шалтгаанаар тасарсныг харуулна. `ppp` нь зөвхөн эдгээр хоёр төлөвтэй байдаг.

#### 28.4.2.1. Дибаг хийх нь

Хэрэв та шууд шугамтай бөгөөд холболт хийж чадахгүй байгаа юм шиг санагдвал CTS/RTS тоног төхөөрөмжийн урсгалыг `set ctsrts off` болгон хаах хэрэгтэй. Энэ нь ялангуяа та PPP хийж чаддаг терминал серверт холбогдох тохиолдол юм. Энэ тохиолдолд PPP нь таны холбооны шугам дээр өгөгдлийг бичихээр оролдох үед өлгөгдөж хэзээ ч ирэхгүй байж болох CTS юм уу эсвэл Clear To Send буюу Илгээхэд Цэвэр дохиог хүлээж байж болох юм. Хэрэв та энэ тохируулгыг ашиглах юм бол `set accmap` тохируулгыг бас хэрэглэх шаардлагатай. Энэ тохируулга нь зарим нэг параметрүүдийг, ихэнх тохиолдолд XON/XOFF-г төгсгөлөөс төгсгөлд дамжуулахад тоног төхөөрөмжийн хувьд хамааралтай байдлыг үгүй хийхийн тулд шаардлагатай байж болох юм. Энэ тохируулгын талаар дэлгэрэнгүй мэдээллийг болон хэрхэн ашиглагддаг талаар ppp(8) гарын авлагын хуудаснаас үзнэ үү.

Хэрэв та хуучин модемтой бол `set parity even` тохируулгыг ашиглах хэрэгтэй болж болох юм. Parity нь анхдагчаар none буюу байхгүй гэж тохируулагдсан байдаг боловч хуучин модемууд болон ISP-уудад (урсгалын хувьд их хэмжээгээр ихсэхэд) алдаа шалгахад хэрэглэгддэг. Compuserve ISP-ийн хувьд танд энэ тохируулга хэрэгтэй байж болох юм.

PPP нь тушаалын горимд буцахгүй байж болох юм. Энэ нь ихэвчлэн ISP нь танаас харилцан тохиролцохыг эхлүүлэхийг хүлээх тохиролцооны алдаа байдаг. Энэ үед `~p` тушаалыг ашиглах нь ppp-г тохиргооны мэдээллээ илгээж эхлэхэд хүргэх болно.

Хэрэв та нэвтрэлт хүлээх мөрийг хэзээ ч авахгүй байгаа бол та магадгүй дээрх жишээн дээрх UNIX® загварын нэвтрэлт танилтын оронд PAP юм уу эсвэл CHAP нэвтрэлт танилтыг ашиглах хэрэгтэй байж болох юм. PAP эсвэл CHAP-г ашиглахын тулд ердөө л дараах тохируулгуудыг PPP-д терминал горим уруу орохоос өмнө нэмэх хэрэгтэй:

```
ppp ON example> set authname myusername
```

Дээр байгаа *myusername*-г ISP-с өгөгдсөн хэрэглэгчийн нэрээр солих шаардлагатай.

```
ppp ON example> set authkey mypassword
```

Дээр байгаа *mypassword*-г ISP-с өгөгдсөн нууц үгээр солих шаардлагатай.

Хэрэв та зүгээр холбогдсон мөртлөө ямар ч домэйн нэр олохгүй байгаа юм шиг байвал ping(8) тушаалыг IP хаягтай ашиглаж ямар нэг мэдээлэл буцаж ирж байгаа эсэхийг шалгах хэрэгтэй. Хэрэв та 100 хувийн (100%) пакетийн алдагдалд орж байвал танд бараг л анхдагч чиглүүлэлт заагдаагүй байгаа гэсэн үг юм. Холболтын үеэр `add default HISADDR` тохируулга тохируулагдсан эсэхийг давхар шалгаарай. Хэрэв та алсын IP хаяг уруу холбогдож чадаж байвал нэр танигчийн хаяг /etc/resolv.conf файлд нэмэгдээгүй байх магадлалтай юм. Энэ файл нь иймэрхүү харагдах ёстой:

```
domain example.com
nameserver x.x.x.x
nameserver y.y.y.y
```

Дээр байгаа *x.x.x.x* болон *y.y.y.y* нь таны ISP-ийн DNS серверүүдийн IP хаягаар солигдох ёстой. Таныг үйлчилгээнд эхэлж бүртгүүлэхэд энэ мэдээлэл өгөгдсөн юм уу эсвэл өгөгдөөгүй байж болох бөгөөд өөрийн ISP руугаа хурдхан утасдаж энэ мэдээллийг мэдэж болох юм.

Та өөрийн PPP холболтондоо зориулж syslog(3)-г бүртгэл бичихээр болгож болно. Дараах:

```
!ppp
*.*     /var/log/ppp.log
```

мөрийг /etc/syslog.conf файлд нэмнэ. Ихэнх тохиолдолд энэ боломж аль хэдийн орсон байдаг.

# 28.5. Ethernet дээгүүр PPP ашиглах нь (PPPoE)

Энэ хэсэг нь Ethernet дээгүүр PPP (PPPoE) хэрхэн тохируулах талаар тайлбарлах болно.

## 28.5.1. Цөмийг тохируулах нь

Одоо PPPoE-д зориулсан цөмийн тохиргоо шаардлагатай байхаа больжээ. Хэрэв шаардлагатай netgraph-ийн дэмжлэг цөмд бүтээгдээгүй бол энэ нь динамикаар ppp-ээр дуудагдах болно.

## 28.5.2. ppp.conf файлыг тохируулах

Энд ажилладаг ppp.conf файлын жишээ байна:

```
default:
  set log Phase tun command # you can add more detailed logging if you wish
  set ifaddr 10.0.0.1/0 10.0.0.2/0

name_of_service_provider:
  set device PPPoE:xl1 # replace xl1 with your Ethernet device
  set authname YOURLOGINNAME
  set authkey YOURPASSWORD
  set dial
  set login
  add default HISADDR
```

### 28.5.3. ppp-г ажиллуулах

`root` хэрэглэгчээр та доор дурдсаныг ажиллуулж болно:

```
# ppp -ddial name_of_service_provider
```

### 28.5.4. ppp-г ачаалах үед эхлүүлэх

доор дурдсаныг өөрийн /etc/rc.conf файлд нэмнэ:

```
ppp_enable="YES"
ppp_mode="ddial"
ppp_nat="YES"   # if you want to enable nat for your local network, otherwise NO
ppp_profile="name_of_service_provider"
```

### 28.5.5. PPPoE үйлчилгээний хаягийг ашиглах

Заримдаа өөрийн холболтыг хийхийн тулд үйлчилгээний хаягийг ашиглах шаардлагатай болж болох юм. Үйлчилгээний хаягууд нь өгөгдсөн сүлжээнд залгагдсан өөр өөр PPPoE серверүүдийг ялгахын тулд хэрэглэгддэг.

Таны ISP-ээс өгсөн баримтад шаардлагатай үйлчилгээний хаягийн мэдээлэл байх ёстой. Хэрэв та ийм мэдээлэл олоогүй бол өөрийн ISP-ийн техникийн дэмжлэг үзүүлэгч ажилтнаас асуугаарай.

Хамгийн сүүлд та Портын цуглуулгад байх Архирах Оцон шувуу PPPoE хаягт санал болгосон аргыг туршиж болох юм. Гэхдээ энэ нь таны модемийг буцаан програмчилж ажиллагаагүй болгож магадгүй болохоор үүнийг хийхээсээ өмнө дахин бодоорой. Таны үйлчилгээ үзүүлэгчийн модемтой цуг ирсэн програмыг ердөө л суулгаарай. Дараа нь програмаас **System** цэсд хандаарай. Таны тохиргооны нэр тэнд жагсаагдсан байх ёстой. Энэ нь ихэвчлэн *ISP* гэж байдаг.

Тохиргооны нэр (үйлчилгээний хаяг) нь ppp.conf файл дахь PPPoE тохиргооны оруулгад `set device` тушаалын үйлчилгээ үзүүлэгч хэсэг болон ашиглагдах болно (бүрэн мэдээллийг ppp(8)

гарын авлагын хуудаснаас үзнэ үү). Энэ нь иймэрхүү харагдах ёстой:

```
set device PPPoE:xl1:ISP
```

*xl1*-г өөрийн Ethernet картны зөв төхөөрөмжөөр солихоо мартуузай.

*ISP* гэснийг дээрх дөнгөж олсон тохиргоогоор солихоо мартуузай.

Нэмэлт мэдээллийг доорх хаягуудаас үзнэ үү:

- Рено Валдурагийн DSL дээр FreeBSD-ээр арай хямд өргөн зурвас ашиглах (Cheaper Broadband with FreeBSD on DSL).

### 28.5.6. 3Com® HomeConnect™ ADSL Хос Холбоос Модемтой PPPoE

Энэ модем нь RFC 2516 (Л. Мамакос, К. Лидл, Ж. Эвартс, Д. Каррел, Д. Симоне, болон Р. Вийлэр нарын бичсэн *PPP-г Ethernet дээгүүр (PPPoE) дамжуулах арга*) стандартыг дагадаггүй. Харин Ethernet хүрээнүүдийн хувьд пакетийн төрлийн өөр кодуудыг ашигладаг. Хэрэв та үүнийг PPPoE-ийн тодорхойлолтыг мөрдөх ёстой гэж бодож байгаа бол 3Com уруу гомдоллоно уу.

Энэ төхөөрөмжтэй FreeBSD-г холбогдох боломжтой болгохын тулд sysctl тохируулагдах ёстой. /etc/sysctl.conf файлыг шинэчилж ачаалалтын үед энэ нь автоматаар хийгдэж болно:

```
net.graph.nonstandard_pppoe=1
```

эсвэл дараах тушаалаар нэн даруй хийгдэж болно:

```
# sysctl net.graph.nonstandard_pppoe=1
```

Харамсалтай нь энэ нь системийн дагуух тохиргоо болохоор хэвийн PPPoE клиент эсвэл сервер болон 3Com® HomeConnect™ ADSL модемтой нэгэн зэрэг холбогдох боломжгүй болох юм.

## 28.6. ATM дээгүүр PPP (PPPoA) ашиглах

Дараах нь ATM дээгүүр PPP-г (PPPoA) хэрхэн тохируулах талаар тайлбарлана. PPPoA нь Европийн DSL үзүүлэгчдийн дунд алдартай сонголт болдог.

### 28.6.1. Alcatel SpeedTouch™ USB-тэй PPPoA-г ашиглах нь

Энэ төхөөрөмжийн PPPoA дэмжлэг нь FreeBSD-д порт хэлбэрээр байдаг. Учир нь үүний firmware Alcatel-ийн лицензийн гэрээгээр түгээгддэг бөгөөд FreeBSD-ийн үндсэн системтэй цуг чөлөөтэй түгээгдэж болдоггүй.

Програм хангамжийг суулгахын тулд ердөө л Портуудын цуглуулгыг ашиглана. net/pppoa портыг суулгаад түүнтэй хамт ирсэн зааврыудыг дагаарай.

Олон USB төхөөрөмжүүдийн адил Alcatel-ийн SpeedTouch™ USB нь зөв ажиллахын тулд хост компьютераас firmware татах хэрэгтэй болдог. FreeBSD дээр энэ дамжуулалт нь төхөөрөмж USB порт уруу холбогдсон үед хийгдэхээр болгож энэ процессийг автоматчилах боломжтой. Энэ автомат firmware дамжуулалтыг идэвхжүүлэхийн тулд дараах мэдээллийг /etc/usbd.conf файлд нэмж болох юм. Энэ файлыг `root` хэрэглэгчээр засварлах шаардлагатай.

```
device "Alcatel SpeedTouch USB"
    devname "ugen[0-9]+"
    vendor 0x06b9
    product 0x4061
    attach "/usr/local/sbin/modem_run -f /usr/local/libdata/mgmt.o"
```

usbd USB дэмонг идэвхжүүлэхийн тулд дараах мөрийг /etc/rc.conf файлд нэмнэ:

```
usbd_enable="YES"
```

Эхлүүлэхэд залгадаг байхаар ppp-г тохируулах боломжтой байдаг. Үүнийг хийхийн тулд дараах мөрүүдийг /etc/rc.conf файлд нэмнэ. Ахин хэлэхэд энэ процедурын хувьд та `root` хэрэглэгчээр нэвтрэн орох хэрэгтэй болно.

```
ppp_enable="YES"
ppp_mode="ddial"
ppp_profile="adsl"
```

Үүнийг зөв ажиллуулахын тулд та net/pppoa порттой цуг ирсэн ppp.conf файлыг ашиглах хэрэгтэй болно.

### 28.6.2. mpd ашиглах нь

Та төрөл бүрийн үйлчилгээнүүдэд ялангуяа PPTP үйлчилгээнүүдэд холбогдохын тулд mpd-г ашиглаж болно. Та mpd-г Портуудын цуглуулга net/mpd сангаас олж болно. Олон ADSL модемууд нь модем болон компьютерийн хооронд PPTP хоолой үүсгэгдсэн байхыг шаарддаг бөгөөд тийм модемийн нэг нь Alcatel SpeedTouch™ Home юм.

Та эхлээд портоо суулгах ёстой бөгөөд дараа нь өөрийн шаардлага болон үйлчилгээ үзүүлэгчийн тохиргоонуудад тааруулан mpd-г тохируулж болно. Порт нь жишээ тохиргооны файлуудыг байрлуулдаг бөгөөд эдгээр нь PREFIX/etc/mpd/-д сайн баримтжуулагдсан байдаг. Энд нэг юмыг тэмдэглэх хэрэгтэй. *PREFIX* гэдэг нь таны портууд суулгагдсан сан бөгөөд анхдагчаар /usr/local/ байдаг. mpd-г тохируулах бүрэн заавар порт суулгагдсаны дараа HTML хэлбэрээр орсон байдаг. Энэ нь PREFIX/shared/doc/mpd/ санд байрладаг. Энд mpd-ээр ADSL үйлчилгээнд холбогдох жишээ тохиргоо байна. Тохиргоо нь хоёр файлаас тогтох бөгөөд эхний mpd.conf нь ийм байна:

Энэ жишээ mpd.conf файл нь mpd 4.x хувилбартай зөвхөн ажиллана.

```
default:
    load adsl

adsl:
    new -i ng0 adsl adsl
    set bundle authname username ①
    set bundle password password ②
    set bundle disable multilink

    set link no pap acfcomp protocomp
    set link disable chap
    set link accept chap
    set link keep-alive 30 10

    set ipcp no vjcomp
    set ipcp ranges 0.0.0.0/0 0.0.0.0/0

    set iface route default
    set iface disable on-demand
    set iface enable proxy-arp
    set iface idle 0

    open
```

① Таны ISP-тай холбогдон өөрийгөө таниулан нэвтрэх хэрэглэгчийн нэр.

② Таны ISP-тай холбогдон өөрийгөө таниулан нэвтрэх нууц үг.

mpd.links файл нь таны хийхийг хүссэн холбоос эсвэл холбоосуудын тухай мэдээллийг агуулдаг. Дээрх жишээнд тохирох харгалзах жишээ mpd.links файлыг доор үзүүлэв:

```
adsl:
    set link type pptp
    set pptp mode active
    set pptp enable originate outcall
    set pptp self 10.0.0.1 ①
    set pptp peer 10.0.0.138 ②
```

① Таны mpd ашиглах FreeBSD компьютерийн IP хаяг.

② Таны ADSL модемийн IP хаяг. Alcatel SpeedTouch™ Home-ийн хувьд энэ хаяг нь анхдагчаар `10.0.0.138` байна.

Дараах тушаалыг `root`-ээр ажиллуулж холболтыг хялбараар эхлүүлэх боломжтой байдаг:

```
# mpd -b adsl
```

Та холболтын төлөвийг дараах тушаалаар харж болно:

```
% ifconfig ng0
ng0: flags=88d1<UP,POINTOPOINT,RUNNING,NOARP,SIMPLEX,MULTICAST> mtu 1500
     inet 216.136.204.117 --> 204.152.186.171 netmask 0xffffffff
```

FreeBSD дээр ADSL үйлчилгээнд холбогдохын тулд mpd-г ашиглахыг зөвлөдөг юм.

### 28.6.3. pptpclient ашиглах нь

Мөн бусад PPPoA үйлчилгээнүүдэд FreeBSD-ээр холбогдохын тулд net/pptpclient ашиглах боломжтой байдаг.

net/pptpclient-г ашиглан DSL үйлчилгээнд холбогдохын тулд порт болон багцыг суулгаад өөрийн /etc/ppp/ppp.conf файлыг засварлана. Эдгээр үйлдлүүдийг хийж гүйцэтгэхийн тулд та `root` болсон байх ёстой. ppp.conf файлын жишээ хэсгийг доор үзүүлэв. ppp.conf-ийн тохируулгуудын тухай илүү мэдээллийг ppp гарын авлагын хуудас ppp(8)-с лавлана уу.

```
adsl:
 set log phase chat lcp ipcp ccp tun command
 set timeout 0
 enable dns
 set authname username ①
 set authkey password ②
 set ifaddr 0 0
 add default HISADDR
```

① DSL үйлчилгээ үзүүлэгчтэй холбогдох таны бүртгэлийн хэрэглэгчийн нэр.

② Таны бүртгэлийн нууц үг.

Та өөрийн бүртгэлийн нууц үгийг ppp.conf файлд цэвэр текст хэлбэрээр тавих ёстой болохоор энэ файлын агуулгыг хэн ч уншиж чадахгүй болгосон эсэхээ шалгах хэрэгтэй. Дараах сери тушаалууд нь энэ файлыг зөвхөн `root` бүртгэлээр уншигдахаар болгох юм. Дэлгэрэнгүй мэдээллийг chmod(1) болон chown(8) гарын авлагын хуудаснуудаас лавлана уу.

```
# chown root:wheel /etc/ppp/ppp.conf
# chmod 600 /etc/ppp/ppp.conf
```

Энэ нь PPP сессэд зориулан таны DSL чиглүүлэгч уруу хоолой нээх болно. Ethernet DSL модемууд нь LAN IP хаягаар урьдчилан тохируулагдсан байдаг бөгөөд та түүн уруу холбогдоно. Alcatel SpeedTouch™ Home-ийн хувьд энэ хаяг нь `10.0.0.138` байна. Таны чиглүүлэгчийн баримтад төхөөрөмж ямар хаяг ашиглаж байгааг хэлсэн байх ёстой. Хоолой нээж PPP сесс эхлүүлэхийн тулд дараах тушаалыг ажиллуулна:

```
# pptp address adsl
```

Та өмнөх тушаалын төгсгөлд "&" тэмдгийг нэмэхийг хүсэж болох юм. Ингэхгүй бол pptp нь танд тушаал хүлээх мөрийг буцаадаггүй.

tun виртуал хоолой төхөөрөмж нь pptp болон ppp процессуудын хоорондох харилцаанд зориулан үүсгэгдэх болно. Тушаал хүлээх мөрөнд буцаж ирснийхээ дараа юм уу эсвэл pptp нь холболтыг баталгаажуулсны дараа та хоолойг ингэж шалгаж болно:

```
% ifconfig tun0
tun0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1500
        inet 216.136.204.21 --> 204.152.186.171 netmask 0xffffff00
        Opened by PID 918
```

Хэрэв та холбогдож чадахгүй байгаа бол өөрийн чиглүүлэгчийн тохиргоог шалгаарай. Энэ нь ихэвчлэн telnet юм уу эсвэл вэб хөтчөөр хандах боломжтой байдаг. Хэрэв та холбогдож чадахгүй хэвээр байвал `pptp` тушаалын гаралт болон ppp бүртгэлийн файл /var/log/ppp.log -оос ямар нэг сэжүүр байгаа эсэхийг шалгах хэрэгтэй.

# 28.7. SLIP ашиглах

Энэ хэсэг зөвхөн FreeBSD 7.X хувилбаруудад зориулагдсан.

## 28.7.1. SLIP клиент тохируулах

Дараах нь FreeBSD машиныг SLIP-д зориулан статик хост сүлжээн дээр тохируулах нэг арга юм. Динамикаар хостын нэрийг өгөхийн тулд (таныг залгаж холбогдох болгонд таны хаяг өөрчлөгддөг) танд илүү төвөгтэй тохируулга хэрэгтэй болох байх.

Эхлээд таны модем аль цуваа портод холбогдсоныг тодорхойлно. Олон хүн /dev/modem гэх зэргээр жинхэнэ төхөөрөмжийн нэр /dev/cuadN уруугаа заасан симболын холбоосыг тохируулдаг. Энэ нь модемийг өөр порт уруу шилжүүлэх шаардлагатай болсон тохиолдолд жинхэнэ төхөөрөмжийн нэрийг хийсвэрлэх боломжийг танд олгож байгаа юм. Бүх системийн дагуу /etc дэх олон файлууд болон .kermrc файлуудыг засварлах шаардлагатай болох үед энэ нь нэлээн ярвигтай болж ирдэг!

/dev/cuad0 нь COM1, /dev/cuad1 нь COM2, гэх мэт.

Өөрийн цөмийн тохиргооны файлд дараах нь байгаа эсэхийг шалгаарай:

```
device   sl
```

Энэ нь GENERIC цөмд орсон байдаг бөгөөд та түүнийг устгаагүй л бол энэ нь асуудал биш юм.

#### 28.7.1.1. Зөвхөн нэг удаа хийх зүйлс

1. Өөрийн гэр машин, гарц болон нэрийн серверүүдийг өөрийн /etc/hosts файлд нэмнэ. Биднийх иймэрхүү харагдах болно:

```
127.0.0.1               localhost loghost
136.152.64.181          water.CS.Example.EDU water.CS water
136.152.64.1            inr-3.CS.Example.EDU inr-3 slip-gateway
128.32.136.9            ns1.Example.EDU ns1
128.32.136.12           ns2.Example.EDU ns2
```

2. Өөрийн /etc/nsswitch.conf файлын `hosts:` хэсэгт `files` мөр нь `dns` мөрөөс өмнө байгаа эсэхийг шалгаарай. Эдгээр параметрүүдгүй бол сонин зүйлс болж магадгүй юм.
3. /etc/rc.conf файлыг засна.
   a. Дараах мөрийг засварлаж өөрийн хостын нэрийг тохируулна:

   ```
   hostname="myname.my.domain"
   ```

   Таны машины Интернэтийн бүрэн хостын нэрийг энд тавих ёстой. ..

   Дараах мөрийг өөрчилж анхдагч чиглүүлэлт:

   ```
   defaultrouter="NO"
   ```

   гэдгийг:

   ```
   defaultrouter="slip-gateway"
   ```

   болгоно.
4. доор дурдсаныг агуулсан /etc/resolv.conf файл үүсгэнэ:

```
domain CS.Example.EDU
nameserver 128.32.136.9
nameserver 128.32.136.12
```

   Таны харж байгаагаар эдгээр нь нэрийн серверийн хостуудыг тохируулж байна. Мэдээж жинхэнэ домэйн нэрс болон хаягууд нь таны орчноос хамаарна.
5. `root` болон `toor` (нууц үггүй бусад бүртгэлүүдийн) нууц үгийг тохируулна.
6. Өөрийн машиныг дахин ачаалж зөв хостын нэртэйгээр ачаалсан эсэхийг нь шалгаарай.

#### 28.7.1.2. SLIP холболт хийх нь

1. Залгаад хүлээх мөрөн дээр `slip` гэж бичээд өөрийн машины нэр болон нууц үгийг оруулна. Юу оруулах шаардлагатай нь таны орчноос хамаарна. Хэрэв та Kermit ашиглаж байгаа бол та иймэрхүү скриптийг ажиллуулахаар оролдож болно:

```
# kermit setup
set modem hayes
set line /dev/modem
set speed 115200
set parity none
set flow rts/cts
set terminal bytesize 8
set file type binary
# The next macro will dial up and login
define slip dial 643-9600, input 10 =>, if failure stop, -
output slip\x0d, input 10 Username:, if failure stop, -
output silvia\x0d, input 10 Password:, if failure stop, -
output ***\x0d, echo \x0aCONNECTED\x0a
```

   Мэдээж өөртөө тааруулаад өөрийн хэрэглэгчийн нэр болон нууц үгийг солих хэрэгтэй. Ингэснийхээ дараа та Kermit-ийн хүлээх мөрөөс холбогдохын тулд `slip` гэж бичих хэрэгтэй.

   Файлын системд хаа нэгтээ өөрийн нууц үгийг цэвэр текстээр үлдээх нь ерөнхийдөө *буруу* юм. Үүнийг хийх нь аз туршсан явдал юм.

2. Kermit-г тэнд нь үлдээж (та үүнийг `Ctrl` + `z` хослолоор зогсоож болно) `root`-ээр дараах тушаалыг бичнэ:

```
# slattach -h -c -s 115200 /dev/modem
```

   Хэрэв та чиглүүлэгчийн нөгөө талд байгаа хостууд уруу `ping` хийж чадаж байвал та холбогдсон байна! Хэрэв энэ нь ажиллахгүй байгаа бол `slattach` тушаалд `-c` нэмэлт өгөгдлийн оронд `-a`-г тавьж оролдоод үзээрэй.

#### 28.7.1.3. Холболтыг хэрхэн зогсоох вэ

доор дурдсаныг хийж:

```
# kill -INT `cat /var/run/slattach.modem.pid`
```

`slattach`-г зогсооно. Дээр дурдсаныг хийхийн тулд `root` байх шаардлагатайг санаарай. Дараа нь `kermit` уруугаа (хэрэв та түр зогсоосон бол `fg` тушаалыг ажиллуулан) буцаж түүнээс гарна (`q`).

slattach(8) гарын авлагын хуудсанд интерфэйсийг зогссон гэж тэмдэглэхийн тулд `ifconfig sl0 down` тушаалыг ашиглах ёстой гэсэн байдаг боловч энэ нь ямар ч өөрчлөлт хийдэггүй юм шиг санагддаг. (`ifconfig sl0` тушаал нь бас үүнтэй адилыг гаргадаг.)

Заримдаа таны модем зөөгчийг унагахаасаа татгалзаж болох юм. Тийм тохиолдолд ердөө л дахин `kermit`-г ажиллуулаад тэгээд түүнээс гарах хэрэгтэй. Энэ нь ихэвчлэн хоёр дахь оролдлогод болдог.

#### 28.7.1.4. Алдааг олж засварлах

Хэрэв энэ нь ажиллахгүй байгаа бол freebsd-net захидлын жагсаалтаас чөлөөтэй асуугаарай. Одоогоор хүмүүсийн аялж мэдсэн зүйлс гэх юм бол:

- `slattach` тушаалд `-c` эсвэл `-a` тохируулгуудыг ашиглахгүй байх (энэ нь мөхлийн байх ёсгүй, гэхдээ энэ нь зарим хэрэглэгчдийн асуудлуудыг шийддэг гэж бичсэн байдаг.)
- `sl0`-ийн оронд `s10` тохируулгыг ашиглах (зарим фонтууд дээр ялгааг нь олж харах хэцүү байж болох юм).
- Өөрийн интерфэйсийн төлөвийг харахын тулд `ifconfig sl0` тушаалыг ажиллуулж үзээрэй. Жишээ нь та иймэрхүү дүн харж болох юм:

```
# ifconfig sl0
sl0: flags=10<POINTOPOINT>
        inet 136.152.64.181 --> 136.152.64.1 netmask ffffff00
```

- Хэрэв та ping(8) тушаалаас `no route to host` мэдэгдлүүдийг авах юм бол таны чиглүүлэлтийн хүснэгтэд асуудал байж болзошгүй юм. Тухайн үед байгаа чиглүүлэлтүүдийг үзүүлэхийн тулд `netstat -r` тушаалыг ашиглаж болно :

```
# netstat -r
Routing tables
Destination      Gateway            Flags     Refs     Use  IfaceMTU    Rtt
Netmasks:

(root node)
(root node)

Route Tree for Protocol Family inet:
(root node) =>
default          inr-3.Example.EDU  UG          8   224515  sl0 -      -
localhost.Exampl localhost.Example. UH          5    42127  lo0 -       0.438
inr-3.Example.ED water.CS.Example.E UH          1        0  sl0 -      -
water.CS.Example localhost.Example. UGH        34 47641234  lo0 -       0.438
(root node)
```

Өмнөх жишээнүүд нь харьцангуй завгүй байгаа системүүдийнх юм. Таны систем дээрх тоонууд нь сүлжээний ачааллаас хамаарч өөр өөр байх болно.

## 28.7.2. SLIP сервер тохируулах нь

Энэхүү баримт нь FreeBSD систем дээр SLIP серверийн үйлчилгээнүүдийг тохируулахад зориулсан зөвлөгөөнүүдийг санал болгодог. Энэ нь юу гэсэн үг вэ гэхээр алсын SLIP клиентүүдийн нэвтрэлтийн үед холболтуудыг автоматаар эхлүүлэхээр болгож таны системийг тохируулна гэсэн үг юм.

### 28.7.2.1. Шаардлагатай зүйлс

Энэ баримт нь өөрийн үндсэн чанарын хувьд хэт техникийн учраас тодорхой мэдлэг шаардах болно. Энэ нь таныг TCP/IP сүлжээний протоколыг тодорхой хэмжээгээр мэддэг бөгөөд ялангуяа сүлжээ болон цэгийн хаяглалт, сүлжээний хаягийн багууд, дэд сүлжээ үүсгэх, чиглүүлэлт болон RIP зэрэг чиглүүлэлтийн протоколуудыг мэддэг гэж тооцдог. SLIP үйлчилгээнүүдийг dial-up сервер дээр тохируулах нь эдгээр ухагдахуунуудын тухай мэдлэгийг шаарддаг бөгөөд хэрэв та эдгээрийг мэдэхгүй бол O'Reilly & Associates, Inc-ийн хэвлэсэн Крэг Хантын *TCP/IP Network Administration* буюу TCP/IP сүлжээний удирдлага номыг юм уу эсвэл TCP/IP протоколуудын тухай Даглас Комерийн номнуудыг уншина уу.

Мөн таныг модемоо суулгаж өөрийн модемоор дамжин нэвтрэх боломжуудыг зөвшөөрөхөөр тохируулсан байгаа гэж үздэг. Хэрэв та үүнд зориулж өөрийн системийг бэлдэж амжаагүй байгаа бол dialup үйлчилгээнүүдийн тохиргооны тухай мэдээллийг Dial-in буюу гаднаас залгах үйлчилгээ хэсгээс үзнэ үү. Та бас цуваа портын төхөөрөмжийн драйверийн тухай мэдээлэл болон модемууд дээр нэвтрэлтүүдийг хүлээн авахаар болгож системийг тохируулах тухай мэдээллийг sio(4) болон ttys(5), gettytab(5), getty(8), & init(8) гарын авлагын хуудаснуудаас шалгахыг хүсэж болох юм. Цуваа портын параметрүүдийг (шууд холбогдсон цуваа интерфэйсүүдийн хувьд `clocal` зэрэг) тохируулах тухай мэдээллийг stty(1) гарын авлагын хуудаснаас үзнэ үү.

### 28.7.2.2. Түргэн дүгнэлт

FreeBSD-г SLIP сервер болгон ашиглах ердийн тохиргоонд дараах маягаар ажилладаг: SLIP сервер нь таны FreeBSD SLIP сервер системийг гадагш залгуулдаг бөгөөд /usr/sbin/sliplogin -ийг тусгай хэрэглэгчийн бүрхүүл болгон ашигладаг тусгай SLIP нэвтрэх ID-аар нэвтэрдэг. `sliplogin` програм нь тусгай хэрэглэгчийн хувьд таарах мөрийг олохын тулд /etc/sliphome/slip.hosts файлыг гүйлгэн үздэг бөгөөд хэрэв олсон бол цуваа шугамыг байгаа SLIP интерфэйс уруу холбож дараа нь SLIP интерфэйсийг тохируулахын тулд /etc/sliphome/slip.login бүрхүүлийн скриптийг ажиллуулдаг.

#### 28.7.2.2.1. SLIP серверийн нэвтрэлтийн жишээ

Жишээ нь хэрэв SLIP хэрэглэгчийн ID нь `Shelmerg` байсан бол /etc/master.passwd файл дахь `Shelmerg`-ийн оруулга иймэрхүү байж болох юм:

```
Shelmerg:password:1964:89::0:0:Guy Helmer -
SLIP:/usr/users/Shelmerg:/usr/sbin/sliplogin
```

`Shelmerg` нэвтрэх үед `sliplogin` нь таарах хэрэглэгчийн ID-тай мөрийг /etc/sliphome/slip.hosts файлаас хайх болно, жишээ нь /etc/sliphome/slip.hosts файлд дараах мөр байж болох юм:

```
Shelmerg        dc-slip sl-helmer       0xffffc00       autocomp
```

sliplogin тушаал нь тэр таарах мөрийг олох бөгөөд цуваа шугамыг дараачийн байгаа SLIP интерфэйс уруу залгаж дараа нь /etc/sliphome/slip.login-ийг иймэрхүүгээр ажиллуулна:

```
/etc/sliphome/slip.login 0 19200 Shelmerg dc-slip sl-helmer 0xffffc00 autocomp
```

Хэрэв бүгд зүгээр болж өнгөрвөл /etc/sliphome/slip.login нь өөрийгөө залгасан (дээрх жишээн дээр slip.login уруу өгөгдсөн жагсаалтан дахь эхний параметр болох SLIP интерфэйс 0) SLIP интерфэйсийн хувьд ifconfig тушаалыг ажиллуулж локал IP хаяг (dc-slip), алсын IP хаяг (sl-helmer), SLIP интерфэйсийн сүлжээний баг (0xffffc00) болон бусад нэмэлт тугнуудыг (autocomp) тохируулдаг. Хэрэв ямар нэг зүйл буруу болвол sliplogin нь ихэвчлэн /var/log/messages уруу бүртгэх syslogd дэмон боломжоор дамжуулан боломжийн мэдээллийн чанартай мэдэгдлүүдийг ихэвчлэн бүртгэдэг. (syslogd(8) болон syslog.conf(5)-ийн гарын авлагын хуудаснуудыг үзнэ үү. /etc/syslog.conf-ийг шалгаж syslogd-ийн юу бүртгэдэг болон хаана бүртгэдгийг харна уу).

#### 28.7.2.3. Цөмийн тохиргоо

FreeBSD-ийн анхдагч цөм (GENERIC) нь SLIP (sl(4)) дэмжлэгтэй цуг ирдэг; тухайлан өөрчилсөн цөмийн хувьд бол та өөрийн цөмийн тохиргооны файлдаа дараах мөрийг нэмэх хэрэгтэй:

```
device   sl
```

Анхдагчаар таны FreeBSD машин пакетуудыг урагш дамжуулахгүй байх болно. Хэрэв та өөрийн SLIP серверийг чиглүүлэгч маягаар ашиглахыг хүсвэл /etc/rc.conf файлыг засварлах хэрэгтэй болох бөгөөд gateway_enable хувьсагчийн тохиргоог YES болгож өөрчлөх хэрэгтэй. Ингэж зааж өгснөөр чиглүүлэлтийн тохиргоог дахин ачаалсны дараа ч тогтмол байлгадаг.

Тохиргоог даруй идэвхтэй болгохын тулд та root хэрэглэгчээр дараах тушаалыг ажиллуулж болно:

```
# service routing start
```

FreeBSD-ийн цөмийг тохируулах талаар бичсэн FreeBSD цөмийг тохируулах нь-с өөрийн цөмийг дахин тохируулах талаар лавлана уу.

#### 28.7.2.4. Sliplogin тохиргоо

Урьд нь дурдаж байсны дагуу /usr/sbin/sliplogin файлын тохиргооны хэсэг болох гурван файл /etc/sliphome санд байдаг (sliplogin-ий талаар мэдээллийг sliplogin(8) гарын авлагын хуудаснаас үзнэ үү). Эдгээрт SLIP хэрэглэгчид болон тэдгээрийн холбоотой IP хаягуудыг тодорхойлдог slip.hosts файл, SLIP интерфэйсийг ихэвчлэн тохируулдаг slip.login файл, болон (нэмэлт) цуваа холболт төгссөн үед slip.login-ий нөлөөллүүдийг буцаадаг slip.logout файл

ордог.

#### 28.7.2.4.1. slip.hosts тохиргоо

/etc/sliphome/slip.hosts нь өөр хоорондоо зайгаар тусгаарлагдсан хамгийн багаар бодоход дөрвөн хэсгээс тогтох мөрүүдээс тогддог:

- SLIP хэрэглэгчийн нэвтрэх ID
- SLIP холбоосын локал хаяг (SLIP серверийн хувьд локал)
- SLIP холбоосын алсын хаяг
- Сүлжээний баг

Локал болон алсын хаягууд нь хостын нэрс ( /etc/nsswitch.conf файлд заасан таны тодорхойлолтуудаас хамаарч /etc/hosts файлаар юм уу эсвэл домэйн нэрний үйлчилгээгээр IP болон танигддаг) байж болох бөгөөд сүлжээний баг нь /etc/networks-аас хийгдэх хайлтаар танигдах нэр байж болно. Жишээ систем дээр /etc/sliphome/slip.hosts нь иймэрхүү харагдана:

```
#
# login local-addr      remote-addr     mask            opt1    opt2
#                                               (normal,compress,noicmp)
#
Shelmerg  dc-slip       sl-helmerg      0xfffffc00      autocomp
```

Мөрийн төгсгөлд нэг буюу хэд хэд тохируулгууд байна:

- `normal` - толгойн шахалт байхгүй
- `compress` - толгойнуудыг шахна
- `autocomp` - алсын тал толгойнуудыг шахахыг зөвшөөрөх бол хийнэ
- `noicmp` - ICMP пакетуудыг хаана (таны урсгалыг дүүргэхийн оронд "ping" пакетууд хаягдах болно)

Таны SLIP холбоосуудын локал болон алсын хаягуудын сонголт та TCP/IP дэд сүлжээг тусгайлан зориулах уу эсвэл өөрийн SLIP сервер дээрээ "proxy ARP"-г (энэ нь "жинхэнэ" proxy ARP биш боловч үүнийг тайлбарлахын тулд энэ хэсэгт энэ ухагдахуун ашиглагдах болно) ашиглах уу гэдгээс хамаарна. Хэрэв та аль аргыг сонгохдоо эсвэл IP хаягуудыг хэрхэн зааж өгөхдөө эргэлзэж байгаа бол SLIP-ийн шаардлагатай зүйлс (Шаардлагатай зүйлс) хэсэгт дурдсан TCP/IP-ийн номнууд ба/эсвэл өөрийн IP сүлжээний удирдагчаас лавлана уу.

Та өөрийн SLIP клиентүүдийн хувьд тусдаа дэд сүлжээг ашиглах бол та өөрийн өгөгдсөн IP сүлжээний дугаараас дэд сүлжээний дугаарыг хуваарилж өөрийн SLIP клиент бүрийн IP дугааруудыг тэр дэд сүлжээнээс зааж өгөх хэрэгтэй болно. Дараа нь өөрийн хамгийн ойрын чиглүүлэгч дээр SLIP серверээрээ дамжуулан SLIP дэд сүлжээ уруу статик чиглүүлэлт тохируулах хэрэгтэй болох болов уу.

Өөрөөр, хэрэв та "proxy ARP" аргыг ашиглах бол та өөрийн SLIP клиентийн IP хаягуудыг өөрийн SLIP серверийн Ethernet дэд сүлжээнээс олгох хэрэгтэй болох бөгөөд та бас SLIP серверийн ARP хүснэгт дэх "proxy ARP" оруулгуудыг удирдахын тулд arp(8)-г ашиглахаар

болгож өөрийн /etc/sliphome/slip.login болон /etc/sliphome/slip.logout скриптүүдийг тохируулах хэрэгтэй болно.

#### 28.7.2.4.2. slip.login тохиргоо

Ердийн /etc/sliphome/slip.login файл иймэрхүү харагдана:

```
#!/bin/sh -
#
#       @(#)slip.login  5.1 (Berkeley) 7/1/90

#
# generic login file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 inet $4 $5 netmask $6
```

Энэ slip.login файл нь ердөө л тохирох SLIP интерфэйсийн хувьд `ifconfig`-г локал ба алсын хаягууд болон SLIP интерфэйсийн сүлжээний багтайгаар ажиллуулдаг.

Хэрэв та "proxy ARP" аргыг ашиглахаар шийдсэн бол (өөрийн SLIP клиентүүдийн хувьд тусдаа дэд сүлжээг ашиглахын оронд) таны /etc/sliphome/slip.login файл иймэрхүү харагдах ёстой:

```
#!/bin/sh -
#
#       @(#)slip.login  5.1 (Berkeley) 7/1/90

#
# generic login file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 inet $4 $5 netmask $6
# Answer ARP requests for the SLIP client with our Ethernet addr
/usr/sbin/arp -s $5 00:11:22:33:44:55 pub
```

Энэ slip.login файл дахь `arp -s $5 00:11:22:33:44:55 pub` тушаал нь SLIP серверийн ARP хүснэгтэд ARP оруулгыг үүсгэдэг. Ethernet дэх өөр IP цэг SLIP клиентийн IP хаягтай ярилцахыг хүсэх үед энэ ARP оруулга нь SLIP серверийг SLIP серверийн Ethernet MAC хаягаар хариулахад хүргэдэг.

Дээрх жишээг ашиглаж байхдаа Ethernet MAC хаягийг (`00:11:22:33:44:55`) өөрийн системийн Ethernet картын MAC хаягаар солисон эсэхээ шалгаарай. Тэгэхгүй бол таны "proxy ARP" чинь мэдээж ажиллахгүй! Та өөрийн SLIP серверийн Ethernet MAC хаягийг `netstat -i` тушаалыг

ажиллуулан гарсан үр дүнгээс нь харан олж болох юм; гарсан үр дүнгийн хоёр дахь мөр иймэрхүү харагдах ёстой:

```
ed0   1500  <Link>0.2.c1.28.5f.4a         191923    0   129457     0   116
```

Энэ нь системийн Ethernet MAC хаяг нь `00:02:c1:28:5f:4a` гэдгийг үзүүлж байгаа юм - `netstat -i` тушаалаар өгөгдсөн Ethernet MAC хаяг дахь цэгүүд нь тодорхойлох хоёр цэгээр солигдох ёстой бөгөөд arp(8)-ийн хүсдэг хэлбэр уруу хаягийг хөрвүүлэхийн тулд ганц оронтой 16-тын тоо бүрийн өмнө тэг нэмэх ёстой. Ашиглалтын тухай бүрэн мэдээллийг arp(8) гарын авлагын хуудаснаас үзнэ үү.

/etc/sliphome/slip.login болон /etc/sliphome/slip.logout файлыг үүсгэж байхдаа "execute" буюу ажиллуулах битийг (өөрөөр хэлбэл `chmod 755 /etc/sliphome/slip.login /etc/sliphome/slip.logout`) тавих ёстой, тэгэхгүй бол `sliplogin` тушаал түүнийг ажиллуулж чадахгүй байх болно.

#### 28.7.2.4.3. slip.logout тохиргоо

/etc/sliphome/slip.logout нь заавал хатуу чангаар хэрэгтэй биш (та "proxy ARP" хийгээгүй л бол) боловч та үүнийг үүсгэхийг хүсвэл энд үндсэн slip.logout скриптийн жишээ байна:

```
#!/bin/sh -
#
#       slip.logout

#
# logout file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 down
```

Хэрэв та "proxy ARP" ашиглаж байгаа бол SLIP клиентийн хувьд ARP оруулгыг устгах /etc/sliphome/slip.logout файлтай байхыг хүсэж болох юм:

```
#!/bin/sh -
#
#       @(#)slip.logout

#
# logout file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 down
```

```
# Quit answering ARP requests for the SLIP client
/usr/sbin/arp -d $5
```

`arp -d $5` тушаал нь ARP оруулгыг устгаж SLIP клиент нэвтрэх үед "proxy ARP"slip.login нэмэгддэг.

/etc/sliphome/slip.logout файлыг үүсгэснийхээ дараа ажиллуулах битийг тавьсан эсэхээ шалгаарай (өөрөөр хэлбэл `chmod 755 /etc/sliphome/slip.logout`).

#### 28.7.2.5. Чиглүүлэлтийн эргэцүүллүүд

Хэрэв та өөрийн SLIP клиентүүд болон таны бусад сүлжээний хоорондох (болон магадгүй Интернэт хоорондох) чиглүүлэлтийн пакетуудын хувьд "proxy ARP" аргыг ашиглахгүй байгаа бол өөрийн SLIP клиентүүдийн дэд сүлжээг өөрийн SLIP серверээр дамжуулан чиглүүлэхийн тулд өөрийн хамгийн ойрын анхдагч чиглүүлэгчид(үүдэд) статик чиглүүлэлтүүдийг нэмэх хэрэгтэй болж болох юм.

##### 28.7.2.5.1. Статик чиглүүлэлтүүд

Өөрийн хамгийн ойрын чиглүүлэгчүүдэд статик чиглүүлэлтүүдийг нэмэх нь яршигтай байж болох юм (эсвэл хэрэв та тэгж хийх зөвшөөрөлгүй бол боломжгүй байх юм...). Та өөрийн байгууллагадаа олон чиглүүлэгчтэй бол Cisco болон Proteon-ий хийсэн зарим чиглүүлэгчүүд нь зөвхөн SLIP дэд сүлжээ уруу статик чиглүүлэлттэй байхаар тохируулагдсан байх шаардлагагүй байж болох бөгөөд бас аль статик чиглүүлэлтүүд нь бусад чиглүүлэгчүүдэд хэлж байгааг хэлж өгөх хэрэгтэй болох юм. Тийм болохоор статик чиглүүлэлт дээр тулгуурласан чиглүүлэлтийг ажиллуулахад зарим нэг чадамж болон алдаа олж засварлах/тааруулах чадвар хэрэгтэй болж болох юм.

# Chapter 29. Цахим Шуудан

## 29.1. Ерөнхий Агуулга

"Цахим Захидал" буюу имэйл гэж нийтэд илүүтэй танигдсан энэ зүйл нь өнөө үед хамгийн өргөн хэрэглэгдэх болсон харилцааны хэлбэр юм. Энэ бүлэгт FreeBSD дээр захидлын серверийг зохион байгуулах анхан шатны танилцуулга болон FreeBSD ашиглан цахим захидал илгээх болон хүлээн авах талаар үзэх болно; гэвч, энэ нь бүрэн хэмжээний танилцуулга болж чадахгүй бөгөөд олон чухал зүйлсийг оруулаагүй болно. Энэ сэдвээр бүрэн дүүрэн мэдээлэл авахыг хүсвэл, Ном зүй хэсэгт байгаа номнуудын жагсаалтыг үзнэ үү.

Энэ бүлгийг уншсаны дараа та дараах зүйлсийг мэдэж авах болно:

- Цахим захидал хүлээн авах болон илгээхэд ямар програмууд оролцдог талаар
- sendmail-н үндсэн тохиргооны файлууд FreeBSD дээр хаана байрладаг талаар
- Алсын болон дотоодын захидлын хайрцгуудын ялгаа
- Спам тараагчдыг өөрийн серверийг хууль бусаар дамжуулагч (relay) болгон ашиглахаас хэрхэн сэргийлэх талаар
- sendmail-н оронд өөр Захидал Дамжуулах Агентийг хэрхэн суулгаж тохируулах талаар
- Захидлын сервертэй холбоотой түгээмэл тохиолддог асуудлуудыг хэрхэн шийдвэрлэх талаар
- SMTP-г UUCP-тэй хэрхэн ашиглах
- Системийг хэрхэн зөвхөн захидал илгээдэг болгож тохируулах талаар
- Утасны үзүүр ашигласан холболтоор захидлыг хэрхэн хэрэглэх талаар
- Аюулгүй байдлыг нэмэгдүүлэхийн тулд SMTP Магадлалтыг хэрхэн нэмэх талаар
- mutt гэх мэт Хэрэглэгчийн Захидлын Агентийг хэрхэн суулгаж тохируулах талаар
- Өөрт ирсэн захидлыг алсын POP эсвэл IMAP серверээс хэрхэн татаж авах талаар
- Гаднаас ирэх захидлыг хэрхэн автоматаар шүүлтүүр болон дүрмүүдээр шалгах талаар

Энэ бүлгийг уншихын өмнө та дараах шаардлагыг хангасан байх ёстой:

- Сүлжээний холболтоо зөв гүйцэтгэсэн байх (Сүлжээний нэмэлт ойлголтууд).
- Захидлын хост дээрээ DNS мэдээллийг зөв оруулж өгсөн байх (Сүлжээний орчны Серверүүд).
- Гуравдагч этгээдийн програмыг хэрхэн суулгах талаар мэдлэгтэй байх (Програм суулгах. Багцууд болон портууд).

## 29.2. Цахим Захидлыг Хэрэглэх нь

Цахим захидал солилцоход дараах үндсэн таван бүрэлдэхүүн оролцоно. Үүнд: хэрэглэгчийн програм, сервер дэмон, DNS, алсын эсвэл дотоод захидлын хайрцаг ба мэдээж захидлын хост өөрөө юм.

### 29.2.1. Хэрэглэгчийн Програм

Захаас нь эхлээд дурдвал эдгээрт mutt, alpine, elm, `mail` зэрэг тушаалын мөрний програмууд, мөн balsa, xfmail зэрэг GUI програмууд, дээр нь WWW хөтөч зэрэг илүү "нарийн төвөгтэй" програмууд ч орно. Дээрх програмууд нь цахим захидлын гүйлгээг дотоод "захидлын хост" -од сервер демонуудын аль нэгийг дуудан шилжүүлэн өгөх, эсвэл TCP-р дамжуулан очих газар нь хүргэх ажлыг гүйцэтгэдэг.

### 29.2.2. Захидлын Хостын Сервер Демон

FreeBSD нь анхдагч байдлаар sendmail-г агуулсан байдаг, гэхдээ бусад олон захидлын серверийн дэмонг дэмждэг. Үүнээс заримыг дурдвал:

- exim;
- postfix;
- qmail.

Сервер дэмон ихэвчлэн хоёр үүрэг гүйцэтгэнэ-гаднаас ирж байгаа захидлыг хүлээн авах болон гадагшаа явж байгаа захидлыг очих газар нь хүргэх үүрэгтэй. Сервер дэмон нь захидлаа уншихын тулд POP эсвэл IMAP зэрэг протоколыг ашиглан захидлыг цуглуулж авах, эсвэл дотоод mbox эсвэл Maildir захидлын хайрцгууд руу холбогдох зэргийн аль алиных нь үүргийг хүлээдэг*гүй*. Үүний тулд нэмэлт дэмон шаардлагатай.

sendmail-н хуучин хувилбаруудад халдагч таны машины дотоод болон алсын хандалтыг гартаа оруулж болох аюулгүй байдлын хувьд хүндхэн асуудлууд гарч байсан. Иймэрхүү асуудал гарахаас сэргийлж хамгийн сүүлийн хувилбарыг ашиглаж байгаа эсэхээ нягтлах шаардлагатай. Нэмэлт сонголт байдлаар, өөр MTA-г FreeBSD Портуудын Цуглуулгаас суулгаарай.

### 29.2.3. Цахим Захидал ба DNS

Домэйн Нэрийн Систем (DNS) ба түүний дэмон `named` нь цахим захидлыг хүргэх ажилд чухал үүрэг гүйцэтгэдэг. Танай байгаа газраас өөр тийш захидал хүргэхийн тулд, сервер дэмон очих газрын хаягийг өөрөөр хэлбэл захидал хүлээж авах хостыг DNS-с асууна. Энэ үйл явц нөгөө талаас танд захидал явуулахад мөн давтагдана.

DNS нь хост нэрийг IP хаягт буулгах үүрэгтэйгээс гадна, MX бичлэгүүд гэгддэг захидал хүргэхтэй холбоотой тусгай мэдээллийг агуулж байдаг. MX (Mail eXchanger буюу Захидал Солилцогч) бичлэг нь ямар хост, эсвэл хостууд, тухайн домэйны хаяглагдсан захидлыг хүлээж авах ёстойг заана. Хэрэв тухайн домэйны хувьд MX бичлэг байхгүй бол, захидлуудыг тухайн домэйноор нэрлэгдсэн хостын A бичлэгт харгалзах хост руу шууд дамжуулна.

Ямар ч домэйны хувьд MX бичлэгүүдийг host(1) тушаалыг ашиглан үзэж болно. Жишээг дор үзүүлэв:

```
% host -t mx FreeBSD.org
FreeBSD.org mail is handled (pri=10) by mx1.FreeBSD.org
```

### 29.2.4. Захидал хүлээн авах нь

Захидал хүлээж авах ажлыг захидлын хост гүйцэтгэнэ. Энэ хост таны домэйнд хаяглагдсан бүх захидлыг хүлээж авах ба түүнийг таны тохиргооноос хамаарч эсвэл mbox (захидал хадгалах анхдагч арга зам) эсвэл Maildir хэлбэрээр хадгална. Захидлыг хадгалсан л бол, түүнийг дотоодоос mail(1) эсвэл mutt зэрэг програмыг ашиглан уншиж болно. Эсвэл алсаас хандан, POP эсвэл IMAP зэрэг протоколыг ашиглан цуглуулж авсны дараа уншиж болно. Энэ нь хэрэв та захидлаа зөвхөн дотоодоос шалгах хүсэлтэй байгаа бол, POP эсвэл IMAP сервер суулгах шаардлагагүй гэсэн үг юм.

#### 29.2.4.1. POP ба IMAP ашиглан алсын захидлын хайрцагт хандах нь

Захидлын хайрцагт алсаас хандахын тулд, та POP эсвэл IMAP сервер руу хандах хэрэгтэй болно. Эдгээр протоколууд нь хэрэглэгчдэд өөрсдийн захидлын хайрцагт алсаас төвөггүйгээр хандах боломжийг олгодог. Хэдийгээр POP ба IMAP хоёул хэрэглэгчдэд захидлын хайрцагтаа алсаас хандах боломжийг олгодог боловч, IMAP нь илүү давуу талтай, дурдвал:

- IMAP нь алсын серверээс захидлыг зөөж авчрахаас гадна мөн хадгалж чадна.
- IMAP нь зэрэг хийх өөрчлөлтийг дэмждэг.
- IMAP нь захидлыг бүтнээр нь биш, зөвхөн түүний төрхийг нь татаж авах боломжийг хэрэглэгчдэд олгодог тул, хурд удаантай холболттой үед үнэхээр хэрэгтэй байж чаддаг. Мөн сервер болон харилцагчийн хооронд дамжих өгөгдлийг хамгийн бага байлгахын тулд, сервер дээр хайлт явуулах зэрэг үйлдлийг хийж чаддаг.

POP эсвэл IMAP сервер суулгахын тулд, дараах алхмуудыг хийх хэрэгтэй:

1. Өөрийнхөө хэрэгцээ, шаардлагад тохирсон IMAP эсвэл POP серверийг сонгож авах. Өргөн хэрэглэгддэг зарим POP ба IMAP серверүүдийг жишээ болгон дурдвал:
   - qpopper;
   - teapop;
   - imap-uw;
   - courier-imap;
   - dovecot;
2. Өөрийн сонгосон POP эсвэл IMAP дэмонг портуудын цуглуулгаас суулгах.
3. Шаардлагатай бол, POP эсвэл IMAP серверийг ачаалахын тулд /etc/inetd.conf файлыг засварлах.

POP ба IMAP хоёул хэрэглэгчийн нэр, нэвтрэх үг зэрэг чухал мэдээллийг цэвэр текст чигээр нь дамжуулдаг болохыг энд тэмдэглэх хэрэгтэй болов уу. Энэ нь эдгээр протоколоор мэдээлэл дамжуулалтыг аюулгүй болгоё гэвэл сессүүдээ ssh(1) хонгилоор эсвэл SSL ашиглан дамжуулах хэрэгтэй гэсэн үг юм. Сессүүдийг хонгилоор дамжуулах талаар SSH туннель хийх хэсэгт болон SSL-ийн талаар OpenSSL хэсэгт тайлбарласан байгаа.

#### 29.2.4.2. Дотоод захидлын хайрцгуудад хандах нь

Захидлын хайрцаг байрлаж байгаа сервер дээрх MUA-уудын тусламжтай захидлын хайрцгуудад дотоодоос нь шууд хандаж болно. Үүний тулд mutt эсвэл mail(1) зэрэг програмуудыг ашиглана.

### 29.2.5. Захидлын хост

Захидлын хост гэж таны хост, магадгүй таны сүлжээнд хаяглагдсан захидлыг хүлээн авах, илгээх үүргийг гүйцэтгэж байгаа серверийг нэрлэнэ.

## 29.3. sendmail-г Тохируулах нь

FreeBSD-н хувьд sendmail(8) бол анхдагч Mail Transfer Agent буюу Захидал Дамжуулах Агент(MTA) юм. sendmail-н үүрэг бол Mail User Agent буюу Хэрэглэгчийн Захидлын Агент(MUA)-с захидал хүлээн авч, тохиргооны файлд хэрхэн зааж өгснөөс хамааран зохих захидал түгээгчид дамжуулна. sendmail нь мөн сүлжээний холболтуудыг хүлээн авч, захидлыг дотоод захидлын хайрцагт эсвэл өөр програмд дамжуулах чадвартай.

sendmail дараах тохиргооны файлуудыг ашигладаг:

| Файлын Нэр | Үүрэг |
|---|---|
| /etc/mail/access | sendmail-н хандалтын өгөгдлийн сангийн файл |
| /etc/mail/aliases | Захидлын хайрцагны хуурамч дүрүүд |
| /etc/mail/local-host-names | sendmail эдгээр хостуудаас захидал хүлээж авна. |
| /etc/mail/mailer.conf | Захидал түгээгч програмын тохиргоо |
| /etc/mail/mailertable | Захидал түгээгчийн түгээлтийн хүснэгт |
| /etc/mail/sendmail.cf | sendmail-н гол тохиргооны файл |
| /etc/mail/virtusertable | Хийсвэр хэрэглэгчид болон домэйн хүснэгтүүд |

### 29.3.1. /etc/mail/access

Хандалтын өгөгдлийн санд ямар хостууд болон IP хаягуудаас дотоод захидлын серверт хандах, ямар төрлийн хандалт хийхийг зааж өгсөн байна. Хостуудыг `OK`, `REJECT`, `RELAY` гэж тэмдэглэж болно эсвэл sendmail-н алдаа зохицуулах дэгд тодорхой алдааг зааж өгөн дамжуулж болно. `OK` гэсэн анхдагч утгаар тэмдэглэгдсэн хостууд энэ хостод хаяглагдсан захидлыг дамжуулах эрхтэй. `REJECT` гэж тэмдэглэгдсэн хостуудаас үүсэх холболтыг хүлээж авахгүй. `RELAY` гэж тэмдэглэгдсэн хостуудын хувьд энэ захидлын серверээр дамжуулан хааш нь ч хамаагүй захидал илгээж болно.

*Жишээ 33. sendmail Хандалтын Өгөгдлийн санг Тохируулах нь*

```
cyberspammer.com                550 We do not accept mail from spammers
FREE.STEALTH.MAILER@            550 We do not accept mail from spammers
another.source.of.spam          REJECT
okay.cyberspammer.com           OK
128.32                          RELAY
```

Энэ жишээн дээр бид таван бүртгэл харж байна. Захидал илгээгчийн хувьд хүснэгтийн зүүн баганад байгаа утгатай дүйвэл баруун баганад байгаа үйлдлийг гүйцэтгэнэ. Эхний хоёр жишээний хувьд sendmail-н алдаа зохицуулах дэгд алдааны мэдээллийг өгөх болно. Хүснэгтийн зүүн баганад байгаа утгатай захидал дүйсэн тохиолдолд алсын хостод энэ мэдээллийг өгөх болно. Дараагийн жишээн дээр, `another.source.of.spam` гэсэн тодорхой хостоос ирэх захидлыг бүгдийг буцаах болно. Түүний дараагийн жишээн дээр, `okay.cyberspammer.com` гэсэн хостоос ирэх холболтуудыг хүлээж авах болно. Энд хостын нэр өмнөх мөрний `cyberspammer.com`-г бодвол илүү тодорхой байна. Илүү тодорхой заалт нь ерөнхий заалтыг дарж хүчингүй болгоно. Хамгийн сүүлийн жишээн дээр, `128.32`-р эхэлсэн IP хаягтай хостуудаас ирэх захидлыг өртөөлөн дамжуулахыг зааж байна. Эдгээр хостууд бусад серверт хаягласан захидлыг энэ захидлын серверээр дамжуулах боломжтой гэсэн үг юм.

Энэ файлд өөрчлөлт оруулсны дараа, /etc/mail/ дотор `make` тушаалыг өгөн өгөгдлийн санг шинэчлэх хэрэгтэй.

### 29.3.2. /etc/mail/aliases

Хуурамч дүрүүдийн өгөгдлийн санд хэрэглэгчид, файлууд, програмууд эсвэл бусад хуурамч дүрүүд рүү задардаг хийсвэр захидлын хайрцгуудын жагсаалтыг хадгалж байдаг. /etc/mail/aliases дотор хэрэглэж болох хэдэн жишээг дор үзүүлэв:

*Жишээ 34. Захидлын Хуурамч дүрүүд*

```
root: localuser
ftp-bugs: joe,eric,paul
bit.bucket:  /dev/null
procmail: "|/usr/local/bin/procmail"
```

Бичих хэлбэр нь амархан; тодорхойлох хоёр цэгний зүүн талд байгаа захидлын хайрцагны нэр нь баруун талд байгаа зүйл болон задарна. Эхний жишээн дээр `root` захидлын хайрцгийг `localuser` захидлын хайрцаг болгон задалж байна. Энэ нэрийг цааш дахин хуурамч дүрийн өгөгдлийн сангаас хайх болно. Хэрэв энэ нэр олдохгүй бол, захидлыг `localuser` нэртэй дотоод хэрэглэгчид хүргэх болно. Дараагийн жишээн дээр захидлын жагсаалтыг үзүүлсэн байна. `ftp-bugs` захидлын хайрцагт ирэх захидлыг `joe`, `eric`, ба `paul` гэсэн гурван дотоод хэрэглэгчдэд задлах болно. Алсын захидлын хайрцгийг user@example.com хэлбэрээр оруулж болохыг энд тэмдэглэх хэрэгтэй. Дараагийн жишээн дээр захидлыг файлд, энэ жишээн дээр /dev/null-д бичихийг зааж байна. Хамгийн сүүлийн жишээн дээр, захидлын програмд илгээхийг, энэ

тохиолдолд захидлыг UNIX® хоолойгоор дамжуулан /usr/local/bin/procmail-н стандарт оролтод өгч байна.

Энэ файлд өөрчлөлт оруулсны дараа, /etc/mail/ дотор `make` тушаалыг өгөн өгөгдлийн санг шинэчлэх хэрэгтэй.

### 29.3.3. /etc/mail/local-host-names

Энэ бол sendmail(8) дотоод хостууд байдлаар хүлээн авах хостуудын нэрийн жагсаалт юм. sendmail захидлыг нь хүлээж авах домэйн болон хостын нэрийг энэ файлд бичих хэрэгтэй. Жишээлбэл, хэрэв энэ захидлын сервер `example.com` домэйн болон `mail.example.com` хостын захидлыг хүлээж авах ёстой бол, local-host-names файл дараах байдалтай байна:

```
example.com
mail.example.com
```

Энэ файлд өөрчлөлт ороход, өөрчлөлтүүдийг хүчинтэй болгохын тулд sendmail(8)-г дахин эхлүүлэх шаардлагатай.

### 29.3.4. /etc/mail/sendmail.cf

sendmail-н гол тохиргооны файл болох sendmail.cf нь захидлын хаягийг дахин бичихээс эхлүүлээд алсын захидлын серверт таттагалзсан зурвас илгээх хүртэл бүх зүйлийг оруулаад sendmail-н ерөнхий үйл ажиллагааг хянана. Мэдээжийн хэрэг, ийм олон талын үүрэгтэй энэ тохиргооны файл нилээд ойлгомжгүй байх ба нарийвчлан авч үзэх нь энэ бүлгийн мэдлээс хальсан асуудал болно. Нэг баярлууштай нь, стандарт захидлын сервер байдлаар ашиглахад энэ файлд өөрчлөлт оруулах бараг шаардлагагүй байдаг.

sendmail-н гол тохиргооны файлыг sendmail-н шинж чанар, төлвийг илэрхийлсэн m4(1) макронуудаас үүсгэж болно. Дэлгэрэнгүйг /usr/src/contrib/sendmail/cf/README файлаас үзэж болно.

Энэ файлд өөрчлөлт ороход, өөрчлөлтүүдийг хүчинтэй болгохын тулд sendmail-г дахин эхлүүлэх шаардлагатай.

### 29.3.5. /etc/mail/virtusertable

virtusertable нь хийсвэр домэйнууд ба захидлын хайрцгийн хаягуудыг жинхэнэ захидлын хайрцгуудад буулгана. Эдгээр захидлын хайрцгууд нь дотоод, алсын, /etc/mail/aliases дотор тодорхойлогдсон хуурамч дүр эсвэл файл байж болно.

*Жишээ 35. Хийсвэр Домэйны Захидлын Буулгалтын Жишээ*

```
root@example.com                root
postmaster@example.com          postmaster@noc.example.net
@example.com                    joe
```

Дээрх жишээн дээр, example.com домэйны хувь дахь буулгалтыг үзүүлсэн байна. Энэ файлыг дээрээс нь доош нь, эхний таарсныг авах байдлаар боловсруулна. Эхний мөрөнд root@example.com-г root гэсэн дотоод захидлын хайрцагт буулгаж байна. Дараагийн мөрөнд postmaster@example.com-г noc.example.net хост дээр байгаа postmaster захидлын хайрцагт буулгаж байна. Эцэст нь, хэрэв example.com-той хаяг дээр дурдсаны алинд ч тохироогүй бол, хамгийн сүүлийн мөрөнд тохирох болно. Энэ мөрөнд example.com домэйнтой бүх захидлын хаяг тохирох бөгөөд бүгд joe гэсэн дотоод захидлын хайрцагт буулгагдах болно.

# 29.4. Өөрийн Захидал Дамжуулах Агентийг Солих нь

Урьд нь дурдсан шиг, FreeBSD дотор sendmail нь MTA (Mail Transfer Agent буюу Захидал Дамжуулах Агент) байдлаар аль хэдийн суучихсан ирдэг. Тиймээс анхдагч байдлаар, таны явуулж байгаа болон ирж байгаа захидлыг хариуцан ажиллаж байгаа.

Гэвч янз бүрийн шалтгаанаар зарим системийн администраторууд системийнхээ MTA-г солих хүсэлтэй байдаг. Шалтгаан нь өөр MTA-г хэрэглэж үзэх хүсэлдээ хөтлөгдсөнөөс, эсвэл бусад захидал дамжуулагчдаас улбаалсан тусгай чадвар эсвэл багц шаардлагатай болох зэрэг янз бүр байдаг. Баярлууштай нь, ямар ч шалтгаантай байсан гэсэн FreeBSD нь өөрчлөлт хийхэд амархан байх боломжийг олгодог.

## 29.4.1. Шинэ MTA суулгах нь

Танд олон төрлийн MTA-н сонголт байна. Өргөн сонголттой FreeBSD Портуудын Цуглуулгаас эхлэх нь хамгийн зөв эхлэл болно. Мэдээж FreeBSD дээр ажиллуулж чадаж л байвал, та хүссэн газраасаа хүссэн MTA-аа суулгах эрхтэй.

Эхлээд шинэ MTA-аа суулгаж эхлэх хэрэгтэй. Нэгэнт суулгасны дараа, шинэ програм таны хэрэгцээг үнэхээр хангаж чадаж байна уу гэдгийг шалгах, sendmail-г бүрэн солихоос өмнө шинэ програмын тохиргоог хийх боломжтой. Ингэх явцад таны суулгаж байгаа шинэ програм /usr/bin/sendmail зэрэг системийн хоёртын файлуудыг дарж бичихгүй байхыг анхаарч байх хэрэгтэй. Үгүй бол, таныг тохиргоо хийхээс чинь өмнө захидлын шинэ програм чинь ажиллаж эхлэх болно.

Сонгож авсан MTA програмынхаа тохиргоог хийхдээ тухайн програмын өөрийн холбогдох баримт бичгүүдийг үзнэ үү.

## 29.4.2. sendmail-г идэвхгүй болгох нь

Хэрэв та sendmail-н гадагш захидал илгээх үйлчилгээг идэвхгүй болгосон бол, өөр захидал түгээх системээр түүнийг орлуулах нь чухал юм. Хэрэв тэгэхгүй бол periodic(8) гэх мэт системийн функцууд өөрсдийн үр дүнгээ цахим захидлаар илгээх боломжгүй болж, хэвийн үйл ажиллагаа нь доголдох болно. Системийн нилээд хэсэгт sendmail ашигладаг байхаар бодолцсон байгаа. Хэрэв sendmail-г идэвхгүй болгосны дараа програмууд sendmail-н хоёртын файлыг ашиглан захидлыг үргэлжлүүлэн илгээсээр байвал, тэдгээр захидлууд sendmail-н идэвхгүй дараалалд орж, хэзээ ч хүрэх

> газраа очиж чадахгүй болно.

sendmail-г бүхэлд нь идэвхгүй болгохын тулд, гадагш захидал илгээх үйлчилгээг оруулаад, дараах мөрүүдийг:

```
sendmail_enable="NO"
sendmail_submit_enable="NO"
sendmail_outbound_enable="NO"
sendmail_msp_queue_enable="NO"
```

/etc/rc.conf файлд оруулж өгөх шаардлагатай.

Хэрэв та sendmail-н зөвхөн захидал хүлээн авах үйлчилгээг идэвхгүй болгох хүсэлтэй бол, дараах мөрийг:

```
sendmail_enable="NO"
```

/etc/rc.conf файлд оруулж өгөх шаардлагатай. sendmail-н эхлэл тохируулгуудын талаар дэлгэрэнгүй мэдээллийг rc.sendmail(8) заавар хуудаснаас үзнэ үү.

### 29.4.3. Өөрийн шинэ MTA-г систем ачаалах үед ажиллуулах нь

postfix-д зориулсан дараах жишээн дээрх шиг тохиргооны мөрийг /etc/rc.conf файлд нэмэн шинэ MTA-г ачаалах үед эхлүүлж болно.

```
# echo 'postfix_enable=«YES»' >> /etc/rc.conf
```

MTA нь одоо автоматаар ачаалах үед эхлэх болно.

### 29.4.4. sendmail-г Системийн Анхдагч Захидал Түгээгч болгох нь

sendmail нь UNIX® системүүдийн стандарт програм байхаар хаа сайгүй хэрэглэгддэг болохоор, зарим програм түүнийг системд аль хэдийн суучихсан, ажиллаж байгаа гэж тооцсон байдаг. Тиймээс, бусад олон MTA-ууд sendmail-н тушаалын мөрний интерфэйстэй нийцтэй шийдлүүдийг гаргадаг бөгөөд энэ нь sendmail-н оронд тэдгээр MTA-уудыг ашиглах боломжтой болгодог.

Тиймээс, хэрэв өөр захидал түгээгч хэрэглэж байгаа бол, програмууд sendmail-н /usr/bin/sendmail гэх зэрэг хоёртын файлын оронд таны сонгосон захидал түгээгчийг хэрэглэж байгаа эсэхийг нягтлаарай. Аз болоход, FreeBSD танд энэ ажлыг хийж өгөх mailwrapper(8) гэсэн системийг агуулж байдаг.

sendmail нь анх суусан байдлаараа ажиллаж байх үед, /etc/mail/mailer.conf файл дотор дараах мөрүүдийг харах болно:

```
sendmail    /usr/libexec/sendmail/sendmail
```

```
send-mail   /usr/libexec/sendmail/sendmail
mailq       /usr/libexec/sendmail/sendmail
newaliases  /usr/libexec/sendmail/sendmail
hoststat    /usr/libexec/sendmail/sendmail
purgestat   /usr/libexec/sendmail/sendmail
```

Энэ нь, эдгээр өргөн хэрэглэгддэг тушаалуудын аль нэгийг (sendmail өөрөө гэх мэт) ажиллуулахад, систем үнэн хэрэгтээ sendmail нэртэй захидлын дундын хянагчийн хуулбарыг дуудна гэсэн үг юм. Энэ захидлын дундын хянагч mailer.conf файлыг шалгаж, /usr/libexec/sendmail/sendmail-г оронд нь ажиллуулна. Энэ системийн тусламжтай дээрх анхдагч sendmail функцүүдийг дуудахад үнэндээ ажиллах хоёртын файлуудыг солиход амархан болгодог.

Тиймээс хэрэв та sendmail-н оронд /usr/local/supermailer/bin/sendmail-compat-г ажиллуулах хүсэлтэй бол, /etc/mail/mailer.conf файлыг дараах байдлаар засаж бичих хэрэгтэй:

```
sendmail     /usr/local/supermailer/bin/sendmail-compat
send-mail   /usr/local/supermailer/bin/sendmail-compat
mailq       /usr/local/supermailer/bin/mailq-compat
newaliases  /usr/local/supermailer/bin/newaliases-compat
hoststat    /usr/local/supermailer/bin/hoststat-compat
purgestat   /usr/local/supermailer/bin/purgestat-compat
```

### 29.4.5. Төгсгөл

Өөрийн хүссэнээр бүх зүйлсийг тохируулсны дараа, нэгэнт хэрэггүй болсон sendmail процессуудыг зогсоож шинэ програмд хамаарах процессуудыг эхлүүлэх, эсвэл зүгээр л системийг дахин ачаалах хэрэгтэй. Системийг дахин ачаалах нь систем ачаалах явцад таны шинэ MTA-г ачаалах тохиргоог зөв хийсэн эсэхээ шалгахад тань тусална.

## 29.5. Гэмтэл саатлыг арилгах нь

### 29.5.1. Яагаад би өөрийн сүлжээнд байгаа хостуудын хувьд FQDN-г хэрэглэх ёстой вэ?

Үнэндээ тухайн хост өөр домэйнд байгаа болохыг та анзаарах байх; жишээ нь, хэрэв та өөрөө `foo.bar.edu` домэйнд байгаа бөгөөд `bar.edu` домэйнд байгаа `mumble` хост руу хандах хүсэлтэй бол түүнийг зүгээр `mumble` гэхийн оронд `mumble.bar.edu` гэсэн fully-qualified domain name буюу бүрэн-баталгаажсан домэйн нэрээр нь заах хэрэгтэй.

Урьд нь, BSD BIND тайлагчийн хувьд дээрх шиг хэрэглэж болдог байсан. Гэвч FreeBSD-тэй хамт ирдэг BIND-н одоогийн хувилбар дээр өөрийн оршин байгаа домэйноос бусад бүрэн баталгаажаагүй домэйн нэрийн хувьд анхдагч товчилсон нэрүүдийг хэрэглэж болохгүй болсон. Тиймээс баталгаажаагүй `mumble` гэсэн хостыг эсвэл `mumble.foo.bar.edu` гэж ойлгоно, эсвэл эх домэйнд хайх болно.

Хайлт цааш `mumble.bar.edu`, ба `mumble.edu` болж үргэлжлэх бөгөөд энэ нь санаснаас шал өөр

зүйл болон хувирна. Үүнийг яагаад буруу үйлдэл, бүр аюулгүй байдлын цоорхой гэж тооцох болсныг RFC 1535-с олж үзээрэй.

Дээрх байдлаас гарахын тулд, өөрийн /etc/resolv.conf файл дотор:

```
search foo.bar.edu bar.edu
```

гэсэн мөрийг:

```
domain foo.bar.edu
```

мөрийн оронд бичих хэрэгтэй. Гэхдээ, хайлт хийх дэс дараалал нь RFC 1535-д заасны дагуу "дотоод болон гадаад удирдлагын хязгаар"ыг давах ёсгүйг анхаараарай.

### 29.5.2. sendmail захидал өөр дээр минь тойроод ирж байна гээд байна

Энэ асуултанд sendmail-н FAQ дараах байдалтай хариулсан байгаа:

```
I'm getting these error messages:

553 MX list for domain.net points back to relay.domain.net
554 <user@domain.net>... Local configuration error

How can I solve this problem?

You have asked mail to the domain (e.g., domain.net) to be
forwarded to a specific host (in this case, relay.domain.net)
by using an MX record, but the relay machine does not recognize
itself as domain.net. Add domain.net to /etc/mail/local-host-names
[known as /etc/sendmail.cw prior to version 8.10]
(if you are using FEATURE(use_cw_file)) or add Cw domain.net
to /etc/mail/sendmail.cf.
```

sendmail-н FAQ-г http://www.sendmail.org/faq/ хаягаар орж үзэж болох ба хэрэв та захидлын серверийнхээ тохиргоогоор нилээд "оролдох" бол эхлээд үүнийг уншихыг зөвлөж байна.

### 29.5.3. Утсаар залгаж холбогддог PPP хост дээр захидлын серверийг хэрхэн ажиллуулах вэ?

Та LAN-д байгаа FreeBSD машиныг Интернэтэд холбох хүсэлтэй байна. Энэ FreeBSD машин тухайн LAN-ий хувьд захидлын гарц байх болно. PPP холболт бол тусгайлан бий болгосон зориулагдсан холболт биш юм.

Үүний тулд хоёр арга байна. Нэг арга нь UUCP ашиглах юм.

Нөгөө арга нь таны домэйны хувьд хоёрдогч MX үйлчилгээг үзүүлэх байнгын холболттой Интернэт сервертэй болох явдал юм. Жишээлбэл, хэрэв танай байгууллагын домэйн нэр

example.com бөгөөд танай Интернэтийн үйлчилгээ үзүүлэгч танай домэйны хувьд хоёрдогч MX үйлчилгээг үзүүлэхээр example.net домэйныг дараах байдалтай тохируулжээ:

```
example.com.          MX        10      example.com.
                      MX        20      example.net.
```

Зөвхөн ганц хост эцсийн захидал хүлээн авагч байх ёстой (example.com хостын /etc/mail/sendmail.cf файлд Cw example.com-г нэмнэ).

Захиа явуулах гэж байгаа sendmail захидлыг явуулахаар оролдох үед эхлээд модем холбоосоор тантай (example.com) холбогдохыг оролдоно. Та Интернэтэд холбогдоогүй байгаа учраас ихэнх тохиолдолд тантай холбогдож чадахгүй. Энэ тохиолдолд, sendmail автоматаар хоёрдогч MX сервер рүү, өөрөөр хэлбэл танай Интернэтийн үйлчилгээ үзүүлэгчид (example.net) илгээх болно. Хоёрдогч MX сервер танай сервертэй холбогдохыг тогтмол оролдох ба ийм замаар анхдагч MX хост руу (example.com) захидлыг хүргэх болно.

Та дор үзүүлсэнтэй төстэй нэвтрэх скриптийг хэрэглэхийг хүсэж болзошгүй:

```
#!/bin/sh
# Put me in /usr/local/bin/pppmyisp
( sleep 60 ; /usr/sbin/sendmail -q ) &
/usr/sbin/ppp -direct pppmyisp
```

Хэрэв та хэрэглэгчийн хувьд тусдаа нэвтрэх скрипт үүсгэх гэж байгаа бол дээрх скрипт дээр байгаагийн оронд sendmail -qRexample.com-г хэрэглэж болох юм. Энэ нь таны захидлын дараалалд байгаа example.com-д очих бүх захидлыг тэр дор нь боловсруулахыг шаардана.

Байдлыг хэрхэн цааш сайжруулахыг дор үзнэ үү:

FreeBSD Интернетийн үйлчилгээ үзүүлэгчийн захидлын жагсаалт-с хулгайлсан зурвас.

```
> we provide the secondary MX for a customer. The customer connects to
> our services several times a day automatically to get the mails to
> his primary MX (We do not call his site when a mail for his domains
> arrived). Our sendmail sends the mailqueue every 30 minutes. At the
> moment he has to stay 30 minutes online to be sure that all mail is
> gone to the primary MX.
>
> Is there a command that would initiate sendmail to send all the mails
> now? The user has not root-privileges on our machine of course.

In the privacy flags section of sendmail.cf, there is a
definition Opgoaway,restrictqrun

Remove restrictqrun to allow non-root users to start the queue processing.
You might also like to rearrange the MXs. We are the 1st MX for our
customers like this, and we have defined:
```

```
# If we are the best MX for a host, try directly instead of generating
# local config error.
OwTrue

That way a remote site will deliver straight to you, without trying
the customer connection.  You then send to your customer.  Only works for
hosts, so you need to get your customer to name their mail
machine customer.com as well as
hostname.customer.com in the DNS.  Just put an A record in
the DNS for customer.com.
```

### 29.5.4. Бусад хостоос захидал явуулахад яагаад Relaying Denied гэсэн алдаа байнга гараад байна вэ?

FreeBSD-н анхдагч суулгацад sendmail-г зөвхөн ажиллаж байгаа хостоосоо захидал илгээдэг байхаар тохируулсан байдаг. Жишээ нь, хэрэв нэг POP сервер байгаа бол, хэрэглэгчид сургууль, ажил, эсвэл бусад алсын байршлаас захидлаа шалгах боломжтой байх болно. Гэвч гаднаас захидал илгээж чадахгүй хэвээр байх болно. Ихэвчлэн, оролдого хийснээс хэдэн хормын дараа, MAILER-DAEMON `5.7 Relaying Denied` гэсэн алдааны мэдээлэлтэй захидлыг илгээх болно.

Энэ байдлаас гарах хэд хэдэн зам байна. Хамгийн энгийн арга бол өөрийн ISP-н хаягийг /etc/mail/relay-domains-д байгаа өртөөлөх (дамжуулах) домэйнуудын файлд бичих хэрэгтэй. Үүнийг хийх амархан аргыг үзүүлэв:

```
# echo "your.isp.example.com" > /etc/mail/relay-domains
```

Энэ файлыг үүсгэсэн эсвэл зассаны дараа, sendmail-г дахин эхлүүлэх хэрэгтэй. Хэрэв та сервер хариуцагч байгаад, захидлыг дотооддоо явуулах хүсэлгүй, эсвэл өөр машин дээрээс эсвэл бүр өөр ISP-с хулганы товч дараад л хийчих хялбар систем ашиглах хүсэлтэй байгаа бол энэ арга яг тохирно. Энэ арга ганцхан эсвэл хоёрхон захидлын бүртгэл үүсгэсэн тохиолдолд мөн тус болно. Хэрэв дээрх файлд олон тооны хаяг нэмэх шаардлагатай бол, зүгээр өөрийн дуртай засварлагч програмаар нээгээд домэйнуудаа мөр болгонд нэг байхаар нэмж бичих хэрэгтэй:

```
your.isp.example.com
other.isp.example.net
users-isp.example.org
www.example.org
```

Одоо, энэ жагсаалтад (хэрэглэгч таны систем дээр бүртгэлтэй гэсэн нөхцөлтэйгөөр) байгаа ямар ч хостоос, таны системээр дамжуулан явуулсан захидал амжилттай явах болно. Энэ бол хэрэглэгчиддээ таны системээр дамжуулан SPAM явуулах боломжийг олголгүйгээр, таны системээс алсаас захидал илгээх боломжийг олгох үнэхээр сайн арга юм.

# 29.6. Дээд түвшний сэдвүүд

Энэ хэсэг захидлын серверийн тохиргоо, бүхэл домэйнд зориулсан захидлын серверийг зохион байгуулах зэрэг илүү гүнзгий сэдвүүдийг хамрах болно.

## 29.6.1. Үндсэн Тохиргоо

Хэрэв та өөрийн нэрийн сервер ажиллуулж байгаа эсвэл /etc/resolv.conf-г тохируулсан л бол гадаад хостууд руу захидал илгээж чадах ёстой. Хэрэв та өөрийн хостод хаягласан захидлыг таны өөрийн FreeBSD машин дээрх MTA (жишээ нь sendmail) хүлээж авдаг болгохыг хүсэж байгаа бол, дараах хоёр арга зам байна:

- Өөрөө нэрийн сервер ажиллуулаад, өөрийн домэйн нэр авах. Жишээлбэл, `FreeBSD.org`
- Захидлыг шууд өөрийн хост дээр хүлээж авдаг байх. Үүний тулд өөрийн машины одоогийн DNS нэрээр захидлыг хүлээн авах. Жишээлбэл, `example.FreeBSD.org`.

Дээрхээс алийг нь ч сонгосон бай, өөр дээрээ захидал шууд хүлээн авдаг байхын тулд байнгын статик IP хаягтай байх ёстой (ихэнх PPP залгадаг холболтынх шиг динамик хаяг биш). Хэрэв галт ханын цаана байгаа бол, галт хана тань руу чиглэсэн SMTP урсгалыг нэвтрүүлэх ёстой. Захидлыг шууд өөрийн хост дээр хүлээн авахын тулд, дараах хоёр нөхцөлийн аль нэгийг хангасан байх ёстой:

- Таны DNS дээрх (хамгийн бага дугаартай) MX бичлэг таны хостын IP хаяг руу заасан байх ёстой.
- Таны DNS дээр таны хостын IP хаяг руу заасан MX бичлэг байх ёсгүй.

Дээр дурдсаны аль нэгийг хангасан байхад, та өөрийн хост дээр захидал хүлээн авах боломжтой болно.

Туршаад үзээрэй:

```
# hostname
example.FreeBSD.org
# host example.FreeBSD.org
example.FreeBSD.org has address 204.216.27.XX
```

Хэрэв дээрхтэй адил хариу гарч байвал, yourlogin@example.FreeBSD.org хаяг руу илгээсэн захидал ямар ч асуудалгүй хүрэх газраа очих ёстой (`example.FreeBSD.org` дээр sendmail зөв ажиллаж байгаа гэж үзэв).

Үүний оронд доор дурдсантай адил хариу авбал:

```
# host example.FreeBSD.org
example.FreeBSD.org has address 204.216.27.XX
example.FreeBSD.org mail is handled (pri=10) by hub.FreeBSD.org
```

таны хост руу илгээсэн бүх захидлууд (`example.FreeBSD.org`) шууд таны хост руу илгээгдэхийн

оронд hub дээр адилхан хэрэглэгчийн нэр дээр цугларах болно.

Дээрх мэдээллийг DNS сервер зохицуулан ажиллана. DNS-н *M*ail e*X*change буюу MX бичлэгт захидлыг хааш нь чиглүүлэх талаар мэдээлэл агуулагддаг. Хэрэв ямар ч MX бичлэг байхгүй бол, захидлыг IP хаягаар нь хостод дамжуулна.

freefall.FreeBSD.org-н MX бичлэг дараах байдалтай байна:

```
freefall        MX  30  mail.crl.net
freefall        MX  40  agora.rdrop.com
freefall        MX  10  freefall.FreeBSD.org
freefall        MX  20  who.cdrom.com
```

Таны харж байгаачлан, freefall-н хувьд олон MX бичлэг байна. Хамгийн бага MX дугаартай хост ажиллаж байвал захидлыг шууд хүлээж авах болно; хэрэв ямар нэг байдлаар энд хост руу хандах боломжгүй байвал бусад нь (заримдаа "нөөц MX-үүд" гэж нэрлэх нь бий) түр хугацаанд захидлыг хүлээж аваад, арай бага дугаартай хосттой холбоо тогтоох боломжтой болмогц түүн рүү дамжуулах гэм мэтчилэн явсаар, эцэст нь хамгийн бага дугаартай хост захидлыг хүлээн авах болно.

Бусад MX-үүд таныхаас тусдаа Интернэт холболттой байх шаардлагатай. Таны ISP эсвэл өөр найрсаг газрууд танд энэ үйлчилгээг үзүүлэхэд асуудал байх ёсгүй.

### 29.6.2. Таны домэйнд хаягласан Захидал

"Захидлын хост" (захидлын сервер гэж нэрлэх нь бий) зохион байгуулахын тулд, янз янзын ажлын машин руу явуулсан захидлыг энэ сервер рүү чиглүүлсэн байх ёстой. Үндсэндээ, өөрийн домэйнд байгаа ямар ч хостын захидлыг "нэхэмжилж аваад"(энэ тохиолдолд *.FreeBSD.org) түүнийг өөрийн захидлын сервер рүү дахин чиглүүлнэ гэсэн үг юм. Ингэснээр таны хэрэглэгчдийн захидлууд эзэн захидлын сервер дээр цуглах болно.

Үүнийг хялбар болгохын тулд, нэг *хэрэглэгчийн нэр*тэй хэрэглэгчийн данс хоёр машин дээр хоёулан дээр нь байх шаардлагатай. Үүнийг хийхдээ adduser(8)-г ашиглаарай.

Таны хэрэглэх захидлын хост таны сүлжээнд байгаа бүх ажлын машины хувьд томилж гаргасан захидал солилцогч байх ёстой. Энэ тохиргоог DNS сервер дээр дараах байдалтай хийнэ:

```
example.FreeBSD.org A   204.216.27.XX       ; Workstation
            MX  10 hub.FreeBSD.org  ; Mailhost
```

Энэ нь ажлын машинд хаягласан захидлыг, A бичлэг хааш нь заахаас үл хамааран захидлын хост руу дахин чиглүүлэх болно. Ингэснээр захидал MX хост руу илгээгдэх болно.

Та DNS сервер ажиллуулдаггүй бол дээр дурдсаныг өөрөө бие даан хийх боломжгүй юм. Хэрэв та DNS сервер ажиллуулдаггүй, эсвэл DNS сервер ажиллуулах боломжгүй бол өөрийн ISP эсвэл DNS үйлчилгээ үзүүлдэг газартаа хандаарай.

Хэрэв та хийсвэр захидал байршуулах үйлчилгээ үзүүлдэг бол, дараах мэдээлэл тус болно. Таны үйлчлүүлэгч өөрийн домэйн нэртэй, тухайлбал customer1.org, ба та customer1.org-д хаягласан бүх захидлыг өөрийн захидлын хост mail.myhost.com руу чиглүүлэх хүсэлтэй байна гэж бодъё. Тэгвэл таны DNS дээрх бичлэг дараах байдалтай байх болно:

```
customer1.org       MX  10  mail.myhost.com
```

customer1.org домэйны зөвхөн захидлыг хариуцахын тулд A бичлэг *шаардлагагүй*.

А бичлэг байхгүй бол customer1.org руу ping хийх боломжгүй болохыг анхаарна уу.

Хамгийн сүүлд хийх нэг зүйл бол өөрийн захидлын хост дээрх sendmail-д ямар ямар домэйнууд ба/эсвэл хостуудад хаягласан захидлыг хүлээн авахыг хэлж өгөх явдал юм. Үүнийг хэд хэдэн өөр аргаар хийж болно. Дараах аргуудын алийг ч хэрэглэж болно:

- Хэрэв та FEATURE(use_cw_file)-г хэрэглэж байгаа бол хостуудыг /etc/mail/local-host-names файл дотор нэмж бичих.
- Хэрэв та sendmail 8.10 эсвэл түүнээс хойно гарсан хувилбарыг хэрэглэж байгаа бол Cwyour.host.com гэсэн мөрийг /etc/sendmail.cf эсвэл /etc/mail/sendmail.cf файл дотор бичих.

## 29.7. UUCP-тэй SMTP

FreeBSD-тэй цуг ирдэг sendmail-ийн тохиргоо нь Интернэт рүү шууд холбогддог сайтуудад зориулагдсан юм. Өөрсдийн захидлыг UUCP-ээр солилцохыг хүссэн сайтууд sendmail-ийн өөр тохиргооны файлыг суулгах ёстой.

/etc/mail/sendmail.cf-г гараар тохируулах нь илүү нарийн, нэмэлт сэдэв юм. sendmail хувилбар 8 нь тохиргооны файлуудаа m4(1) урьдчилан процесс хийлтээр үүсгэдэг. Жинхэнэ тохиргоо нь илүү дээд хийсвэрлэлтийн түвшинд болдог. m4(1)-ийн тохиргооны файлуудыг /usr/shared/sendmail/cf сангаас олж болно. cf сан дахь README файл нь m4(1) тохиргооны үндсэн танилцуулга болон үйлчилдэг.

UUCP хүргэлтийг дэмжих хамгийн шилдэг арга бол mailertable боломжийг ашиглах явдал юм. Энэ нь чиглүүлэлтийн шийдвэрүүд гаргахдаа sendmail-ийн ашигладаг мэдээллийн баазыг үүсгэдэг.

Эхлээд та өөрийн .mc файлыг үүсгэх хэрэгтэй. /usr/shared/sendmail/cf/cf сан цөөн хэдэн жишээнүүдийг агуулдаг. Таныг өөрийнхөө файлыг foo.mc гэж нэрлэсэн гэж үзэх юм бол түүнийг зөв sendmail.cf болгож хөрвүүлэхийн тулд хийх зүйл тань:

```
# cd /etc/mail
# make foo.cf
# cp foo.cf /etc/mail/sendmail.cf
```

Ердийн .mc файл иймэрхүү харагдаж болно:

```
VERSIONID(`Your version number') OSTYPE(bsd4.4)

FEATURE(accept_unresolvable_domains)
FEATURE(nocanonify)
FEATURE(mailertable, `hash -o /etc/mail/mailertable')

define(`UUCP_RELAY', your.uucp.relay)
define(`UUCP_MAX_SIZE', 200000)
define(`confDONT_PROBE_INTERFACES')

MAILER(local)
MAILER(smtp)
MAILER(uucp)

Cw    your.alias.host.name
Cw    youruucpnodename.UUCP
```

accept_unresolvable_domains, nocanonify, болон confDONT_PROBE_INTERFACES боломжуудыг агуулсан мөрүүд нь захидал хүргэх явцын үед DNS-ийг ашиглуулахгүй байх болно. UUCP_RELAY гэдэг нь UUCP хүргэлтийг дэмжихэд шаардлагатай. .UUCP псевдо домэйн хаягуудтай ажиллаж чаддаг Интернэтийн хостын нэрийг тавих хэрэгтэй; та өөрийн ISP-ийн захидал өртөөлөгчийг энд бараг оруулах байх даа.

Ингэснийхээ дараа танд /etc/mail/mailertable файл хэрэгтэй болно. Хэрэв танд таны бүх захидлуудад ашиглагддаг, гадагш холбогдсон зөвхөн ганц холболт байвал дараах файл хангалттай байх болно:

```
#
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
.                           uucp-dom:your.uucp.relay
```

Арай илүү төвөгтэй жишээ иймэрхүү харагдаж болно:

```
#
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
#
horus.interface-business.de   uucp-dom:horus
.interface-business.de        uucp-dom:if-bus
interface-business.de         uucp-dom:if-bus
.heep.sax.de                  smtp8:%1
horus.UUCP                    uucp-dom:horus
if-bus.UUCP                   uucp-dom:if-bus
.                             uucp-dom:
```

Эхний гурван мөр нь домэйн руу хаяглагдсан захидал анхдагч чиглүүлэлт рүү биш, харин хүргэх замыг "товчлохын" тулд UUCP хөрш рүү илгээгдэхийг зохицуулдаг. Дараагийн мөр нь локал Ethernet домэйн руу хаяглагдсан, SMTP ашиглан хүргэгдэж болох захидалтай

ажилладаг. Эцэст нь .UUCP псевдо домэйн маягаар бичигдсэн UUCP хөршүүд `uucp-neighbor !recipient`-д анхдагч дүрмүүдийг дарж бичих боломжийг олгохын тулд дурдагдсан. Сүүлийн мөр нь үргэлж нэг цэг байдаг. Энэ нь UUCP хүргэлтээр дэлхий рүү гарах таны универсал захидлын гарц болон үйлчилдэг UUCP хөрш рүү гаргана гэж бусад бүгдийн хувьд зааж өгч байна. `uucp-dom:` түлхүүр үгийн ард байрлах бүх цэгийн нэрс зөв UUCP хөршүүд байх ёстой бөгөөд та тэдгээрийг `uuname` тушаал ашиглан шалгаж болно.

Санаж явах зүйл бол энэ файл нь ашиглагдахаасаа өмнө DBM мэдээллийн баазын файл руу хөрвүүлэгдэх шаардлагатай байдаг. Үүнийг хийх тушаалын мөрийг mailertable файлын дээр хэсэгт тайлбар маягаар хийх нь зүйтэй. Өөрийн mailertable файлыг өөрчлөх бүртээ та энэ тушаалыг ажиллуулах ёстой юм.

Төгсгөлийн зөвлөгөө: хэрэв та тухайн нэг захидлын чиглүүлэлт ажиллаж байгаа эсэхэд эргэлзэж байгаа бол sendmail-ийн `-bt` тохируулгыг тогтоох хэрэгтэй. Энэ нь sendmail-ийг *хаягийн тест горим*д эхлүүлдэг; `3,0` гэж оруулаад захидал чиглүүлэлтийг тест хийхийг хүссэн хаягаа оруулах хэрэгтэй. Сүүлийн мөр нь ашиглагдсан дотоод захидлын агент, энэ агентийн дуудаж холбогдох хүрэх хост болон (магадгүй хөрвүүлэгдсэн) хаягийг хэлдэг. Энэ горимоос `Ctrl` + `D` даран гарна.

```
% sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>
> 3,0 foo@example.com
canonify           input: foo @ example . com
...
parse            returns: $# uucp-dom $@ your.uucp.relay $: foo < @ example . com . >
> ^D
```

# 29.8. Зөвхөн илгээхээр тохируулах нь

Өртөөлөх серверээр дамжуулан захидлыг илгээхийг зөвхөн хүсэх олон тохиолдлууд байдаг. Зарим жишээнүүдийг дурдвал:

- Таны компьютер ширээний машин, гэхдээ та send-pr(1) зэрэг програмуудыг ашиглахыг хүсэж байна гэж бодъё. Ингэхийн тулд та өөрийн ISP-ийн захидал өртөөлөх серверийг ашиглах ёстой.
- Компьютер нь захидлыг локалаар зохицуулдаггүй сервер боловч бүх захидлуудыг боловсруулахын тулд өртөөлөх сервер рүү дамжуулах хэрэгтэй байгаа.

Бараг MTA бүр үүнийг хийх чадвартай байдаг. Харамсалтай нь захидлыг зөвхөн дамжуулахын тулд өргөн боломжтой MTA-г зөв тохируулах нь маш төвөгтэй байдаг. sendmail болон postfix зэрэг програмуудыг энэ зорилгоор ашиглах нь хэтэрхий юм.

Мөн хэрэв та энгийн Интернэт хандалтын үйлчилгээг ашиглаж байгаа бол таны гэрээ таныг "захидлын сервер" ажиллуулахыг хориглож магадгүй юм.

Тэдгээр хэрэгцээнүүдийг хангах хялбар арга бол mail/ssmtp портыг суулгах явдал юм. Дараах тушаалуудыг `root` хэрэглэгчээр ажиллуулна:

```
# cd /usr/ports/mail/ssmtp
# make install replace clean
```

Суулгасны дараа mail/ssmtp-г /usr/local/etc/ssmtp/ssmtp.conf-д байрлах дөрвөн мөр бүхий файлаар тохируулж болно:

```
root=yourrealemail@example.com
mailhub=mail.example.com
rewriteDomain=example.com
hostname=_HOSTNAME_
```

root-д зориулж өөрийн жинхэнэ захидлын хаягаа ашиглахаа мартуузай. mail.example.com -ийн оронд өөрийн ISP-ийн гарах захидлын өртөөлөх серверийг оруулаарай (зарим ISP-ууд нь үүнийг "гарах захидлын сервер" эсвэл "SMTP сервер" гэдэг).

Гарах захидлын үйлчилгээг нь оролцуулаад sendmail-г хориглохоо мартуузай. Дэлгэрэнгүйг sendmail-г идэвхгүй болгох нь хэсгээс үзнэ үү.

mail/ssmtp нь бас бусад тохируулгуудтай байдаг. Жишээнүүд болон дэлгэрэнгүй мэдээллийг /usr/local/etc/ssmtp доторх жишээ тохиргооны файл эсвэл ssmtp-ийн гарын авлагын хуудаснаас үзнэ үү.

Энэ маягаар ssmtp-г тохируулах нь зөв ажиллахын тулд захидал илгээхийг шаарддаг, таны компьютер дээрх ямар ч програмыг зөвшөөрөх болно. Ингэхдээ таны ISP-ийн хэрэглээний бодлогыг зөрчилгүйгээр эсвэл таны компьютерийг спам илгээх зорилгоор булаан эзлэхийг зөвшөөрдөггүй байна.

# 29.9. Гадагш залгах холболтоор захидлыг ашиглах нь

Хэрэв танд статик IP хаяг байгаа бол та анхдагч утгуудыг өөрчилж тааруулах шаардлагагүй юм. Өөрийн хостын нэрийг өөрт заагдсан Интернэтийн нэрээр тохируулах хэрэгтэй бөгөөд sendmail үлдсэнийг нь хийх болно.

Хэрэв та динамикаар заагдсан IP хаягтай бөгөөд Интернэт рүү диал ап PPP холболтыг ашигладаг бол та өөрийн ISP-ийн захидлын сервер дээр магадгүй захидлын хайрцагтай байна байх. Таны ISP-ийн домэйн example.net, таны хэрэглэгчийн нэр user, та өөрийн машиныг bsd.home гэж нэрлэсэн бөгөөд таны ISP танд relay.example.net-г захидал өртөөлөх сервер болгож ашиглаж болно гэж хэлсэн гэж үзье.

Өөрийн захидлын хайрцгаас захидал авахын тулд та татаж авах агентийг суулгах ёстой. fetchmail хэрэгсэл нь олон төрлийн протоколуудыг дэмждэг болохоор үүнд сайн сонголт болох юм. Энэ програм нь багц хэлбэрээр эсвэл портын цуглуулгад байдаг (mail/fetchmail). Ихэвчлэн таны ISPPOP үйлчилгээ үзүүлдэг. Хэрэв та хэрэглэгчийн PPP ашиглаж байгаа бол /etc/ppp/ppp.linkup дахь дараах оруулгыг ашиглан Интернэтийн холболт тогтох үед өөрийн захидлыг автоматаар авч болох юм:

```
MYADDR:
!bg su user -c fetchmail
```

Хэрэв та sendmail-г локал биш бүртгэлүүдэд захидал илгээхийн тулд ашиглаж байгаа бол (доор үзүүлсэн шиг) та өөрийнхөө Интернэтийн холболтыг тогтоосныхоо дараа өөрийн захидлын дарааллыг sendmail-ээр процесс хийлгэхийг хүсэх байх. Үүнийг хийхийн тулд энэ тушаалыг /etc/ppp/ppp.linkup файлд `fetchmail` тушаалын дараа байрлуулах хэрэгтэй:

```
  !bg su user -c "sendmail -q"
```

Та `bsd.home` дээр `user`-т зориулсан бүртгэлтэй гэж бодъё. `bsd.home` дээрх `user`-ийн гэрийн санд .fetchmailrc файлыг үүсгэх хэрэгтэй:

```
poll example.net protocol pop3 fetchall pass MySecret
```

Энэ файл нь `MySecret` гэсэн нууц үгийг агуулдаг болохоор уг файлыг `user`-ээс өөр хэн ч унших ёсгүй юм.

Зөв `from:` толгойтойгоор захидал илгээхийн тулд та sendmail-д user@bsd.home-ийн оронд user@example.net-г ашиглахыг хэлж өгөх ёстой. Та бас sendmail-д хурдан захидал дамжуулах боломжийг бүрдүүлэхийн тулд бүх захидлыг дамжуулахдаа `relay.example.net`-ийг ашиглахыг бас хэлж өгөхийг хүсэж болох юм.

Дараах .mc файл нь хангалттай байх болно:

```
VERSIONID(`bsd.home.mc version 1.0')
OSTYPE(bsd4.4)dnl
FEATURE(nouucp)dnl
MAILER(local)dnl
MAILER(smtp)dnl
Cwlocalhost
Cwbsd.home
MASQUERADE_AS(`example.net')dnl
FEATURE(allmasquerade)dnl
FEATURE(masquerade_envelope)dnl
FEATURE(nocanonify)dnl
FEATURE(nodns)dnl
define(`SMART_HOST', `relay.example.net')
Dmbsd.home
define(`confDOMAIN_NAME',`bsd.home')dnl
define(`confDELIVERY_MODE',`deferred')dnl
```

Энэ .mc файлыг sendmail.cf файл болгож хэрхэн хувиргах тухай дэлгэрэнгүйг өмнөх хэсгээс лавлана уу. Бас sendmail.cf-г шинэчилснийхээ дараа sendmail-г дахин эхлүүлэхээ мартуузай.

# 29.10. SMTP нэвтрэлт танилт

Өөрийн захидлын сервертээ SMTP Authentication буюу нэвтрэлт танилттай байх нь хэд хэдэн талаараа ашигтай байдаг. SMTP нэвтрэлт танилт нь sendmail-д аюулгүй байдлын өөр давхарга нэмж чадах бөгөөд хостоо сольдог хөдөлгөөнт хэрэглэгчдийн хувьд өөрийн захидлын клиентийнхээ тохиргоог дахин тохируулалгүйгээр захидлын серверээ ашиглах боломжийг олгодгоороо ашигтай байдаг.

1. security/cyrus-sasl2-г портоос суулгана. Та энэ портыг security/cyrus-sasl2 сангаас олж болно. security/cyrus-sasl2 порт нь хэд хэдэн эмхэтгэх үеийн тохируулгуудыг дэмждэг. Бидний ашиглах SMTP нэвтрэлт танилтын аргын хувьд `LOGIN` тохируулга хаагдаагүй эсэхийг шалгаарай.
2. security/cyrus-sasl2-г суулгасны дараа /usr/local/lib/sasl2/Sendmail.conf-г (байхгүй бол үүсгээрэй) засварлаж дараах мөрийг нэмэх хэрэгтэй:

   ```
   pwcheck_method: saslauthd
   ```

3. Дараа нь security/cyrus-sasl2-saslauthd-г суулгаж дараах мөрийг нэмэхийн тулд /etc/rc.conf-г засварлана:

   ```
   saslauthd_enable="YES"
   ```

   тэгээд эцэст нь saslauthd дэмонг эхлүүлнэ:

   ```
   # service saslauthd start
   ```

   Энэ дэмон нь таны FreeBSD passwd мэдээллийн баазаас нэвтрэлт таниулахын тулд sendmail-ийн хувьд зуучлагч болдог. Энэ нь SMTP нэвтрэлт танилт ашиглах хэрэгтэй хэрэглэгч бүрийн хувьд шинэ хэрэглэгчийн нэрс болон нууц үгсийг үүсгэх шаардлагагүй болгож нэвтрэлт болон захидлын нууц үгийг өөрчлөлгүйгээр хэвээр байлгадаг юм.

4. Одоо /etc/make.conf-г засварлаад дараах мөрүүдийг нэмэх хэрэгтэй:

   ```
   SENDMAIL_CFLAGS=-I/usr/local/include/sasl -DSASL
   SENDMAIL_LDFLAGS=-L/usr/local/lib
   SENDMAIL_LDADD=-lsasl2
   ```

   Эдгээр мөрүүд нь sendmail-д cyrus-sasl2 руу эмхэтгэх үед холбоос хийхэд шаардлагатай зөв тохиргооны сонголтуудыг өгөх болно. sendmail-ийг суулгахаас өмнө cyrus-sasl2 суусан байгаа эсэхийг баталгаажуулаарай.

5. Дараах тушаалуудыг ажиллуулж sendmail-г дахин эмхэтгэнэ:

```
# cd /usr/src/lib/libsmutil
# make cleandir && make obj && make
# cd /usr/src/lib/libsm
# make cleandir && make obj && make
# cd /usr/src/usr.sbin/sendmail
# make cleandir && make obj && make && make install
```

/usr/src тийм ч ихээр өөрчлөгдөөгүй бөгөөд хэрэгцээтэй хуваалцдаг сангууд байгаа тохиолдолд sendmail-ийн эмхэтгэлт ямар нэгэн асуудалгүй хийгдэх ёстой.

6. sendmail эмхэтгэгдэж дахин суулгагдсаны дараа өөрийн /etc/mail/freebsd.mc файлыг (аль эсвэл өөрийн .mc файл болгож ашигладаг тэр файлыг. Дахин давтагдахгүй байхын тулд олон администраторууд hostname(1)-ээс гарах гаралтыг ашиглан .mc файлаа болгон сонгодог) засварлах хэрэгтэй. Эдгээр мөрүүдийг нэмэх хэрэгтэй:

```
dnl set SASL options
TRUST_AUTH_MECH(`GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN')dnl
define(`confAUTH_MECHANISMS', `GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN')dnl
```

Хэрэглэгчдийн нэвтрэлт танилтын хувьд sendmail-д байдаг өөр аргуудыг эдгээр тохируулгууд нь тохируулдаг. Хэрэв та pwcheck-с өөр аргыг ашиглахыг хүсэж байгаа бол хамт ирсэн баримтаас үзнэ үү.

7. Эцэст нь /etc/mail дотор байхдаа make(1)-г ажиллуулна. Энэ нь таны шинэ .mc файлыг ажиллуулж freebsd.cf гэгдэх (өөрийн .mc файлд зориулж ашигласан тэр) .cf файлыг үүсгэх болно. Дараа нь `make install restart` тушаалыг ашиглах хэрэгтэй. Энэ нь файлыг sendmail.cf руу хуулж sendmail-г зөв ажиллуулах болно. Энэ процессийн тухай дэлгэрэнгүй мэдээллийг /etc/mail/Makefile файлаас лавлана уу.

Хэрэв бүгд зөв явсан бол та өөрийн нэвтрэх мэдээллийг захидлын клиент уруугаа оруулж тест захидал илгээж чадах ёстой. Илүү судлахын тулд sendmail-ийн `LogLevel`-г 13 болгон тохируулж /var/log/maillog файлд алдаа гарах эсэхийг харах хэрэгтэй.

SMTP нэвтрэлт танилттай холбоотой дэлгэрэнгүй мэдээллийг sendmail хуудаснаас үзнэ үү.

## 29.11. Хэрэглэгчийн захидал дамжуулагчид

Mail User Agent (MUA) буюу хэрэглэгчийн захидал дамжуулагч гэдэг нь захиаг илгээх болон хүлээн авах зориулалттай програмууд байдаг. Мөн цаашилбал хоорондоо холбогдож "захидал" чулуудацгааснаар хэрэглэгчдийн ажлыг түргэтгэж амарчилж чаддагаараа MUA нар нь илүү хүчирхэг болж байгаа билээ. FreeBSD нь өөртөө маш олон захиа дамжуулагч програмуудыг дэмждэг бөгөөд FreeBSD портын цуглуулгаас та суулгах боломжтой. Хэрэглэгчид магадгүй график дүрслэлтэй evolution эсвэл balsa, мөн консол дээр тулгуурлаж ажилладаг програмууд болох mutt, alpine эсвэл `mail`, үгүй бол том байгууллагуудад ихэнхдээ хэрэглэгддэг вэб харуулалттай захиа үзэгч зэргүүдээс сонгон хэрэглэх боломж буй.

### 29.11.1. mail

mail(1) бол FreeBSD-н хэрэглэгчийн хэрэглэж болох анхдагч захиа дамжуулагч програм юм. Энэ нь консол дээр ажилладаг бөгөөд текст хэлбэрийн захианууды дамжуулах болон хүлээж авах бүх ерөнхий гүйцэтгэлийг санал болгодог ч зарим хавсаргасан файлуудтай ажиллахад зарим хүчин чадал нь дутмаг бөгөөд зөвхөн дотоод захианы хайрцган дээр ажилладаг.

Хэдийгээр `mail` гуай анхнаасаа POP эсвэл IMAP серверүүдийг дэмждэггүй боловч эдгээр сервер дээрх захианы хайрцгуудыг өөрийнхөө дотоод mbox файл уруу fetchmail програмыг ашиглан хуулж чаддаг. fetchmail-н талаар энэ бүлэгт сүүлд танилцуулах болно (fetchmail-г ашиглах нь).

Захиа хүлээж авах болон илгээхийн тулд `mail` тушаал өгнө:

```
% mail
```

Хэрэглэгч бүрд зориулсан /var/mail доторх захианы хайрцаг нь `mail` хэрэгслээр уншигддаг. Хэрэв захианы хайрцаг хоосон байвал ямар ч захиа алга байна гэсэн мэдэгдэл харуулаад энэ хэрэгсэл гардаг. Захианы хайрцаг доторх захиа уншигдах үед энэ хэрэгслийн харуулах нүүр хэсэг ажиллаж эхлэн захианууды жагсаалт болгож харуулдаг. Дараах маягаар захианууд нь автоматаар дугаарлагдаж харуулагддаг:

```
Mail version 8.1 6/6/93.  Type ? for help.
"/var/mail/marcs": 3 messages 3 new
>N  1 root@localhost        Mon Mar  8 14:05  14/510   "test"
 N  2 root@localhost        Mon Mar  8 14:05  14/509   "user account"
 N  3 root@localhost        Mon Mar  8 14:05  14/509   "sample"
```

Захианууды одоо `t захианы-дугаар` гэж тушаал өгснөөр тухайн дугаартай захиаг уншина. Доорх жишээнд эхний захиаг уншиж харуулав:

```
& t 1
Message 1:
From root@localhost  Mon Mar  8 14:05:52 2004
X-Original-To: marcs@localhost
Delivered-To: marcs@localhost
To: marcs@localhost
Subject: test
Date: Mon,  8 Mar 2004 14:05:52 +0200 (SAST)
From: root@localhost (Charlie Root)

This is a test message, please reply if you receive it.
```

Дээрх жишээн дээр харуулснаар `t` товч дарж уншссан захианы толгой хэсгийг бүтнээр нь харуулж байна. Захианы үндсэн жагсаалт уруу буцаж очихын тулд `h` товчлуур хэрэглэгддэг.

Хэрэв захианд хариулж бичих тохиолдол гарвал mail-н R эсвэл r товчнуудыг ашигладаг. R товч нь mail-д зөвхөн илгээсэн хүнд нь хариулж бичихийг зааж өгдөг байхад r товч нь илгээсэн хүнээс гадна мөн өөр бусад хүлээн авагчид уруу илгээх боломж өгдөг. Та мөн энэ тушаалуудын ард нь захианы дугаар бичиж өгөн тухайн оноосон захианд хариулт бичиж болно. Ингэж хариулж бичих үйлдэл хийх үед захианы хамгийн сүүлд нэг . гэсэн цэгтэй мөр нэмэгддэгийг доорх жишээнээс харна уу:

```
& R 1
To: root@localhost
Subject: Re: test

Thank you, I did get your email.
.
EOT
```

Шинэ захиа илгээхийн тулд m товч хэрэглэгддэг бөгөөд энэ товчлуурын араас хүлээн авах хүний захианы хаягийг бичиж өгдөг. Олон хаяг бичигдэх тохиолдолд захианы хаяг бүр , гэсэн таслал тэмдгээр тусгаарлагддаг. Дараа нь захианы гарчгийг бичиж сүүлд нь захиагаа бичдэг. Захианы төгсгөл хэсгийг тусдаа нэг мөрөнд . гэж цэг оруулж өгснөөр захиаг дуусгадаг.

```
& mail root@localhost
Subject: I mastered mail

Now I can send and receive email using mail ... :)
.
EOT
```

mail хэрэгсэл дотор ? товчлуур дарснаар ямар ч үед тусламж мэдээллийг харуулж болдог бөгөөд мөн mail(1)-н гарын авлаганд илүү дэлгэрэнгүй мэдээллийг харах боломж бий.

Өмнө тайлбарлаж байсны дагуу mail(1) хэрэгсэл нь анхнаасаа хавсралт файлтай ажиллах зориулалттай хийгдээгүй болохоор жаахан ядуувтар хэрэглэгддэг. Сүүлийн үеийн захидал дамжуулагчид болох mutt нь хавсралт файлтай арай ухаалаг ажилладаг. Гэхдээ та заавал mail-г хэрэглэмээр байгаа бол converters/mpack порт танд хэрэг болох байх аа.

### 29.11.2. mutt

mutt бол жижигхэн боловч маш хүчтэй захидал дамжуулагч бөгөөд дараах аятайхан шинж чанаруудыг өөртөө агуулсан байдаг нь:

- Захианууддтай давхар зэрэг ажиллах
- Захианд нууцлалын хувиргалт хийх болон цахим гарын үсэг зурахад зориулагдсан PGP дэмжлэгтэй
- MIME дэмжлэгтэй

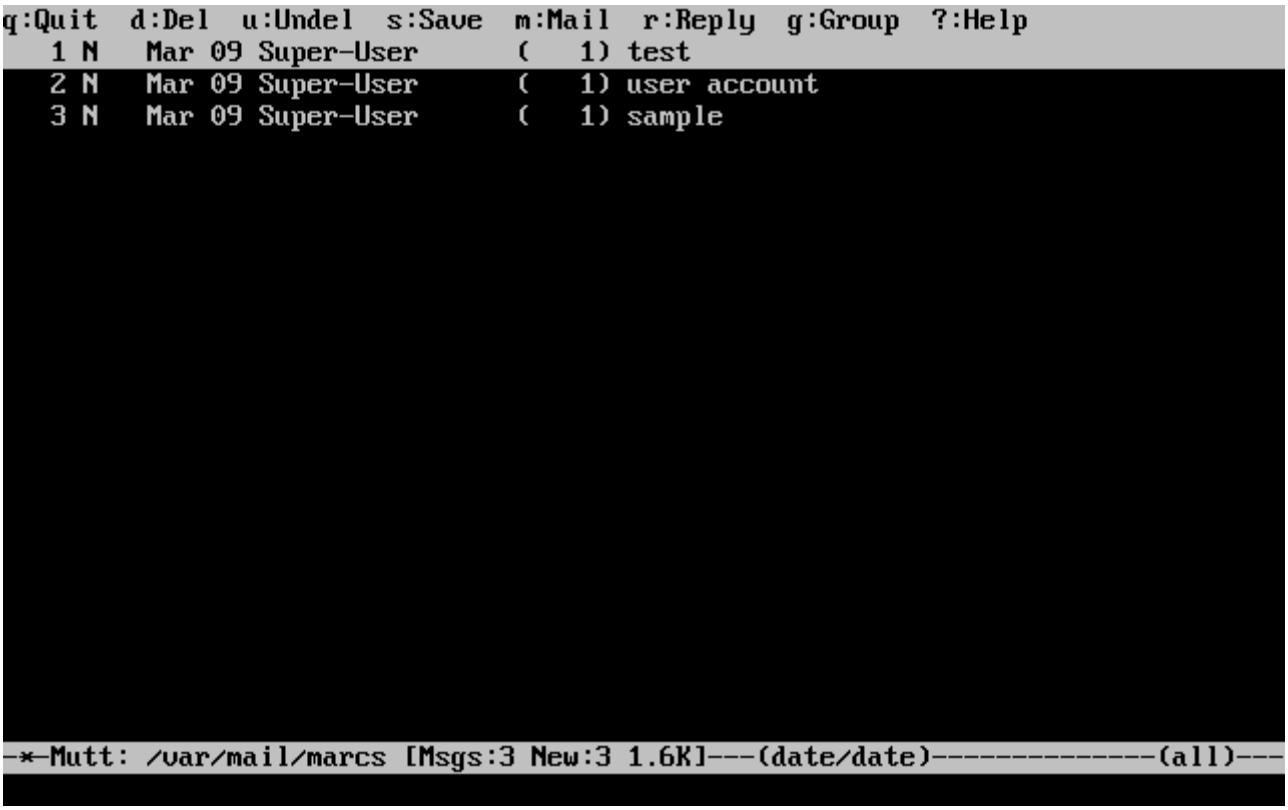
- Maildir дэмжлэгтэй
- Дурын өөрчлөлт хийх боломжтой.

Эдгээр шинж чанаруудын ачаар mutt нь хамгийн өргөн тархсан хэрэглэгчийн захидал дамжуулагч болж чадсан юм. http://www.mutt.org хаягнаас mutt-н тухай дэлгэрэнгүй мэдээллийг харна уу.

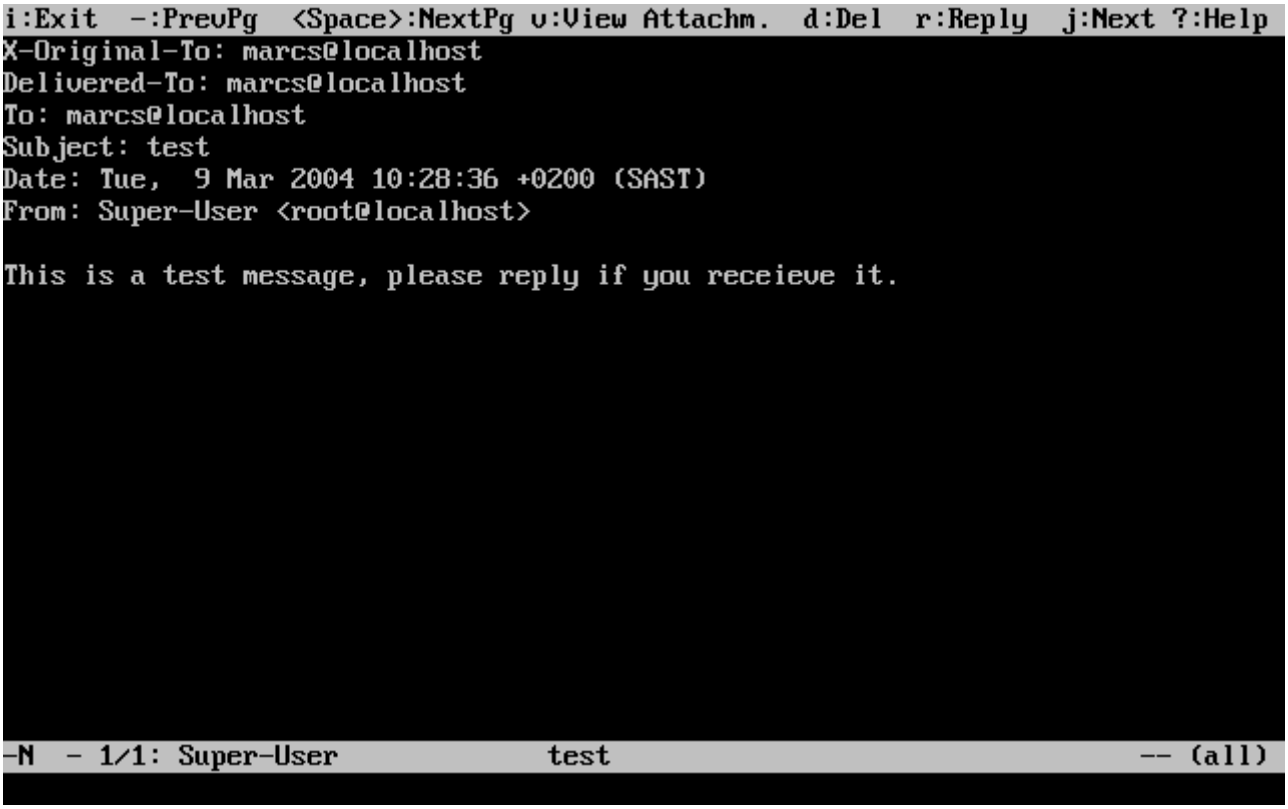
mutt-н тогтвортой хувилбарыг mail/mutt портоос суулгаж болох бөгөөд хэрэв яг одоо хөгжүүлж буй хувилбарыг нь суулгахыг хүсвэл mail/mutt-devel портоос суулгаарай.Портоо суулгасны дараа mutt-г эхлүүлэхдээ дараах тушаалыг өгдөг:

```
% mutt
```

mutt нь хэрэглэгчийн захианы хайрцгийг автоматаар /var/mail сангаас уншиж хэрэв дотор нь захиа байвал харуулдаг. Хэрэв ямар ч захиа байхгүй байвал хэрэглэгчээс тушаал оруулахыг mutt хүлээх болно. Доорх жишээнд mutt-н захидал харуулалтыг үзүүлэв:



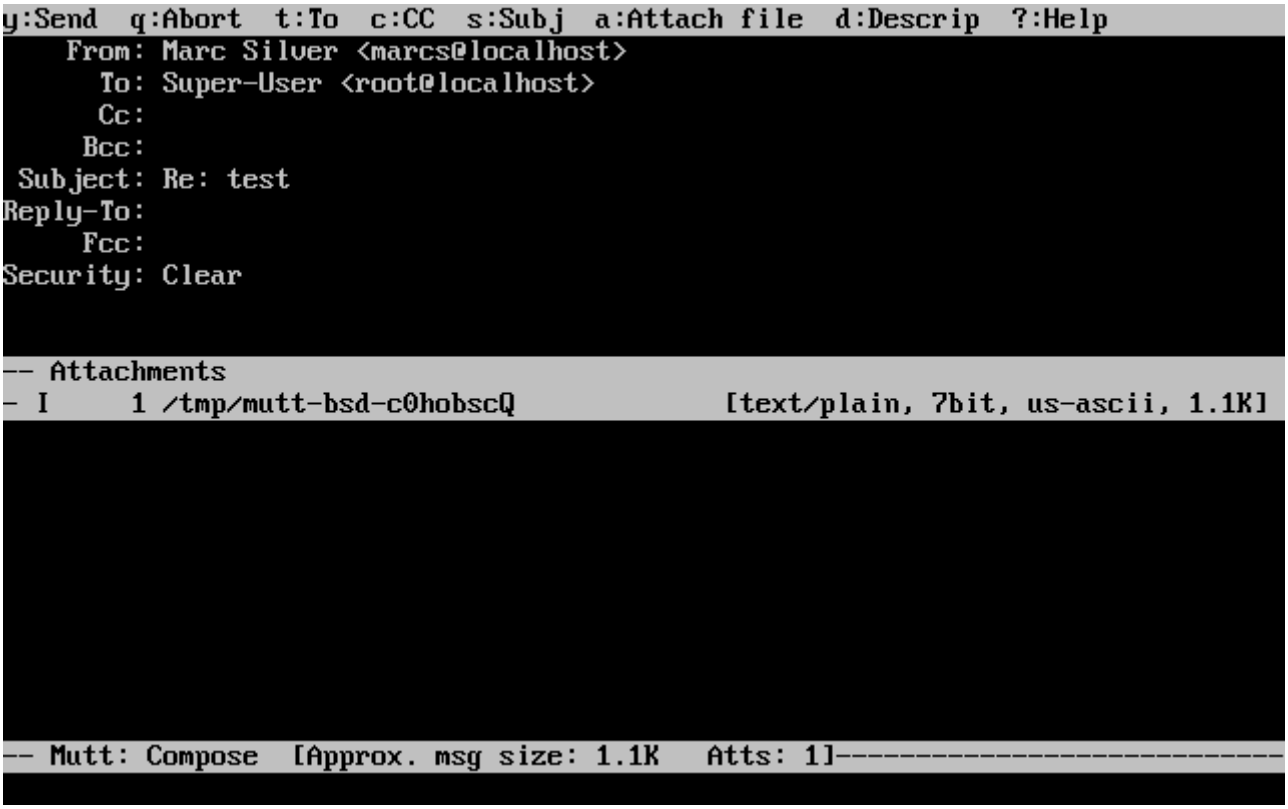

Захиаг уншихын тулд сумтай товчоор захидлыг сонгож `Enter` товч дарна. mutt-н захидал хэрхэн дэлгэж харуулсныг доорх зурган дээр харуулав:




mail(1) хэрэгсэлтэй ижилхэн mutt нь мөн зөвхөн илгээсэн хүнд нь юм уу эсвэл олон хүлээн авагч уруу хариулт бичиж болдог. Зөвхөн илгээсэн хүнд нь хариу бичихийн тулд `r` товч дарна. Захиа илгээсэн хүнийг нь оруулаад бүлэг хүмүүс рүү захиа хариулж бичихийн тулд `g` товч дардаг.

mutt нь захиа бичих болон хариулахын тулд vi(1) програмыг ашиглан захиа засварлалт хийлгэдэг. Гэхдээ өөрийн эхлэл сан дотроо .muttrc файл үүсгэж `editor` хувьсагч үүсгэх эсвэл `EDITOR` гэсэн орчны хувьсагч үүсгэж өөрийн дуртай засварлагчаа зааж өгч бас болно. http://www.mutt.org/ хаягнаас та mutt-н талаар хэрхэн тохируулж болох дэлгэрэнгүй зааврыг харна уу.

Шинэ захиа бичихийн тулд `m` товч хэрэглэнэ. Гарчгийг оруулсны дараа vi(1) програмыг mutt ажиллуулж захиагаа бичих боломж олгодог. Захиаг бичиж дууссаны дараа `vi` програмаас хадгалаад гарсны дараа илгээх захиаг mutt програм нэгтгэж харуулдаг. Захиагаа илгээхдээ `y` товч дарна. Илгээхийн өмнөх нэгтгэсэн харуулалтыг дор зурганд үзүүлье:




mutt нь өөртөө мөн тусламжийн мэдээлэлтэй өгөөд `?` товч дарж харуулагдана. Дэлгэцийн дээд хэсэгт ямар товч дарж болох тухай тайлбар мэдээлэл харуулаастай байдаг.

### 29.11.3. alpine

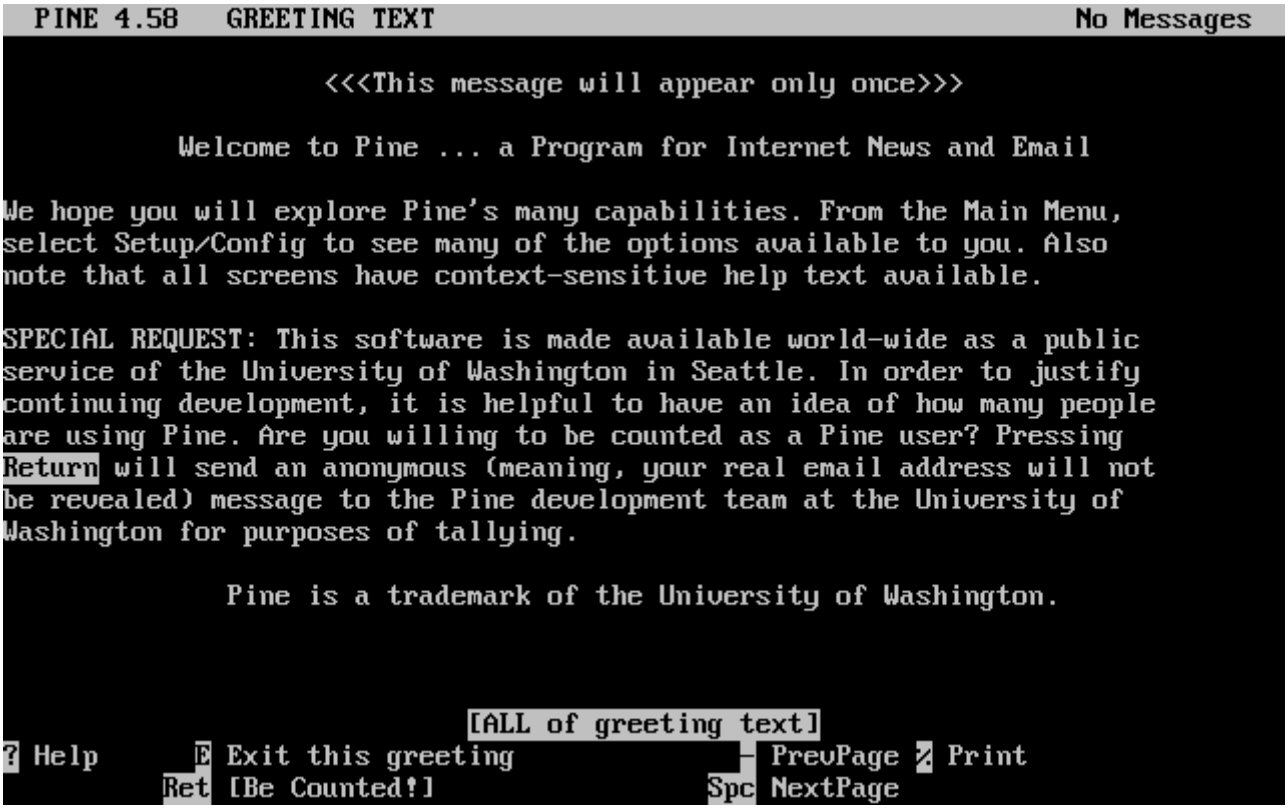
alpine програмыг эхлэн хэрэглэгчдэд зориулж үүсгэсэн бөгөөд нэмэлт шинж чанаруудыг агуулсан байдаг.

alpine-н өмнөх хувилбаруудад алсаас холбогдсон халдагчид тусгай кодыг хэрэглэгч маягаар холбогдон захиа маягаар илгээж уг аюултай кодыг гүйцэтгэж болдог аюултай нүх байжээ. Эдгээр *мэдэгдсэн* хүндрэлүүд нь засагдсан боловч alpine-ы програмын аюулгүйн хамгаалалт загвар нь тун сул болохоор FreeBSD-н аюулаас хамгаалагч нөхдүүд уг програмд өөр онгорхой нүх бий гэж итгэдэг юм байна. Та өөртөө гарах эрсдэлийг тооцож мэдэж байгаа бол alpine-г суулгаарай.

alpine-н одоогийн хөгжүүлж байгаа хувилбарыг mail/alpine портоос суулгана. Порт суугдсаны дараа alpine-г дараах тушаалаар эхлүүлдэг:

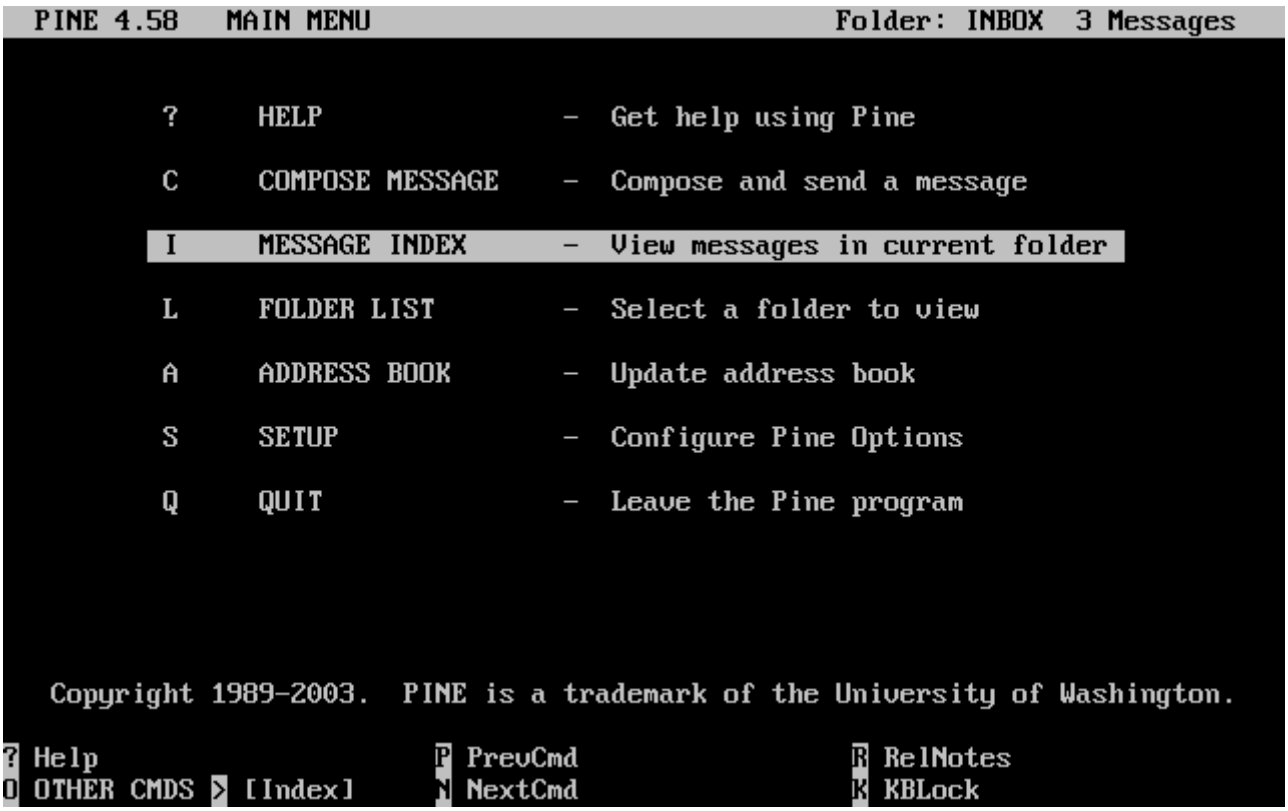
```
% alpine
```

Анх удаагаа alpine ажиллах үедээ мэндчилгээ болон товч заавар харуулаад alpine зохиосон хүмүүст мэдээлэх журмын нэргүй захиа илгээхийг хүсдэг. Тэр нэргүй захиаг илгээхийн тулд `Enter` товч дарах эсвэл мөн `E` товч дарж нэргүй захиа илгээх хэсгээс хадгалахгүй гардаг. Мэндчилгээний хуудас дараах маягаар харагддаг:

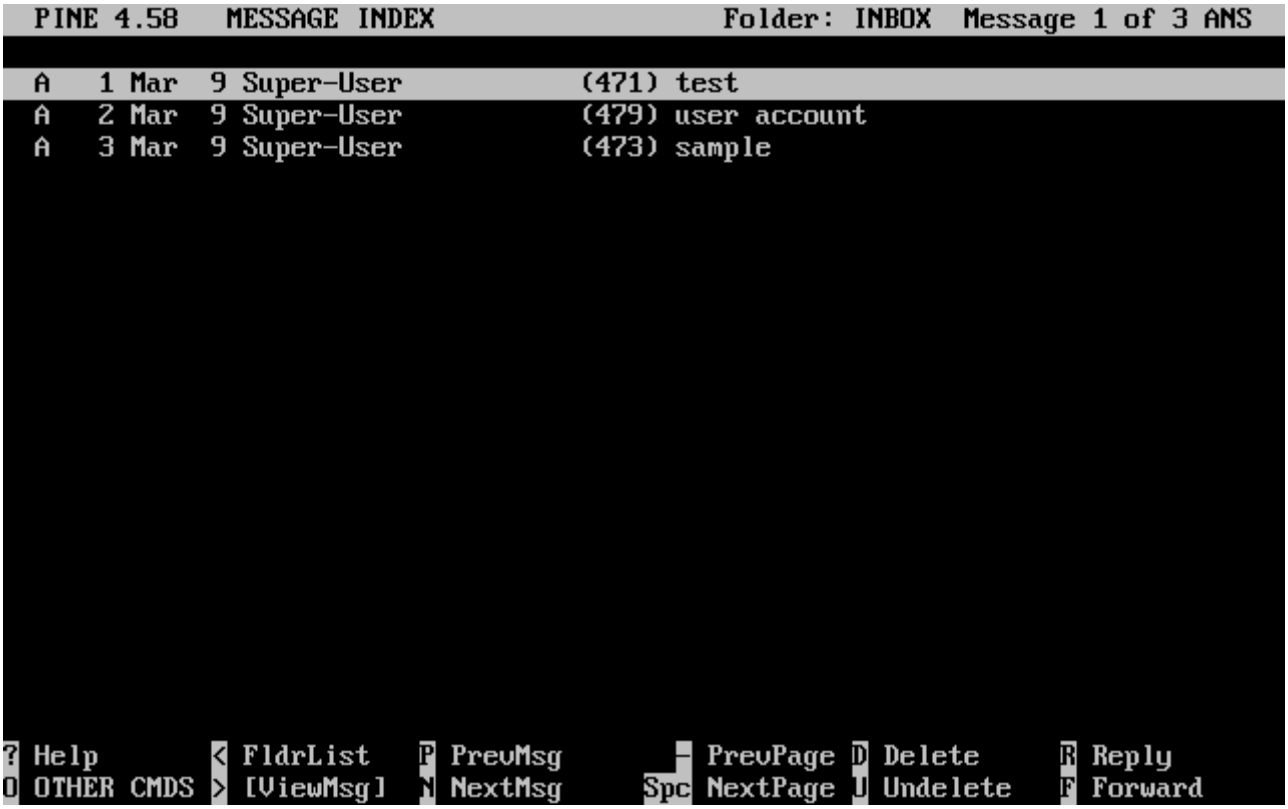



Хэрэглэгчдэд дараа нь сумтай товчоор сонгож болдог үндсэн цэс харуулагддаг. Үндсэн цэс дотор шинэ захиа бичих, захиа агуулсан сан доторхоо харах, мөн хаяг хадгалалтаа зохицуулах зэрэг боломж байдаг. Доорх зурагт эдгээр боломжуудын цэсийг харуулсан үндсэн цэсийг үзүүлэв.

alpine-р нээгдсэн үндсэн сан бол inbox гэсэн нэртэй сан байдаг. Захианы жагсаалтыг харахын тулд `I` товч дарах, эсвэл MESSAGE INDEX цэсийг сонгоно:



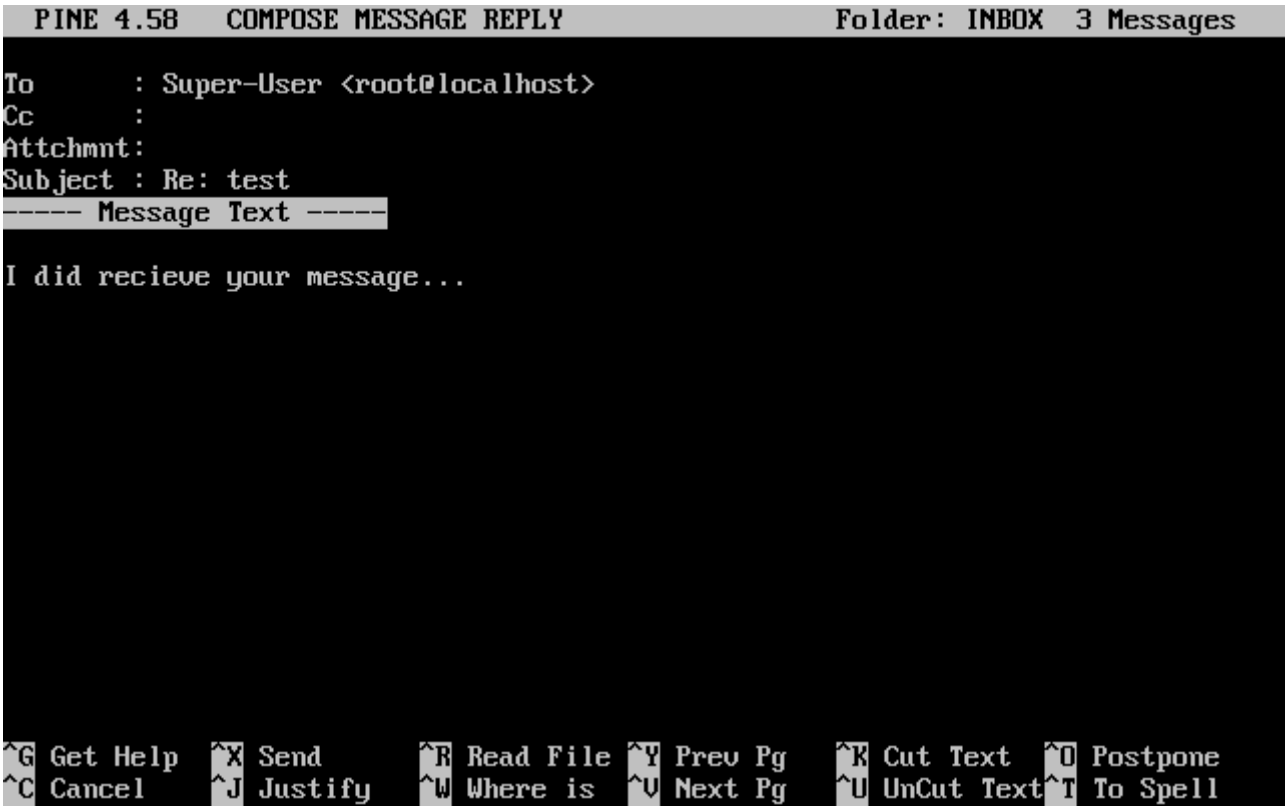
Захианы санд буй захианууды жагсааж харуулсны дараа сумтай товчоор сонгож `Enter` товч дарж уншина.




Доорх зурган дээр alpine дээр энгийн захиаг харуулав. Дэлгэцийн доод хэсэгт лавлах болгож гарын товчлууруудыг харуулдаг. Жишээ нь `r` товч дарснаар одоогийн харуулж байгаа захианд хариу бичнэ.

alpine дээр захианд хариу бичих үед pico засварлагч хэрэглэгддэг бөгөөд энэ засварлагч нь

alpine-тай хамт суугддаг. pico хэрэгсэл нь захиа бичихэд маш энгийн бөгөөд vi(1) эсвэл mail(1) зэргийг ашиглагчдыг бодвол шинэ хэрэглэгчид рүү илүү дөхүүлсэн байдаг. Хариулт захиагаа бичиж дуусаад `Ctrl` + `X` товч дарж илгээдэг. Илгээхийн өмнө alpine танаас лавлаж асуух болно.




Үндсэн цэснээс SETUP гэж сонгосноор alpine програмыг та хүссэнээрээ өөрчлөх боломжтой. http://www.washington.edu/alpine/ хаягаас дэлгэрэнгүй мэдээллийг харна уу.

## 29.12. fetchmail-г ашиглах нь

fetchmail нь хэрэглэгчдэд алсын IMAP болон POP серверүүдээс захидал автоматаар татан авч локал захидлын хайрцгуудад хадгалах боломжийг олгодог, өргөн боломжтой, IMAP болон POP клиент юм. Тэдгээр захидлын хайрцгуудад дараа нь хандахад илүү хялбар байдаг. fetchmail-г mail/fetchmail портыг ашиглан суулгаж болох бөгөөд төрөл бүрийн боломжуудыг санал болгодгоос заримыг дурдвал:

- POP3, APOP, KPOP, IMAP, ETRN болон ODMR протоколуудын дэмжлэг.
- SMTP-г ашиглан захидал дамжуулах чадвар. Энэ нь шүүлт хийх, дамжуулалт хийх болон хэвийн ажиллахын тулд хуурамч дүр үзүүлэх боломжийг олгодог.
- Шинэ захидлуудыг байнга шалгахын тулд дэмон горимоор ажиллуулж болно.
- Олон захидлын хайрцгуудыг хүлээн авч тэдгээрийг тохиргооноос хамааран өөр локал хэрэглэгчүүд рүү дамжуулж чаддаг.

fetchmail-ийн бүх боломжуудыг тайлбарлах нь энэ баримтын хүрээнээс гадуур боловч зарим нэг үндсэн боломжуудыг тайлбарлах болно. fetchmail хэрэгсэл нь зөв ажиллахын тулд .fetchmailrc гэгддэг тохиргооны файлыг шаарддаг. Энэ файл нь серверийн мэдээлэл болон нэвтрэх итгэмжлэлүүдийг агуулдаг. Энэ файлын агуулгын мэдрэмтгий чанараас болоод энэ

файлыг зөвхөн эзэмшигч нь уншиж чаддаг байхаар болгон дараах тушаал ашиглаж хийхийг зөвлөдөг:

```
% chmod 600 .fetchmailrc
```

Дараах .fetchmailrc файл нь ганц хэрэглэгчийн захидлын хайрцгийг POP ашиглан татаж авах жишээ юм. Энэ нь fetchmail-г `joesoap` хэрэглэгчийн нэр болон `XXX` нууц үгийг ашиглан `example.com` руу холбогдыг зааж байна. Энэ жишээ нь `joesoap` хэрэглэгчийг локал систем дэх хэрэглэгч гэж бас үздэг.

```
poll example.com protocol pop3 username "joesoap" password "XXX"
```

Дараах жишээ нь олон POP болон IMAP серверүүд рүү холбогдож шаардлагатай тохиолдолд өөр локал хэрэглэгчид рүү дахин чиглүүлж байна:

```
poll example.com proto pop3:
user "joesoap", with password "XXX", is "jsoap" here;
user "andrea", with password "XXXX";
poll example2.net proto imap:
user "john", with password "XXXXX", is "myth" here;
```

fetchmail хэрэгслийг `-d` туг болон түүний дараа .fetchmailrc файлд жагсаагдсан серверүүдийг байнга шалгах тэр хугацааг (секундээр) зааж өгөн дэмон горимд ажиллуулж болдог. Дараах жишээ нь fetchmail-г 600 секунд бүр шалгахаар тохируулж байна:

```
% fetchmail -d 600
```

fetchmail-ийн тухай дэлгэрэнгүй мэдээллийг http://fetchmail.berlios.de/ хаягаас олж болно.

## 29.13. procmail-г ашиглах нь

procmail хэрэгсэл нь ирж байгаа захидлыг шүүхэд хэрэглэгддэг гайхамшигтай хүчтэй програм юм. Энэ нь хэрэглэгчдэд ирж байгаа захидлуудад тааруулж "дүрмүүдийг" тодорхойлох боломжийг олгодог бөгөөд энэ дүрмүүдийг ашиглан тусгай үйлдлүүд хийх эсвэл захидлыг өөр захидлын хайрцаг ба/эсвэл цахим захидлын хаягууд руу дахин чиглүүлэх зэрэг ажлуудыг хийж болох юм. procmail-г mail/procmail порт ашиглан суулгаж болно. Суулгасны дараа үүнийг MTA-уудтай цуг ажиллахаар тохируулж болно; дэлгэрэнгүй мэдээллийг өөрийн MTA-ийн баримтаас лавлана уу. Мөн procmail-н боломжуудыг ашиглаж байгаа хэрэглэгчийн гэрийн сан дахь .forward файлд дараах мөрийг нэмж өгч болно:

```
"|exec /usr/local/bin/procmail || exit 75"
```

Дараах хэсэг нь procmail-ийн зарим үндсэн дүрмүүдийг үзүүлэх бөгөөд тэдгээрийн юу

хийдэг талаар товч тайлбар өгөх болно. Эдгээр дүрмүүд болон бусдууд нь .procmailrc файлд хийгдсэн байх ёстой бөгөөд энэ файл нь хэрэглэгчийн гэрийн санд байрлах ёстой.

Эдгээр дүрмүүдийн ихэнхийг procmailex(5)-ийн гарын авлагын хуудаснаас олж болно.

user@example.com-с ирсэн бүх захидлыг goodmail@example2.com гадаад хаяг руу дамжуулна:

```
:0
* ^From.*user@example.com
! goodmail@example2.com
```

1000 байтаас бага бүх захидлуудыг goodmail@example2.com гадаад хаяг руу дамжуулна:

```
:0
* < 1000
! goodmail@example2.com
```

alternate@example.com руу илгээсэн бүх захидлыг alternate гэгдсэн захидлын хайрцагт хийнэ:

```
:0
* ^TOalternate@example.com
alternate
```

"Spam" сэдэв бүхий бүх захидлыг /dev/null руу илгээнэ:

```
:0
^Subject:.*Spam
/dev/null
```

Ирж байгаа FreeBSD.org захидлын жагсаалтуудыг шалгаж жагсаалт бүрийг өөрийнх нь захидлын хайрцаг руу хийдэг, ашигтай зааврыг доор үзүүлэв:

```
:0
* ^Sender:.owner-freebsd-\/[^@]+@FreeBSD.ORG
{
    LISTNAME=${MATCH}
    :0
    * LISTNAME??^\/[^@]+
    FreeBSD-${MATCH}
}
```

# Chapter 30. Сүлжээний орчны Серверүүд

## 30.1. Ерөнхий агуулга

Энэ бүлэгт UNIX® системүүдэд өргөн хэрэглэгддэг, сүлжээний орчинд ажилладаг зарим нэг үйлчилгээнүүдийн талаар авч үзнэ. Бид тэдгээр үйлчилгээнүүдийг хэрхэн суулгах, тохируулах, турших болон үйлчилгээг хариуцах талаар үзэх болно. Танд зориулж жишээ тохиргооны файлуудыг мөн оруулж өгсөн байгаа.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- inetd дэмоныг хэрхэн удирдах.
- Сүлжээний орчны файл системийг хэрхэн зохион байгуулах.
- Хэрэглэгчийн бүртгэлийг хуваалцах сүлжээний орчны мэдээллийн серверийг хэрхэн зохион байгуулах.
- DHCP ашиглан автоматаар сүлжээний тохиргоог хэрхэн хийх.
- Домэйн нэрийн серверийг хэрхэн зохион байгуулах.
- Apache HTTP Серверийг хэрхэн зохион байгуулах.
- File Transfer Protocol буюу Файл Дамжуулах Протокол(FTP) Серверийг хэрхэн зохион байгуулах.
- Samba ашиглан Windows® хэрэглэгчдэд зориулсан файл болон хэвлэгч серверийг хэрхэн зохион байгуулах.
- NTP протокол ашиглан цаг болон өдрийг тохируулах хийгээд цагийн серверийг хэрхэн зохион байгуулах.
- How to configure the standard logging daemon, `syslogd`, to accept logs from remote hosts.

Энэ бүлгийг уншихаасаа өмнө, та дараах шаардлагыг хангасан байх хэрэгтэй:

- /etc/rc скриптүүдийн үндсийг ойлгосон байх.
- Сүлжээний үндсэн нэр томъёоллыг мэддэг байх.
- Гуравдагч этгээдийн програмыг(Програм суулгах. Багцууд болон портууд) хэрхэн нэмж суулгахыг мэддэг байх.

## 30.2. inetd"Супер-Сервер"

### 30.2.1. Ерөнхий агуулга

inetd(8) нь олон тооны үйлчилгээний сүлжээний холболтыг удирддаг тул заримдаа түүнийг "Интернэт Супер-Сервер" гэж нэрлэх нь бий. Гаднаас үүсч буй холболтыг inetd хүлээн авч, аль програмтай холбогдохыг тодорхойлон, тухайн процессийг салаалуулж, сокетийг түүн рүү чиглүүлнэ (програмын стандарт ролт, гаралт болон алдааны дескрипторoор үйлчилгээний сокетийг өгнө). Байнга ашиглагддаггүй үйлчилгээний хувьд inetd-г ажиллуулах нь бүх дэмонг дангаар бие-даах горимд ажиллуулсантай харьцуулахад

системийн нийт ачааллыг бууруулж өгдөг.

Голчлон, inetd нь бусад дэмонуудыг салаалуулахад хэрэглэгддэг боловч chargen, auth, ба daytime гэх мэт нилээд олон ердийн протоколуудыг шууд зохицуулан ажиллуулж чадна.

Энэ хэсэгт inetd-н үндсэн тохиргоог тушаалын мөрний тохируулгаар, мөн /etc/inetd.conf тохиргооны файлаар хэрхэн хийхийг үзэх болно.

### 30.2.2. Тохиргоо

inetd нь rc(8) системээр эхлүүлэгдэнэ. `inetd_enable` тохируулгын анхдагч утга нь `NO` бөгөөд, системийг суулгах явцад хэрэглэгчийн зааж өгсний дагуу sysinstall програмын тусламжтай идэвхжүүлж болно.

```
inetd_enable="YES"
```

эсвэл

```
inetd_enable="NO"
```

гэсэн мөрийг /etc/rc.conf файл дотор байрлуулснаар inetd-г систем ачаалахад эхэлдэг болгож болно. Доор дурдсан:

```
service inetd rcvar
```

тушаалыг өгөн одоо идэвхтэй байгаа тохиргоог харж болно.

Дээр нь, `inetd_flags` тохируулгаар дамжуулан inetd програмд тушаалын мөрнөөс өөр бусад тохируулгуудыг зааж өгч болно.

### 30.2.3. Тушаалын мөрний тохируулгууд

Ихэнх сервер дэмоны нэгэн адил, inetd нь түүнийг өөрчлөн тохируулахад зориулагдсан олон тооны тохируулгуудын хамт ирдэг. Сонголтуудын бүрэн жагсаалтыг inetd(8) гарын авлагын хуудаснаас үзнэ үү.

/etc/rc.conf файл доторх `inetd_flags` тохируулгыг ашиглан эдгээр тохируулгуудыг inetd-д дамжуулна. Анхдагч байдлаар, `inetd_flags` нь `-wW -C 60` гэсэн утгыг авсан байх ба энэ нь inetd-ны үйлчилгээнүүдийн хувьд TCP wrapping буюу TCP-ийн дундын хяналтыг идэвхжүүлэх ба нэг IP хаягнаас аль нэг үйлчилгээнд нэг минутанд 60-аас дээш удаа хүсэлт тавих боломжгүй болгоно.

Хэдийгээр бид хурдыг хэрхэн хязгаарлахыг доор үзүүлж байгаа ч, анхлан суралцагчдын хувьд эдгээр параметрүүдийг ихэвчлэн өөрчлөх шаардлагагүй байдаг. Эдгээр тохируулга нь гаднаас хэтэрхий олон тооны хандалт хийгдэж байгаа үед тустай байдаг Тохируулгуудын бүрэн жагсаалтыг inetd(8) заавар хуудаснаас үзнэ үү.

**-c maximum**

Үйлчилгээг нэгэн зэрэг хамгийн ихдээ хэдэн удаа дуудаж болохыг заана; Анхдагч утга нь хязгааргүй. Үйлчилгээ тус бүрээр `max-child` параметрийн тусламжтай утгыг дарж өөрчилж болно.

**-C rate**

Үйлчилгээг нэг IP хаягнаас нэг минутын дотор хамгийн ихдээ хэдэн удаа дуудаж болохыг заана; Анхдагч утга нь хязгааргүй. Үйлчилгээ тус бүрээр `max-connections-per-ip-per-minute` параметрийн тусламжтай утгыг дарж өөрчилж болно.

**-R rate**

Үйлчилгээг нэг минутын дотор хамгийн ихдээ хэдэн удаа дуудаж болохыг заана; Анхдагч утга нь 256. 0-г тавьснаар хязгааргүй болгоно.

**-s maximum**

Үйлчилгээг нэг IP хаягнаас хамгийн ихдээ хэдэн удаа дуудаж болохыг заана; Анхдагч утга нь хязгааргүй. Үйлчилгээ тус бүрээр `max-child-per-ip` параметрийн тусламжтай утгыг дарж өөрчилж болно.

### 30.2.4. inetd.conf

inetd-г /etc/inetd.conf файлын тусламжтай тохируулна.

/etc/inetd.conf файлд өөрчлөлт хийсний дараа, inetd-р тохиргооны файлыг дахин уншуулахдаа дараах тушаалыг өгнө:

*Жишээ 36. inetd-н тохиргооны файлыг дахин ачаалах нь*

```
# service inetd reload
```

Тохиргооны файлын мөр бүр тусдаа дэмонг заана. Файл доторх тайлбарууд нь мөрийн эхэнд "#" тэмдэгтэй байна. /etc/inetd.conf файл доторх бичлэгүүдийн формат дараах байдалтай байна:

```
service-name
socket-type
protocol
{wait|nowait}[/max-child[/max-connections-per-ip-per-minute[/max-child-per-ip]]]
user[:group][/login-class]
server-program
server-program-arguments
```

IPv4 ашигладаг ftpd(8) дэмоны хувьд жишээ бичлэг дараах байдалтай байж болно:

```
ftp     stream  tcp     nowait  root    /usr/libexec/ftpd       ftpd -l
```

**service-name**

Тухайн дэмоны үйлчилгээний нэрийг заана. Энэ нь /etc/services файл дотор бичигдсэн үйлчилгээнүүдийн нэг байх ёстой бөгөөд аль портон дээр сонсохыг inetd-д хэлж өгнө. Хэрэв шинэ үйлчилгээ үүсгэсэн бол түүнийг заавал /etc/services файл дотор нэмсэн байх ёстой.

**socket-type**

`stream`, `dgram`, `raw`, эсвэл `seqpacket` эдгээрийн нэг байна. `stream`-г холболтон дээр үндэслэсэн TCP дэмонуудын хувьд хэрэглэдэг бол, `dgram`-г UDP протоколоор ажилладаг дэмонуудын хувьд хэрэглэнэ.

**protocol**

Доор дурдсанаас нэг нь байна:

| Протокол | Тайлбар |
|---|---|
| tcp, tcp4 | TCP IPv4 |
| udp, udp4 | UDP IPv4 |
| tcp6 | TCP IPv6 |
| udp6 | UDP IPv6 |
| tcp46 | TCP IPv4 ба v6 хоёул |
| udp46 | UDP IPv4 ба v6 хоёул |

**{wait|nowait}[/max-child[/max-connections-per-ip-per-minute[/max-child-per-ip]]]**

`wait|nowait` нь inetd-р дуудагдсан дэмон өөрийн сокетийг удирдаж чадах эсэхийг заана. `dgram` төрлийн сокет дэмоны хувьд `wait` тохируулгыг хэрэглэх ёстой байдаг бол, ихэвчлэн олон урсгалтай байдаг `stream` сокет дэмоны хувьд `nowait` тохируулгыг хэрэглэх хэрэгтэй байдаг. `wait` нь ихэвчлэн олон сокетийг нэг дэмонд шилжүүлэн өгдөг бол, `nowait` нь шинээр үүссэн сокет тус бүрт харгалзуулан хүүхэд дэмонг салаалуулан үүсгэдэг.

inetd-ийн салаалуулан үүсгэж болох хамгийн их хүүхэд дэмоны тоог `max-child` тохируулгын тусламжтай зааж өгч болно. Хэрэв тухайн дэмоны ажиллаж болох тохиолдлыг 10-р хязгаарлах бол, `nowait`-н ард `/10` гэж бичнэ. `/0` нь хүүхдийн тоог хязгаарлахгүй гэсэн утгатай.

`max-child`-с гадна, нэг газраас тухайн дэмонтой үүсгэж байгаа холболтын тоог хязгаарладаг өөр хоёр тохируулгыг хэрэглэж болно. `max-connections-per-ip-per-minute` нь тухайн ямар нэг IP хаягнаас нэг минутанд үүсгэж болох холболтын тоог хязгаарлана, жишээлбэл: 10 гэсэн утга нь тухайн ямар нэг IP хаягнаас нэг минутын дотор тухайн үйлчилгээнд холбогдохоор оролдох оролдлогын тоог 10-р хязгаарлана. `max-child-per-ip` нь Тухайн ямар нэг IP хаяг дээр үүсгэгдсэн хүүхдийн тоог хязгаарлана. Эдгээр тохируулгууд нь санаатай болон санамсаргүйгээр нөөцийг хэтрүүлэн хэрэглэх, мөн Үйлчилгээг Зогсоох (DoS) халдлагаас хамгаалахад хэрэгтэй байдаг.

Хэрэглэхдээ, `wait` ба `nowait` хоёрын аль нэгийг заавал хэрэглэх ёстой. Харин `max-child`, `max-connections-per-ip-per-minute` ба `max-child-per-ip` тохируулгуудыг сонгон хэрэглэж болно.

Stream төрлийн олон урсгалтай дэмоны хувьд, `max-child`, `max-connections-per-ip-per-minute` эсвэл `max-child-per-ip` хязгаарлалтуудын алийг ч хэрэглэхгүй тохиолдолд ердөө: `nowait` байна.

Дээрхтэй адил дэмон, 10 хүүхэд дэмоны хязгаарлалттай бол: `nowait/10` байна.

Мөн адил дэмон, 10 хүүхэд дэмоны хязгаарлалттай, минутанд нэг IP хаягнаас үүсгэх холболтын тоог 20-р хязгаарлах бол: `nowait/10/20` болно.

Эдгээр тохируулгуудыг fingerd(8) дэмоны анхдагч тохиргоон дээр жишээ болгон харвал:

```
finger stream  tcp     nowait/3/10 nobody /usr/libexec/fingerd fingerd -s
```

Эцэст нь, 100 хүүхдийн хязгаарлалттай, нэг IP хаягнаас үүсэх холболтын тоог 5-р хязгаарласан дэмоны жишээг авбал: `nowait/100/0/5` байх юм.

**user**

Энд тухайн дэмон ямар хэрэглэгчийн нэрээр ажиллахыг зааж өгнө. Ихэвчлэн дэмонууд `root` хэрэглэгчийн нэр дээр ажилладаг. Аюулгүй байдлын үүднээс, зарим серверүүд `daemon`, эсвэл хамгийн бага эрхтэй `nobody` хэрэглэгчийн нэр дээр ажиллах нь элбэг байдаг.

**server-program**

Энд гаднаас холболт хүлээн авахад ажиллуулах дэмоны бүрэн замыг зааж өгнө. Хэрэв энэ дэмон inetd-р удирдагдсан дотоод үйлчилгээ бол `internal` тохируулгыг хэрэглэх хэрэгтэй.

**server-program-arguments**

Үүнийг `server-program`-тай хамт, `argv[0]`-с эхлэн програмын аргументыг зааж өгөх байдлаар хэрэглэнэ. Хэрэв командын мөрөнд `mydaemon -d` гэсэн байдлаар хэрэглэдэг бол, `server-program-arguments`-н утга `mydaemon -d` байна. Дахин хэлэхэд, хэрэв тухайн дэмон дотоод үйлчилгээний нэг бол `internal`-г энд мөн хэрэглэнэ үү.

## 30.2.5. Аюулгүй байдал

Үйлдлийн системийг суулгах үед хийсэн сонголтуудаас хамааран inetd-н үйлчилгээнүүдийн ихэнх нь идэвхтэй болсон байдаг. Хэрэв хэрэглэх онцын шаардлага байхгүй бол тэдгээрийг идэвхгүй болгоно уу. /etc/inetd.conf файл дотор, идэвхгүй болгох гэж байгаа демоныхоо харгалзах мөрийн урд "#" тэмдгийг тавьж өгнө. Дараа нь inetd-н тохиргоог дахин ачаална. fingerd зэрэг зарим дэмонууд гадны халдагчид хэрэгтэй мэдээллийг түгээж байдаг тул тэдгээр үйлчилгээг бүрмөсөн хааж болох юм.

Зарим дэмонууд аюулгүй байдлыг бодолцолгүйгээр бүтээгдсэн байдаг ба холболт тогтоох харьцангуй урт болзоот хугацаатай, эсвэл болзоот хугацааг огт зааж өгөөгүй байдаг. Энэ нь халдагчид тодорхой дэмон уруу холболт тогтоох хүсэлтийг олон дахин илгээж, нөөцийг дуусгах замаар системд халдах боломжийг олгодог. Хэрэв ямар нэг дэмоны хувьд үүссэн холболтын тоо хэтэрхий олон байвал `max-connections-per-ip-per-minute`, `max-child` эсвэл `max-child-per-ip` тохиргооны тусламжтайгаар хязгаарлалт хийх нь оновчтой байдаг.

Анхдагч байдлаар TCP-ийн дундын хяналт (гүйцэтгэл хялбаршуулалт) идэвхтэй байдаг.

inetd-р дуудагдсан дэмонуудын хувьд TCP хязгаарлалтыг хэрхэн тавих талаар дэлгэрэнгүй мэдээллийг hosts_access(5) заавар хуудаснаас үзнэ үү.

### 30.2.6. Элдэв зүйлс

daytime, time, echo, discard, chargen, ба auth бүгд inetd-н дотоод үйлчилгээнүүд юм.

auth үйлчилгээ нь сүлжээний орчинд, тодорхойлолт өгөх үйлчилгээ үзүүлдэг бөгөөд тодорхой түвшинд тохиргоо хийх боломжтой байдаг бол бусад үйлчилгээнүүдийг зөвхөн идэвхтэй эсвэл идэвхгүй болгох боломжтой.

Дээрх үйлчилгээнүүдийн талаар бүрэн дүүрэн мэдээллийг inetd(8) заавар хуудаснаас үзнэ үү.

## 30.3. Сүлжээний Файлын Систем (NFS)

FreeBSD дээр дэмжигддэг олон файлын системүүдийн нэг бол Network File System буюу Сүлжээний Файлын Систем юм, мөн NFS гэж нэрлэнэ. NFS нь сүлжээний орчинд файл болон санг бусадтай хуваалцах боломжийг олгодог. NFS-г хэрэглэн, хэрэглэгчид болон програмууд алслагдсан систем рүү дотоод файл руу хандаж байгаатай адилаар хандах боломжтой.

NFS-н тэмдэглүүштэй давуу талуудаас дурдвал:

- Өргөн хэрэглэгддэг өгөгдлийг нэгтгэн нэг машин дээр байрлуулж, түүнд алсаас хандах боломжтой болсноор дотоод машинууд илүү бага диск хэрэглэх болно.
- Хэрэглэгчийн хувьд сүлжээнд байгаа машин бүр дээр тус тусдаа гэрийн сантай байх шаардлагагүй болно. Гэрийн санг нэг удаа NFS сервер дээр үүсгээд түүнийгээ сүлжээгээр дамжин хэрэглэх боломжтой.
- Уян диск, CDROM болон Zip® төхөөрөмжүүдийг сүлжээний бусад машинууд хэрэглэх боломжтой болно. Ингэснээр сүлжээнд хэрэглэгдэх зөөвөрлөх боломжтой хадгалах төхөөрөмжүүдийн тоог багасгана.

### 30.3.1. NFS хэрхэн ажилладаг вэ

NFS нь үндсэн хоёр хэсгээс бүрдэнэ: сервер болон нэг ба түүнээс дээш тооны харилцагч. Сервер машин дээр хадгалагдаж байгаа өгөгдөл рүү харилцагч алсаас хандана. Дээрх үйлдлийг зөв гүйцэтгэхийн тулд нилээд хэдэн процессийн тохиргоог хийж, ажиллуулсан байх ёстой.

Сервер дээр дараах дэмонууд ажиллаж байх ёстой:

| Дэмон | Тайлбар |
|---|---|
| nfsd | NFS харилцагчдаас ирэх хүсэлтийг хүлээн авах NFS дэмон. |
| mountd | nfsd(8)-с дамжиж ирсэн хүсэлтийг гүйцэтгэгч NFS холбох дэмон. |

| Дэмон | Тайлбар |
| --- | --- |
| rpcbind | Энэ дэмоны тусламжтай NFS харилцагчид NFS сервер аль портон дээр ажиллаж байгааг олж мэднэ. |

Харилцагч nfsiod гэсэн дэмонг мөн ажиллуулж болно. nfsiod дэмон NFS серверээс ирэх хүсэлтийг гүйцэтгэнэ. Ингэх нь системийг хэвийн, алдаагүй ажиллуулахад зайлшгүй шаардлагагүй боловч зарим үзүүлэлтүүдийг сайжруулдаг тул нэмэлт байдлаар хэрэглэж болно. Дэлгэрэнгүй мэдээллийг nfsiod(8) хуудаснаас үзнэ үү.

### 30.3.2. NFS-н тохиргоог хийх

NFS-н тохиргоог хийх нь харьцангуй амархан. Ажиллах ёстой процессуудыг системтэй хамт автоматаар асдаг болгохын тулд /etc/rc.conf файлыг бага зэрэг өөрчлөхөд хангалттай.

NFS сервер дээрх /etc/rc.conf файл дотор дараах тохируулгууд идэвхжсэн байгаа эсэхийг шалгана уу:

```
rpcbind_enable="YES"
nfs_server_enable="YES"
mountd_flags="-r"
```

mountd нь NFS серверийг идэвхжүүлсэн тохиолдолд өөрөө автоматаар ажиллана.

Харилцагч талд, /etc/rc.conf файл дотор дараах тохируулга идэвхтэй байгаа эсэхийг шалгана уу:

```
nfs_client_enable="YES"
```

/etc/exports файл дотор NFS ямар файл системүүдийг экспорт хийхийг (заримдаа "хуваалцах" гэж мөн нэрлэнэ) зааж өгнө. /etc/exports файлын мөр бүр нь нэг файл системд харгалзана. Энэ файл системд хандах эрхтэй машинуудыг заахаас гадна, ямар тохируулгаар хандахыг мөн зааж өгч болно. Энэ файл дотор бичигдэж болох нилээд олон ийм тохируулгууд байгаа хэдий ч, бид тэдгээрээс зөвхөн заримыг нь энд авч үзэх болно. Та бусад тохируулгуудын талаар exports(5) заавар хуудаснаас уншиж мэднэ үү.

Доор /etc/exports файлаас хэдэн жишээ мөрийг үзүүлэв:

Дараах жишээн дээрээс файл системийг хэрхэн экспортлох санааг олж авах болно. Тохируулгууд нь таны сүлжээний тохиргоо, нөхцөл байдлаас шалтгаалан өөр байхыг анхаарна уу. Жишээ нь, /cdrom гэсэн санг 3 машин руу экспортлохын тулд дараах байдалтай бичнэ. Жишээн дээрх 3 машин сервертэй адил домэйн нэртэй, эсвэл таны /etc/hosts файл дотор тодорхойлогдсон гэж үзсэн болно. `-ro` туг нь экспортлогдож буй файл системийг зөвхөн унших боломжтой болохыг заана. Энэ тугийг тавьснаар алсаас хандаж буй машин энэ файл систем дээр ямар нэг өөрчлөлт хийх боломжгүй болно.

```
/cdrom -ro host1 host2 host3
```

Дараах жишээн дээр /home санг IP хаягаар нь зааж өгсөн 3 машин руу экспортолж байна. Ингэж IP хаягаар нь зааж өгөх нь дотоод сүлжээндээ DNS сервер ажиллуулаагүй үед их хэрэгтэй байдаг. Эсвэл /etc/hosts файл дотор дотоод хостуудын нэрийг тохируулж болно; hosts(5) хэсгийг дахин үзнэ үү. `-alldirs` гэсэн туг нь дэд сангуудыг холболтын цэг байхыг зөвшөөрч өгдөг. Өөрөөр хэлбэл, дэд сангуудыг холболгүй орхиж, харилцагч зөвхөн өөрийн хэрэгцээтэй байгаа сангаа холбохыг зөвшөөрнө гэсэн үг юм.

```
/home  -alldirs  10.0.0.2 10.0.0.3 10.0.0.4
```

Дараах жишээн дээр /a санг хоёр өөр домэйноос 2 харилцагч хандаж болохоор экспортолж байна. `-maproot=root` гэсэн туг нь алслагдсан систем дээрх `root` хэрэглэгч экспортлогдсон файл систем дээр `root` эрхээр бичихийг зөвшөөрнө. Хэрэв `-maproot=root` тугийг тусгайлан зааж өгөөгүй бол, хэдий алслагдсан систем дээрх хэрэглэгч `root` эрхтэй ч экспортлогдсон файл систем дээр бичих эрхгүй болно.

```
/a  -maproot=root  host.example.com box.example.org
```

Харилцагч экспортлогдсон файл систем рүү хандахын тулд эрх нь байх ёстой. Тухайн харилцагч /etc/exports файл дотор бүртгэлтэй эсэхийг шалгаарай.

/etc/exports файл дотор мөр болгон нь нэг файл системийг нэг хост руу экспортлох мэдээллийг төлөөлнө. Алслагдсан хост аль нэг файл системийн хувьд зөвхөн ганц удаа л тодорхойлогдсон байх ёстой ба үүнд харгалзах ганцхан анхдагч бичлэг байж болно. Жишээ нь, /usr нь нэг файл систем гэж бодъё. /etc/exports файл доторх дараах бичлэгүүд нь буруу юм:

```
# Invalid when /usr is one file system
/usr/src   client
/usr/ports client
```

Учир нь /usr гэсэн файл системийг `client` гэсэн хост руу экспортолсон хоёр бичлэг байна. Энэ тохиолдолд дараах форматаар бичвэл зөв болно:

```
/usr/src /usr/ports  client
```

Нэг хост руу экспортлогдож байгаа файл системийн хувьд шинжүүдийг бүгдийг нэг мөрөнд жагсаан бичих ёстой. Харилцагчийг зааж өгөөгүй мөрүүдийг энгийн хост гэж үзнэ. Энэ нь файл системийг экспортлох боломжийг хязгаарлана, гэвч энэ нь ихэнх хүмүүст хүнд асуудал биш байдаг.

Дараагийн жишээн дээр /usr ба /exports гэсэн дотоод файл системийг экспортолсон байна:

```
# Export src and ports to client01 and client02, but only
# client01 has root privileges on it
/usr/src /usr/ports -maproot=root    client01
/usr/src /usr/ports                client02
# The client machines have root and can mount anywhere
# on /exports. Anyone in the world can mount /exports/obj read-only
/exports -alldirs -maproot=root      client01 client02
/exports/obj -ro
```

/etc/exports файл дотор гарсан өөрчлөлтүүдийг хүчинтэй болгохын тулд, өөрчлөлт орсон тухай бүрд mountd дэмонг албадан /etc/exports-г дахин уншуулах хэрэгтэй болдог. Үүний тулд эсвэл HUP дохиог ажиллаж байгаа дэмонд өгөх хэрэгтэй:

```
# kill -HUP `cat /var/run/mountd.pid`
```

эсвэл `mountd` rc(8) скриптийг зохих параметрийн хамт ажиллуулах хэрэгтэй:

```
# service mountd onereload
```

rc скриптийг хэрэглэх зааврыг FreeBSD дээр rc(8) ашиглах нь хэсгээс үзнэ үү.

Бас нэг боломж нь, FreeBSD-г эхнээс нь ачаалж, бүх процессийг дахин эхлүүлэх юм. Гэвч үүний тулд заавал системийг дахин ачаалах шаардлага байхгүй. `root` эрхээр дараах тушаалуудыг өгснөөр зөвхөн хэрэгтэй процессуудаа дахин эхлүүлэх боломжтой.

NFS сервер дээр:

```
# rpcbind
# nfsd -u -t -n 4
# mountd -r
```

NFS харилцагч дээр:

```
# nfsiod -n 4
```

Одоо алсын файл системийг холбоход бэлэн боллоо. Доорх жишээнүүд дээр серверийн нэрийг `server`, харилцагчийн нэрийг `client` гэж авсан болно. Хэрэв та алсын файл системийг зөвхөн түр хугацаагаар холбох гэж байгаа эсвэл тохиргоогоо шалгах гэж байгаа бол, харилцагч талд `root` эрхээр дараах тушаалыг өгөхөд хангалттай:

```
# mount server:/home /mnt
```

Энэ тушаалыг өгснөөр та сервер талд байгаа /home гэсэн санг харилцагч талд байгаа /mnt

сантай холбох болно. Хэрэв бүх зүйл зөв тохируулагдсан бол, та харилцагч талын /mnt сан дотор орж сервер дээр байгаа файлуудыг харж чадах ёстой.

Хэрэв систем шинээр ачаалах бүрд ямар нэг алсын файл системийг холбох хүсэлтэй байгаа бол, түүнийгээ /etc/fstab файл дотор нэмж бичих хэрэгтэй. Жишээ нь:

```
server:/home    /mnt    nfs rw  0   0
```

Боломжит бүх сонголтуудын талаар fstab(5) заавар хуудаснаас үзнэ үү.

### 30.3.3. Цоожлолт

Зарим програмууд (ж.н. mutt) зөв ажиллахын тулд файл цоожлолтыг шаарддаг. NFS-н хувьд, rpc.lockd-г файл цоожлолтонд хэрэглэж болно. Түүнийг идэвхжүүлэхийн тулд, сервер болон харилцагч талд хоёуланд нь /etc/rc.conf файл дотор дараах мөрүүдийг нэмж өгөх хэрэгтэй (NFS сервер болон харилцагч талуудыг аль хэдийн тохируулчихсан гэж үзэв):

```
rpc_lockd_enable="YES"
rpc_statd_enable="YES"
```

Програмыг дараах байдалтай эхлүүлнэ:

```
# service lockd start
# service statd start
```

Хэрэв NFS харилцагч болон NFS сервер талуудын хооронд жинхэнэ файл цоожлолт хийгдэх шаардлагагүй бол, NFS харилцагч талд mount_nfs(8)-д `-L` тохируулгыг өгөн дотоод цоожлолт хийлгэж болно. Дэлгэрэнгүй мэдээллийг mount_nfs(8) заавар хуудаснаас үзнэ үү.

### 30.3.4. Практик хэрэглээ

NFS нь олон практик хэрэглээтэй. Хамгийн элбэг тохиолддог хэрэглээг доор жагсаав:

- Олон машиныг нэг CDROM эсвэл төхөөрөмжийг дундаа хэрэглэдэг байхаар зохион байгуулах. Энэ нь нэг програмыг олон машин дээр суулгах хамгийн хямд, хялбар арга юм.
- Том сүлжээний хувьд, бүх хэрэглэгчдийн гэрийн санг хадгалдаг төвлөрсөн NFS серверийг тохируулах. Эдгээр гэрийн сангуудыг дараа нь сүлжээний орчинд экспортолсноор хэрэглэгчид аль машин дээр ажиллаж буйгаас үл хамааран өөрийн нэг л сан дотор ажиллах боломжтой болно.
- Олон машин дундаа нэг /usr/ports/distfiles сантай байх. Ийм замаар, нэг портыг олон машин дээр суулгах хэрэгтэй үед машин бүр дээр эх файлыг татаж авалгүйгээр хурдан суулгах боломжтой болно.

### 30.3.5. amd-р автоматаар холбох нь

amd(8) (автоматаар холбогч дэмон) нь алсын файл системийн файл эсвэл санд хэрэглэгч хандах тухай бүрт уг файл системийг автоматаар холбодог. Хэсэг хугацааны туршид идэвхгүй байгаа файл системийг amd мөн автоматаар салгана. amd-г хэрэглэснээр /etc/fstab дотор бичигддэг байнгын холболтоос гадна, холболт хийх боломжийг олгодог.

amd нь өөрийгөө, /host ба /net сангууд дээр NFS сервер байдлаар холбож ажиллах бөгөөд эдгээр сангууд доторх файлд хандах үед, amd харгалзах алсын холболтыг хайж олоод автоматаар холбох болно. /net нь экспортлогдсон файл системийг IP хаягаар нь холбоход, харин /host нь хост нэрээр нь холбоход хэрэглэгдэнэ.

/host/foobar/usr сан доторх файлд хандана гэдэг нь amd-г `foobar` гэсэн хост дээр экспортлогдсон /usr санг холбохын зааж өгнө.

*Жишээ 37. Экспортыг amd-р холбох*

Алсын хост дээр байгаа боломжит холболтуудын жагсаалтыг `showmount` тушаалын тусламжтай харж болно. Жишээлбэл, `foobar` нэртэй хостын экспортыг харахын тулд:

```
% showmount -e foobar
Exports list on foobar:
/usr                               10.10.10.0
/a                                 10.10.10.0
% cd /host/foobar/usr
```

Жишээн дээр үзүүлснээр `showmount` нь /usr-г экспортлогдсон болохыг харуулж байна. /host/foobar/usr сан дотор ороход, amd нь `foobar` гэсэн хост нэрийг тайлахыг оролдох ба зааасан санг холбоно.

amd-г эхлэл скриптүүдээр эхлүүлж болох ба үүний тулд /etc/rc.conf файл дотор дараах мөрийг нэмэх хэрэгтэй:

```
amd_enable="YES"
```

Мөн, amd програмд `amd_flags` тохируулгын тусламжтай тугуудыг өгч болно. `amd_flags`-н анхдагч утга нь:

```
amd_flags="-a /.amd_mnt -l syslog /host /etc/amd.map /net /etc/amd.map"
```

/etc/amd.map файл дотор экспортуудыг холбох анхдагч тохируулгуудыг зааж өгсөн байна. /etc/amd.conf файл дотор amd-н илүү дээд түвшний чанаруудыг тодорхойлж өгнө.

Дэлгэрэнгүй мэдээллийг amd(8) ба amd.conf(8) заавар хуудаснаас үзнэ үү.

### 30.3.6. Бусад системтэй нэгтгэхэд тохиолдох асуудлууд

ISA PC системд зориулсан зарим Ethernet адаптерууд учир дутагдалтай байдгаас сүлжээний орчинд ажиллахад, тэр дундаа NFS-тэй ажиллахад нилээд асуудалтай байдаг. Энэ асуудал зөвхөн FreeBSD-д тохиолддоггүй боловч FreeBSD систем үүнд нилээд өртөмтгий байдаг.

Энэ асуудал нь (FreeBSD) PC системийг өндөр үзүүлэлттэй машинуудтай (жишээлбэл, Silicon Graphics, Inc., ба Sun Microsystems, Inc компаниудын хийсэн) сүлжээнд холбох үед бараг үргэлж тохиолддог. NFS холболт хийхэд асуудалгүй, зарим үйлдлүүдийг хийхэд амжилттай байх боловч, гаднаас ирж явж байгаа хүсэлтүүдийг боловсруулж чадаж байгаа хэдий ч сервер гэнэт харилцагчид хариу өгөхгүй болдог. Энэ асуудал мөн харилцагчийн хувьд, харилцагч FreeBSD систем эсвэл ажлын машин байхаас үл шалтгаалан тохиолдоно. Ихэнх системийн хувьд, нэгэнт ийм асуудалд орсон бол харилцагч талыг ном ёсных нь дагуу унтраах боломжгүй болдог. Ганц авдаг арга хэмжээ бол системийг хүчээр унтрааж асаах юм. Учир нь, NFS-н энэ асуудал одоо хир нь шийдэгдээгүй байна.

Хэдийгээр "зөв" шийдэл бол FreeBSD системд тохирох илүү өндөр үзүүлэлттэй, илүү багтаамжтай Ethernet адаптерийг олж авах боловч, боломжит ажиллагааг хангахын тулд нэг арга байна. Хэрэв FreeBSD систем нь *сервер* бол, харилцагч талаас холболт хийхдээ `-w=1024` тохируулгыг оруулж өгөх явдал юм. Хэрэв FreeBSD систем нь *харилцагч* бол, NFS файл системтэй холбогдохдоо `-r=1024` тохируулгыг хэрэглэх юм. Эдгээр тохируулгуудыг автомат холболтын хувьд fstab бичлэгийн дөрөв дэх талбарыг ашиглан, гар аргаар холболт хийх бол mount(8) тушаалын `-o` параметрыг ашиглан зааж өгч болно.

NFS сервер болон харилцагчид өөр өөр сүлжээнд байхад гардаг өөр нэг асуудлыг энэ асуудалтай хольж хутгах тохиолдол байдгийг энд дурдах нь зүйтэй болов уу. Хэрэв тийм бол, чиглүүлэгчид шаардлагатай UDP мэдээллийг дамжуулж чадаж байгаа эсэхийг *нягталж* үзээрэй. Үгүй бол, өөр юу ч хийлээ гээд та үр дүнд хүрч чадахгүй.

Дараах жишээн дээр, `fastws` нь өндөр үзүүлэлттэй ажлын машины хост (интерфэйс) нэр, `freebox` нь бага үзүүлэлттэй Ethernet адаптертай FreeBSD системийн нэр юм. Мөн, /sharedfs нь экспортлогдох гэж байгаа NFS файл систем (exports(5)-г үз), ба /project нь харилцагч талын экспортлогдсон файл системийг холбох цэг байх болно. Аль ч тохиолдолд, `hard` эсвэл `soft` ба `bg` зэрэг нэмэлт тохируулгууд таны хувьд хэрэгтэй байж болох юм.

FreeBSD системийг (`freebox`) `freebox` дээр /etc/fstab дотор харилцагч байдлаар зааж өгөх жишээ:

```
fastws:/sharedfs /project nfs rw,-r=1024 0 0
```

`freebox` дээр гараар холбохдоо:

```
# mount -t nfs -o -r=1024 fastws:/sharedfs /project
```

FreeBSD системийг (`freebox`) `fastws` дээр /etc/fstab дотор сервер байдлаар зааж өгөх жишээ:

```
freebox:/sharedfs /project nfs rw,-w=1024 0 0
```

`fastws` дээр гараар холбохдоо:

```
# mount -t nfs -o -w=1024 freebox:/sharedfs /project
```

Бараг бүх 16-битийн Ethernet адаптерийн хувьд унших ба бичих хэмжээн дээр дээрх байдлаар хязгаарлалт хийлгүйгээр ажиллах боломжтой байдаг.

Сонирхсон улсуудад толилуулахад, дээрх алдаа гарахад чухам юу тохиолддог, яагаад засагдах боломжгүй болох талаар дор тайлбарлав. NFS нь голчлон 8 K (хэдийгээр илүү бага хэмжээтэй хэсэг дээр ажиллаж чадах боловч) хэмжээтэй "блок"ууд дээр ажилладаг. Хамгийн урт Ethernet пакет 1500 байт орчим байх ба, NFS "блок" нь хэд хэдэн Ethernet пакетуудад хуваагдах хэрэгтэй болдог. Дээд түвшний програмын хувьд энэ нь нэг нэгж хэвээр байх ба хүлээж аваад, нийлүүлээд, бататгал хийхэд ч мөн нэг нэгж хэвээр байдаг. Өндөр үзүүлэлттэй ажлын машинууд NFS нэгжийг бүрдүүлж байгаа тэдгээр пакетуудыг стандартад заасны дагуу аль болох ойрхон ойрхон, нэг нэгээр нь цувуулж гаргана. Жижиг, бага багтаамжтай картууд дээр, дээд түвшний програмд дамжуулахаас өмнө сүүлийн пакет нь өмнөх пакетаа дарснаар тухайн нэгжийг буцааж нийлүүлж, бататгах боломжгүй болно. Үүнээс болж, ажлын машины болзоот хугацаа дуусаж бүхэл бүтэн 8 K нэгжийг дахин дамжуулах болно. Энэ үйл ажиллагаа дахин дахин хязгааргүй давтагдах болно.

Нэгжийн хэмжээг Ethernet пакетийн хэмжээнээс бага байлгаснаар, бид Ethernet пакет тус бүрийг бататгаж мухардалд орохоос сэргийлж чадна.

Өндөр үзүүлэлттэй ажлын машинууд PC систем рүү өгөгдлийг цацсаар байх үед давхцал үүссээр байх боловч, илүү сайн карт ашигласнаар NFS "нэгж"ийн хувьд заавал тийм давхцал үүсэх албагүй болно. Давхцал үүссэн тохиолдолд, түүнд өртсөн нэгжийг дахин дамжуулах ба түүнийг хүлээн авч, нийлүүлж, бататгах боломж өндөртэй.

# 30.4. Сүлжээний Мэдээллийн Систем (NIS/YP)

## 30.4.1. Энэ юу вэ?

NIS, нь Network Information Services буюу Сүлжээний Мэдээллийн Үйлчилгээнүүд гэсэн үгийн товчлол бөгөөд UNIX® (анхандаа SunOS™) системүүдийн удирдлагыг төвлөрүүлэх зорилгоор Sun Microsystems анх хөгжүүлсэн. Одоо энэ салбарын үндсэн стандарт болжээ; бүх гол UNIX® төрлийн системүүд (Solaris™, HP-UX, AIX®, Линукс, NetBSD, OpenBSD, FreeBSD, гэх мэт) NIS-г дэмждэг.

NIS анх Yellow Pages буюу Шар Хуудас гэсэн нэртэй байсан боловч худалдааны тэмдгийн асуудлаас болж Sun нэрийг нь сольсон. Хуучин нэр (ба үр) нь одоо хир нь хэрэглэгдсээр байдаг.

Энэ нь RPC дээр үндэслэсэн, нэг NIS домэйнд байгаа бүлэг машинууд дундаа адилхан тохиргооны файлтай боломжийг олгодог харилцагч/сервер систем юм. Үүний тусламжтай

системийн администратор NIS харилцагч системийг зайлшгүй байх үндсэн тохиргоотойгоор үүсгэх, тохиргооны өгөгдлийг нэг дор нэмэх, хасах, өөрчлөх зэрэг үйлдлүүдийг хийх боломжтой болдог.

Энэ нь Windows NT®-н домэйн системтэй төстэй. Хэдийгээр тэдгээрийн дотоод ажиллагаа нь ердөө ч адилхан биш боловч үндсэн үүргийг нь адилтгаж болох юм.

### 30.4.2. Таны мэдэж байх ёстой Нэр томъёо/Процессууд

NIS сервер эсвэл NIS харилцагч байдлаар ажилладаг NIS-г FreeBSD дээр зохион байгуулахын тулд нилээд хэдэн нэр томъёо, чухал хэрэглэгчийн процессуудтай та тааралдах болно:

| Нэр томъёо | Тайлбар |
| --- | --- |
| NIS домэйн нэр | NIS мастер сервер болон түүний бүх харилцагчид (түүний зарц серверийг оруулаад) бүгд NIS домэйн нэртэй байна. Windows NT®-н домэйн нэртэй адилаар, NIS домэйн нэр DNS-тэй ямар ч хамаагүй. |
| rpcbind | RPC-г (Remote Procedure Call буюу Алсын Процедур Дуудах, NIS-н ашигладаг сүлжээний протокол) идэвхтэй байлгахын тулд заавал ажиллаж байх ёстой. Хэрэв rpcbind ажиллахгүй бол, NIS сервер ажиллуулах, NIS харилцагч болох боломжгүй. |
| ypbind | NIS харилцагчийг NIS сервертэй "холбоно". NIS домэйн нэрийг системээс авч, RPC ашиглан сервертэй холбоно. ypbind нь NIS орчны харилцагч-серверийн харилцааны цөм нь болж өгдөг; Хэрэв харилцагчийн машин дээр ypbind үхвэл, NIS сервер рүү хандах боломжгүй болно. |
| ypserv | Зөвхөн NIS сервер дээр ажиллаж байх ёстой; энэ бол NIS сервер процесс өөрөө юм. Хэрэв ypserv(8) үхвэл, сервер NIS хүсэлтэд хариу өгөх боломжгүй болно (магадгүй, түүний үүргийг үргэлжлүүлэх зарц сервер байгаа байх). Зарим NIS-н хувьд (FreeBSD-гийх биш), анх холбогдож байсан сервер байхгүй болбол өөр сервертэй холбоо тогтоохыг оролддоггүй хувилбарууд байдаг. Ихэнхдээ, ийм үед ганц тус болох зүйл бол сервер процессийг дахин эхлүүлэх (эсвэл серверийг бүхлээр нь), эсвэл харилцагч талын ypbind процессийг дахин эхлүүлэх юм. |

| Нэр томъёо | Тайлбар |
| --- | --- |
| rpc.yppasswdd | Зөвхөн NIS эзэн сервер дээр ажиллаж байх ёстой өөр нэг процесс; Энэ дэмон NIS харилцагч нарыг өөрсдийн нэвтрэх үгийг солих боломжийг олгоно. Хэрэв энэ дэмон ажиллахгүй бол, хэрэглэгчид NIS эзэн сервер рүү нэвтэрч орон тэнд нэвтрэх үгээ солих хэрэгтэй болно. |

## 30.4.3. Хэрхэн ажилладаг вэ?

NIS орчинд гурван төрлийн хост байна: эзэн сервер, зарц сервер, ба харилцагч. Серверүүд нь хостуудын тохиргооны мэдээллийг хадгалсан агуулахын үүргийг гүйцэтгэнэ. Эзэн сервер энэ мэдээллийн бүрэн эрхтэй хуулбарыг хадгалж байдаг бол, зарц сервер нь энэ мэдээллийн хуулбарыг нөөцөнд хадгалж байдаг. Серверүүд харилцагчдыг эдгээр мэдээллээр хангана.

Олон файлд байгаа мэдээллийг энэ маягаар хуваалцаж хэрэглэнэ. master.passwd, group, ба hosts гэсэн файлуудыг ихэвчлэн NIS тусламжтай хуваалцана. Эдгээр файлд байдаг мэдээлэл харилцагч талын нэг процессод хэрэгтэй боллоо гэхэд түүнийг өөрийн дотоодоос хайхын оронд түүнд оноогдсон NIS серверээс асуулга хийнэ.

#### 30.4.3.1. Машины төрөл

- *NIS эзэн сервер*. Энэ сервер, Windows NT®-н анхдагч домэйн сервер хянагчийн нэг адил, NIS харилцагчдын хэрэгцээний бүх файлуудыг агуулсан байна. passwd, group ба NIS харилцагчийн хэрэглэх бусад олон файлууд эзэн сервер дээр байна.

Нэг машин нэгээс олон NIS домэйны хувьд NIS эзэн сервер байж болно. Гэхдээ, энд бид бага хэмжээний NIS орчны талаар ярилцах тул энэ талаар энд үзэхгүй.

- *NIS зарц сервер*. Windows NT®-н нөөц домэйн хянагчтай адилаар, NIS зарц сервер нь NIS эзэн серверийн өгөгдлийн файлын хуулбарыг хадгална. NIS зарц серверүүд нь нөөцөнд байдаг. Тэдгээр нь мөн эзэн серверийн ачааллыг хуваалцаж байдаг: NIS Харилцагчид нь хамгийн түрүүнд хариу өгсөн серверт холбогдох ба үүний тоонд зарц серверүүд ч бас орно.
- *NIS харилцагч*. NIS харилцагч нь ихэнх Windows NT® ажлын машины адилаар, NIS серверт шалгуулж (эсвэл Windows NT® ажлын машины хувьд Windows NT® домэйн хянагчид) нэвтэрнэ.

## 30.4.4. NIS/YP-г хэрэглэх нь

Энэ хэсэгт жишээ NIS орчныг үүсгэх болно.

#### 30.4.4.1. Төлөвлөх

Та өөрийгөө нэгэн их сургуулийн жижигхэн лабораторын администратор гэж бод. Энэ лаб 15 FreeBSD машинаас бүрдэх ба одоогоор төвлөрсөн удирдлага байхгүй; машин бүр өөрийн

/etc/passwd ба /etc/master.passwd файлуудтай. Эдгээр файлуудыг адилхан байлгахын тулд гараараа зөөж тавьдаг; одоогийн байдлаар лабораторид шинэ хэрэглэгч нэмэхийн тулд, бүх 15 машин дээр нэг бүрчлэн `adduser` тушаалыг оруулах хэрэгтэй байгаа. Мэдээж үүнийг өөрчлөх хэрэгтэй, иймээс та лабораторидоо NIS хэрэглэхээр боллоо. Машинуудаасаа хоёрыг нь сервер болгохоор сонгож авлаа.

Тиймээс, лабораторын тохиргоо дараах байдалтай байна:

| Машины нэр | IP хаяг | Машины үүрэг |
|---|---|---|
| `ellington` | `10.0.0.2` | NIS эзэн |
| `coltrane` | `10.0.0.3` | NIS зарц |
| `basie` | `10.0.0.4` | Факультетийн ажлын машин |
| `bird` | `10.0.0.5` | Харилцагч машин |
| `cli[1-11]` | `10.0.0.[6-17]` | Бусад харилцагч машинууд |

Хэрэв та NIS зураглалыг анх удаа хийж байгаа бол, хаанаас эхлэхээ эхлээд сайн бодох хэрэгтэй. Сүлжээ чинь ямар ч хэмжээтэй байж болно, гол нь хэд хэдэн сонголт хийх хэрэгтэй.

##### 30.4.4.1.1. NIS Домэйн Нэрийг сонгох нь

Өөрийн тань байнга хэрэглэдэг "домэйн нэр" байж болохгүй. Залруулж хэлбэл "NIS домэйн нэр" байх ёстой. Харилцагч мэдээлэл олж авахын тулд хүсэлтээ цацах үед NIS домэйн нэрийг хэрэглэнэ. Үүгээр нэг сүлжээнд байгаа олон серверүүд хэн нь хэний асуултанд хариулах ёстойгоо мэдэж авна. NIS домэйн нэрийг хоорондоо ямар нэг байдлаар хамаатай бүлэг хостын нэр гэж ойлгож болно.

Зарим байгууллагууд өөрийн Интернэтийн домэйн нэрийг NIS домэйн нэрээр хэрэглэх нь байдаг. Энэ нь сүлжээний ямар нэг асуудлыг задлан шинжлэх явцад түвэг удах тул энэ аргыг зөвлөдөггүй. NIS домэйн нэр нь сүлжээний орчинд цор ганц байх ёстой бөгөөд төлөөлж байгаа бүлэг машинаа онцолсон нэр байвал дөхөм байдаг. Жишээлбэл, Acme Inc. компаний Урлагийн хэлтэс "acme-art" гэсэн NIS домэйнтой байж болох юм. Бид өөрсдийн жишээндээ `test-domain` гэсэн домэйн нэрийг авлаа.

Гэвч, зарим үйлдлийн системүүд (цохон дурдвал SunOS™) өөрийн NIS домэйн нэрийг Интернэт домэйн нэрээр хэрэглэдэг. Хэрэв таны сүлжээний нэг болон түүнээс дээш тооны машин ийм асуудалтай бол, та Интернэт домэйн нэрээ NIS домэйндоо хэрэглэх *ёстой*.

##### 30.4.4.1.2. Серверт тавигдах шаардлагууд

NIS серверт зориулсан машин сонгон авахдаа анхаарах хэд хэдэн зүйлс бий. NIS-тэй холбоотой нэг учир дутагдалтай зүйл бол харилцагчдын серверээс хамаарах хамаарал юм. Хэрэв харилцагч өөрийн NIS домэйныг асуухаар сервертэй холбогдож чадахгүй бол, тэр машин ашиглагдах боломжгүй болдог. Хэрэглэгч болон бүлгийн мэдээлэл дутуугаас ихэнх системүүд түр хугацаанд зогсдог. Тиймээс, дахин дахин асааж унтраалгаад байхаргүй, эсвэл туршилтад хэрэглэгдэхээр машиныг сонгох хэрэгтэй. NIS сервер нь тусдаа, зөвхөн NIS серверт зориулагдсан машин байх ёстой. Хэрэв ачаалал багатай сүлжээнд ажиллаж байгаа

бол, NIS серверийг өөр үйлчилгээ ажиллаж байгаа машин дээр тавьж болох талтай. Хамгийн гол нь NIS сервер чинь ажиллахгүй болбол, *бүх* NIS харилцагчид чинь мөн ажиллахгүй болохыг санаарай.

#### 30.4.4.2. NIS Серверүүд

Бүх NIS мэдээлэл он цагийн дарааллаараа NIS эзэн сервер дээр хадгалагдаж байдаг. Энэ мэдээллийг хадгалж байгаа өгөгдлийн санг NIS буулгалт гэж нэрлэнэ. FreeBSD-д, эдгээр буулгалтууд /var/yp/[domainname] файл дотор байрлана. [domainname] нь NIS домэйн нэр болно. Нэг NIS сервер хэд хэдэн домэйныг зэрэг агуулж чадах тул домэйн тус бүрт зориулсан хэд хэдэн ийм сан байж болно. Домэйн бүр өөрийн гэсэн буулгалтуудтай байна.

NIS эзэн болон зарц серверүүд бүх NIS хүсэлтийг `ypserv` дэмоны тусламжтай удирдаж явуулна. `ypserv` нь NIS харилцагч нараас ирж буй хүсэлтийг хүлээн авч, домэйныг хөрвүүлэн, уг домэйн нэрд харгалзах өгөгдлийн файлын замыг хайж олоод, өгөгдлийг буцаан харилцагчид дамжуулах үүрэгтэй.

##### 30.4.4.2.1. NIS Эзэн Серверийг зохион байгуулах нь

Эзэн NIS серверийг зохион байгуулах нь харьцангуй ойлгомжтой. FreeBSD нь бэлэн NIS суучихсан ирдэг. Зөвхөн /etc/rc.conf файл дотор дараах мөрүүдийг нэмэхэд л хангалттай, үлдсэнийг нь FreeBSD таны өмнөөс хийгээд өгөх болно.

```
nisdomainname="test-domain"
```

1. Энэ мөр сүлжээ асахад (жишээ нь, систем дахин ачаалсны дараа) NIS домэйн нэрийг `test-domain` болгоно.

   ```
   nis_server_enable="YES"
   ```

2. Энэ мөр нь сүлжээ асахад NIS сервер процессуудыг асаахыг хэлж өгнө.

   ```
   nis_yppasswdd_enable="YES"
   ```

3. Энэ мөр нь `rpc.yppasswdd` дэмонг идэвхжүүлнэ. Дээр хэлсэнчлэн, энэ дэмон нь харилцагч машин дээрээс хэрэглэгч өөрийн NIS нэвтрэх үгийг солих боломжтой болгодог.

Таны NIS тохиргооноос хамааран, нэмэлт мөрүүдийг оруулах хэрэгтэй болж магадгүй. NIS сервер мөртлөө давхар NIS харилцагч серверийн тухай хэсгээс, доор, дэлгэрэнгүй мэдээллийг авна уу.

Дээрхийг тохируулсны дараа супер хэрэглэгчийн эрхээр `/etc/netstart` тушаалыг ажиллуулна. Энэ нь таны /etc/rc.conf файл дотор тодорхойлж өгсөн утгуудыг ашиглан бүх зүйлсийг таны өмнөөс хийх болно. Хамгийн сүүлд нь NIS буулгалтуудыг эхлүүлэхээс өмнө

ypserv демоныг гараар ажиллуулах хэрэгтэй.

```
# service ypserv start
```

#### 30.4.4.2.2. NIS Буулгалтуудыг эхлүүлэх нь

*NIS буулгалтууд* нь өгөгдлийн сангийн файлууд бөгөөд /var/yp сан дотор хадгалагдана. Тэдгээрийг NIS эзэн серверийн /etc сан дотор байгаа /etc/master.passwd файлаас бусад тохиргооны файлуудаас үүсгэдэг. Энэ нь их учиртай. Мэдээж та өөрийн `root` болон удирдах эрхтэй дансуудынхаа нэвтрэх үгийг NIS домэйн дахь бүх сервер дээр тарааж тавих хүсэлгүй байгаа биз дээ. Тиймээс, NIS буулгалтуудыг эхлүүлэхийн өмнө, дараах зүйлсийг хийх хэрэгтэй:

```
# cp /etc/master.passwd /var/yp/master.passwd
# cd /var/yp
# vi master.passwd
```

Системийн дансуудад хамаарах мөрүүдийг (`bin`, `tty`, `kmem`, `games`, гэх мэт), мөн NIS харилцагч дээр тарааж тавих хүсэлгүй байгаа дансуудад хамаарах мөрүүдийг (жишээлбэл `root` ба бусад UID 0 (супер хэрэглэгчийн) дансууд) бүгдийг устгах хэрэгтэй.

/var/yp/master.passwd файл бүлгийн болон нийтийн хувьд унших эрхгүй (600 төлөв) байгааг нягтална уу! Шаардлагатай бол `chmod` тушаалыг хэрэглээрэй.

Дээр дурдсаныг гүйцэтгэж дууссаны дараа, сая NIS буулгалтуудыг эхлүүлнэ! FreeBSD нь танд үүнийг хийж өгөх `ypinit` нэртэй скриптийг (холбогдох заавар хуудаснаас дэлгэрэнгүй мэдээллийг авна уу) агуулж байдаг. Энэ скрипт ихэнх UNIX® үйлдлийн системд байдаг боловч, заримд нь байхгүй байх тохиолдол бий. Digital UNIX/Compaq Tru64 UNIX дээр энэ скрипт `ypsetup` гэсэн нэртэй байдаг. Бид NIS эзэн серверийн хувьд буулгалтуудыг үүсгэж байгаа тул `ypinit` тушаалыг `-m` тохируулгын хамт өгнө. Дээрх алхмуудыг бүгдийг хийсний дараа, NIS буулгалтуудыг үүсгэхдээ дараах тушаалыг өгнө:

```
ellington# ypinit -m test-domain
Server Type: MASTER Domain: test-domain
Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.
Do you want this procedure to quit on non-fatal errors? [y/n: n] n
Ok, please remember to go back and redo manually whatever fails.
If you don't, something might not work.
At this point, we have to construct a list of this domains YP servers.
rod.darktech.org is already known as master server.
Please continue to add any slave servers, one per line. When you are
done with the list, type a <control D>.
master server   :  ellington
next host to add:  coltrane
next host to add:  ^D
The current list of NIS servers looks like this:
```

```
ellington
coltrane
Is this correct?  [y/n: y] y

[..output from map generation..]

NIS Map update completed.
ellington has been setup as an YP master server without any errors.
```

ypinit нь /var/yp/Makefile.dist-с /var/yp/Makefile-г үүсгэсэн байх ёстой. Үүсэхдээ, энэ файл таныг ганц NIS сервертэй орчинд зөвхөн FreeBSD машинуудтай ажиллаж байна гэж үзнэ. test-domain нь зарц сервертэй тул, та /var/yp/Makefile файлыг засах хэрэгтэй:

```
ellington# vi /var/yp/Makefile
```

Доорх мөрийг далдлах хэрэгтэй

```
NOPUSH = "True"
```

(хэрэв далдлагдаагүй бол).

#### 30.4.4.2.3. NIS Зарц Серверийг зохион байгуулах нь

NIS зарц серверийг зохион байгуулах нь эзэн серверийг зохион байгуулахаас ч хялбар байдаг. Зарц сервер рүү нэвтэрч ороод түрүүн хийсэн шигээ /etc/rc.conf файлыг засах хэрэгтэй. Ганц ялгаа нь ypinit тушаалыг өгөхдөө -s тохируулгыг өгнө. -s тохируулга нь NIS эзэн серверийн нэрийг хамт оруулахыг шаардах тул бидний тушаалын мөр дараах байдалтай байна:

```
coltrane# ypinit -s ellington test-domain

Server Type: SLAVE Domain: test-domain Master: ellington

Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.

Do you want this procedure to quit on non-fatal errors? [y/n: n]  n

Ok, please remember to go back and redo manually whatever fails.
If you don't, something might not work.
There will be no further questions. The remainder of the procedure
should take a few minutes, to copy the databases from ellington.
Transferring netgroup...
ypxfr: Exiting: Map successfully transferred
Transferring netgroup.byuser...
ypxfr: Exiting: Map successfully transferred
Transferring netgroup.byhost...
```

```
ypxfr: Exiting: Map successfully transferred
Transferring master.passwd.byuid...
ypxfr: Exiting: Map successfully transferred
Transferring passwd.byuid...
ypxfr: Exiting: Map successfully transferred
Transferring passwd.byname...
ypxfr: Exiting: Map successfully transferred
Transferring group.bygid...
ypxfr: Exiting: Map successfully transferred
Transferring group.byname...
ypxfr: Exiting: Map successfully transferred
Transferring services.byname...
ypxfr: Exiting: Map successfully transferred
Transferring rpc.bynumber...
ypxfr: Exiting: Map successfully transferred
Transferring rpc.byname...
ypxfr: Exiting: Map successfully transferred
Transferring protocols.byname...
ypxfr: Exiting: Map successfully transferred
Transferring master.passwd.byname...
ypxfr: Exiting: Map successfully transferred
Transferring networks.byname...
ypxfr: Exiting: Map successfully transferred
Transferring networks.byaddr...
ypxfr: Exiting: Map successfully transferred
Transferring netid.byname...
ypxfr: Exiting: Map successfully transferred
Transferring hosts.byaddr...
ypxfr: Exiting: Map successfully transferred
Transferring protocols.bynumber...
ypxfr: Exiting: Map successfully transferred
Transferring ypservers...
ypxfr: Exiting: Map successfully transferred
Transferring hosts.byname...
ypxfr: Exiting: Map successfully transferred

coltrane has been setup as an YP slave server without any errors.
Don't forget to update map ypservers on ellington.
```

Одоо /var/yp/test-domain нэртэй сан үүссэн байх ёстой. NIS эзэн серверийн буулгалтуудын хуулбарууд энэ сан дотор байх ёстой. Эдгээр файлууд шинэчлэгдэж байгаа эсэхийг нягтлаж байх хэрэгтэй. Таны зарц серверийн /etc/crontab доторх дараах мөрүүд үүнийг хийх болно:

```
20      *       *       *       *       root    /usr/libexec/ypxfr passwd.byname
21      *       *       *       *       root    /usr/libexec/ypxfr passwd.byuid
```

Энэ хоёр мөр нь зарц сервер өөрийн буулгалтуудыг эзэн сервертэй ижилхэн байлгахыг шаарддаг. Эзэн сервер буулгалтын өөрчлөлтийг өөрийн зарц нарт автоматаар оруулж өгөхийг оролддог болохоор эдгээр мөрүүдийг заавал хэрэглэх шаардлагагүй юм. Гэхдээ зарц

серверээс хамааралтай бусад клиентүүд дээрх зөв нууц үгийн мэдээллийн чухлаас хамааранд нууц үгийн буулгалтын шинэчлэлтийг давтамжтайгаар хийхийг зөвлөдөг. Буулгалтын шинэчлэлт үргэлж гүйцэд биш байж болох ачаалал их сүлжээний хувьд энэ нь илүүтэй чухал юм.

Одоо, зарц сервер талд мөн `/etc/netstart` тушаалыг өгч NIS серверийг ажиллуулна.

#### 30.4.4.3. NIS Харилцагчид

NIS харилцагч нь `ypbind` дэмоны тусламжтай тодорхой нэг NIS сервертэй холбоо тогтооно. `ypbind` системийн анхдагч домэйныг шалгах ба (`domainname` тушаалаар өгөгдсөн), дотоод сүлжээнд RPC хүсэлтийг цацаж эхлэнэ. Эдгээр хүсэлтүүд нь `ypbind`-н холбоо тогтоох гэж байгаа домэйн нэрийг зааж өгнө. Хэрэв тухайн домэйнд үйлчлэхээр тохируулагдсан сервер дээрх хүсэлтийг хүлээн авбал, `ypbind`-д хариу өгөх ба хариуг хүлээж авсан тал серверийн хаягийг тэмдэглэж авна. Хэрэв хэд хэдэн сервер хариу өгсөн бол (нэг эзэн ба хэд хэдэн зарц), `ypbind` хамгийн түрүүнд хариу өгсөн серверийг сонгон авна. Түүнээс хойш, харилцагч өөрийн бүх NIS хүсэлтүүдээ тэр сервер рүү явуулна. `ypbind` нь хааяа сервер амьд байгаа эсэхийг нягтлахын тулд "ping" хийж үзнэ. Хэрэв хангалттай хугацааны дотор хариу хүлээж аваагүй бол, `ypbind` энэ домэйнтой холбоо тасарлаа гэж үзээд өөр сервер олохын тулд хүсэлтээ цацаж эхэлнэ.

##### 30.4.4.3.1. NIS Харилцагчийг зохион байгуулах

FreeBSD машин дээр NIS харилцагчийг зохион байгуулах нь нилээд хялбар байдаг.

1. /etc/rc.conf файлыг нээгээд, NIS домэйн нэрийг зааж өгөх ба сүлжээ асах үед `ypbind`-г ажиллуулдаг болгохын тулд дараах мөрүүдийг нэмж бичнэ:

```
nisdomainname="test-domain"
nis_client_enable="YES"
```

2. NIS серверээс хэрэгтэй нэвтрэх үгүүдийг импортолж авахын тулд /etc/master.passwd файл дотор байгаа бүх хэрэглэгчийн дансыг устгаад, файлын төгсгөлд дараах мөрийг нэмэхийн тулд `vipw` тушаалыг ашиглана:

```
+::::::::::
```

Энэ мөр нь NIS серверийн нэвтрэх үгийн буулгалтад байгаа хүчинтэй хэрэглэгчид данс олгоно. Энэ мөрийг өөрчлөх замаар NIS харилцагчийг хэд хэдэн янзаар тохируулж болно. Дэлгэрэнгүй мэдээллийг доорх netgroups section хэсгээс үзнэ үү. Цааш гүнзгийрүүлэн судлах хүсэлтэй бол `NFS ба NIS-г удирдах нь` тухай O'Reilly-н номыг үзнэ үү.

Дор хаяж нэг дотоод эрхийг (өөрөөр хэлбэл NIS-с импортолж аваагүй) /etc/master.passwd файл дотор авч үлдэх хэрэгтэй. Энэ данс `wheel`

бүлгийн гишүүн байх ёстой. Хэрэв NIS дээр ямар нэг асуудал гарлаа гэхэд энэ эрхээр алсаас нэвтрэн орж, `root` болоод асуудлыг шийдвэрлэх болно.

3. NIS серверээс бүх бүлгүүдийг импортолж авахын тулд дараах мөрийг /etc/group файлд нэмнэ:

```
+:*::
```

NIS клиентийг нэн даруй эхлүүлэхийн тулд дараах тушаалыг супер хэрэглэгчийн эрхээр ажиллуулах хэрэгтэй:

```
# /etc/netstart
# service ypbind start
```

Үүний дараа, `ypcat passwd` тушаалыг өгч NIS серверийн passwd буулгалтыг харж чадаж байх ёстой.

### 30.4.5. NIS-н Аюулгүй байдал

Ер нь ямар ч алсын хэрэглэгчийн хувьд өөрийн чинь домэйн нэрийг мэдэж байвал RPC хүсэлтийг ypserv(8)-д явуулж NIS буулгалтыг харах боломжтой. Ийм төрлийн зөвшөөрөгдөөгүй үйлдлээс сэргийлэхийн тулд ypserv(8) нь зөвхөн зааж өгсөн хостуудаас ирсэн хандалтыг зөвшөөрдөг "securenets" гэсэн функцыг агуулж байдаг. Систем анх ачаалахад, ypserv(8) нь securenets-н мэдээллийг /var/yp/securenets гэсэн файлаас ачаална.

Энэ замыг `-p` тохируулгаар зааж өгөх ба янз бүр байж болно. Энэ файлд сүлжээг сүлжээний багийн хамт зайгаар тусгаарлан оруулж өгсөн байна. "#" тэмдгээр эхэлсэн мөрүүд нь тайлбар болно. Жишээ securenets файл дараах байдалтай байна:

```
# allow connections from local host -- mandatory
127.0.0.1     255.255.255.255
# allow connections from any host
# on the 192.168.128.0 network
192.168.128.0 255.255.255.0
# allow connections from any host
# between 10.0.0.0 to 10.0.15.255
# this includes the machines in the testlab
10.0.0.0      255.255.240.0
```

Хэрэв ypserv(8)-н хүсэлт хүлээж авсан хаяг эдгээр дүрмүүдийн аль нэгэнд тохирч байвал хүсэлтийг ердийн байдлаар боловсруулна. Хэрэв энэ хаяг ямар ч дүрмэнд тохирохгүй байвал, хүсэлтийг үл анхаарах бөгөөд анхааруулах бичлэгийг бүртгэлд нэмнэ. Хэрэв /var/yp/securenets гэсэн файл байхгүй бол, `ypserv` нь гаднаас ирсэн бүх хүсэлтийг хүлээн авна.

ypserv програм нь Wietse Venema-н TCP Wrapper багцыг дэмждэг. Энэ нь администраторуудын хувьд /var/yp/securenets-ны оронд TCP Wrapper-н тохиргооны файлыг хандалтыг хянахад хэрэглэх боломжтой болгодог.

Хэдийгээр эдгээр хандалтыг хянах механизмууд нь аюулгүй байдлыг адил түвшинд хангах боловч, хоёул "IP залилах" халдлагад өртөмтгий байдаг. NIS-тэй холбоотой бүх урсгалыг галт хана дээрээ хааж өгөх хэрэгтэй.

/var/yp/securenets хэрэглэж байгаа серверүүд хуучин TCP/IP дээр ажиллаж байгаа зүй ёсны NIS харилцагчид үйлчилж чадахгүй байж магадгүй. Учир нь, тэдгээр нь өргөн цацалт хийхдээ хост битүүдийг бүгдийг тэглэдэг ба өргөн цацалтын хаягийг тооцоолохдоо дэд сүлжээний багийг таньж чаддаггүй болно. Хэдийгээр эдгээр асуудлуудыг харилцагчийн тохиргоог өөрчилснөөр шийдэж болох боловч, бусад асуудлууд нь харилцагчийн системийг цааш ашиглах боломжгүй эсвэл /var/yp/securenets-г болиулах шаардлагатай болдог.

Ийм хуучин TCP/IP дээр ажилладаг сервер дээр /var/yp/securenets-г хэрэглэх нь үнэхээр хэрэггүй бөгөөд сүлжээний ихэнх хэсэгт NIS-г ашиглах боломжгүй байдаг.

TCP Wrapper багцыг ашиглах нь NIS серверийн хоцролтыг ихэсгэдэг. Энэ нэмэлт саатал нь харилцагчийн програм дээр ялангуяа ачаалал ихтэй сүлжээнд, эсвэл удаан NIS сервертэй бол хүлээх хугацаа дуусахад хүргэх талтай. Хэрэв таны харилцагч систем чинь дээрх шинж тэмдгүүдийн аль нэгийг агуулж байгаа бол та энэ харилцагч системээ NIS зарц сервер болгож өөрчлөн хүчээр өөрөөсөө өөртөө холбогдохоор тохируулах хэрэгтэй.

### 30.4.6. Зарим хэрэглэгчдийн нэвтрэхийг хаах

Манай лабораторын жишээн дээр, basie нэртэй нэг машин байгаа. Энэ машиныг зөвхөн багш нар хэрэглэх ёстой. Бид энэ машиныг NIS домэйн дотроос гаргахыг хүсэхгүй байгаа, дээр нь эзэн NIS сервер дээр байгаа passwd файл нь багш нар болон оюутнуудын дансыг хоёуланг агуулж байгаа. Бид одоо яах ёстой вэ?

NIS өгөгдлийн сан дотор бүртгэл нь байгаа ч, зарим хэрэглэгчдийг тухайн машин руу нэвтрэхийг хаах нэг арга байна. Үүний тулд -username гэсэн мөрийг бусад мөрүүдийн адил форматаар харилцагч машин дээр /etc/master.passwd файлын төгсгөлд нэмэх хэрэгтэй. Энд *username* гэдэг нь нэвтрэхийг нь хаах гэж байгаа хэрэглэгчийн нэр юм. Хаасан хэрэглэгчийн мөр + гэж нээсэн NIS хэрэглэгчийн мөрөөс дээр байх ёстой. Дээрх үйлдлийг хийхдээ vipw-г ашиглахыг зөвлөж байна. vipw нь /etc/master.passwd файл дотор хийгдсэн өөрчлөлтийг хянах бөгөөд өөрчлөлт хийж дууссаны дараа нэвтрэх үгийн санг автоматаар дахин үүсгэж өгдөг. Жишээ нь, хэрэв бид bill гэсэн хэрэглэгчийг basie хост дээр нэвтрэхийг хаахыг хүсэж байгаа бол:

```
basie# vipw
[add -bill::::::::: to the end, exit]
vipw: rebuilding the database...
```

```
vipw: done

basie# cat /etc/master.passwd

root:[password]:0:0::0:0:The super-user:/root:/bin/csh
toor:[password]:0:0::0:0:The other super-user:/root:/bin/sh
daemon:*:1:1::0:0:Owner of many system processes:/root:/sbin/nologin
operator:*:2:5::0:0:System &:/:/sbin/nologin
bin:*:3:7::0:0:Binaries Commands and Source,,,:/:/sbin/nologin
tty:*:4:65533::0:0:Tty Sandbox:/:/sbin/nologin
kmem:*:5:65533::0:0:KMem Sandbox:/:/sbin/nologin
games:*:7:13::0:0:Games pseudo-user:/usr/games:/sbin/nologin
news:*:8:8::0:0:News Subsystem:/:/sbin/nologin
man:*:9:9::0:0:Mister Man Pages:/usr/shared/man:/sbin/nologin
bind:*:53:53::0:0:Bind Sandbox:/:/sbin/nologin
uucp:*:66:66::0:0:UUCP pseudo-user:/var/spool/uucppublic:/usr/libexec/uucp/uucico
xten:*:67:67::0:0:X-10 daemon:/usr/local/xten:/sbin/nologin
pop:*:68:6::0:0:Post Office Owner:/nonexistent:/sbin/nologin
nobody:*:65534:65534::0:0:Unprivileged user:/nonexistent:/sbin/nologin
-bill
+:::::::::

basie#
```

### 30.4.7. Netgroups-г Хэрэглэх нь

Цөөхөн тооны машин эсвэл хэрэглэгчийн хувьд тусгай дүрэм хэрэгтэй үед өмнөх хэсэгт дурдсан аргыг хэрэглэх нь илүү тохиромжтой. Харин том сүлжээний хувьд зарим хэрэглэгчийн чухал машин руу нэвтрэх эрхийг хаахаа мартах, эсвэл бүх машиныг нэг бүрчлэн гараараа тохируулж өгөх, өөрөөр хэлбэл NIS-н *төвлөрсөн* удирдлага гэсэн гол санааг ашиглаж чадахгүй байх тохиолдлууд *гарах болно*.

NIS-г хөгжүүлэгчид энэ асуудлыг шийдэхийн тулд *netgroups* буюу сүлжээний бүлгүүд гэсэн шинэ зүйлийг бий болгожээ. Түүний зорилго болон семантикийг UNIX® файл системийн жирийн бүлэгтэй дүйцүүлж болох юм. Гол ялгаанууд нь гэвэл тоон дугаар байхгүй, мөн сүлжээний бүлгийг тодорхойлж өгөхдөө хэрэглэгч болон өөр сүлжээний бүлгийг оруулж болдог.

Сүлжээний бүлэг нь хэдэн зуун хэрэглэгч болон машинтай том, төвөгтэй сүлжээтэй ажиллахад зориулж бүтээгдсэн юм. Нэг талаар, хэрэв та үнэхээр тийм том сүлжээнд ажиллаж байгаа бол энэ нь Сайн Зүйл юм. Харин нөгөө талаас, энэ байдал нь жижигхэн сүлжээнд хялбар жишээн дээр сүлжээний бүлгийг тайлбарлах бараг боломжгүй болгож байна. Энэ хэсгийн үлдсэн хэсэгт хэрэглэж байгаа жишээн дээр энэ асуудлыг харуулахыг оролдлоо.

NIS-г лабораторидоо нэвтрүүлсэн тань танай удирдлагуудын анхаарлыг татсан гэж бодьё. Одоо оюутны хотхон дотор байгаа бусад машиныг NIS домэйнд оруулж өргөтгөх ажлыг хийхийг танд даалгажээ. Дараах хоёр хүснэгтэнд шинээр нэмэх хэрэглэгч болон машины нэрийг товч тайлбарын хамт үзүүллээ.

| Хэрэглэгчийн нэр | Тайлбар |
|---|---|
| alpha, beta | IT хэлтсийн ердийн ажилчид |
| charlie, delta | IT хэлтсийн шинэ дагалдан |
| echo, foxtrott, golf, ... | бусад ердийн ажилчид |
| able, baker, ... | дадлагажигчид |

| Машины нэр | Тайлбар |
|---|---|
| war, death, famine, pollution | Таны хамгийн чухал серверүүд. Зөвхөн IT хэлтсийн ажилчид л нэвтрэх эрхтэй. |
| pride, greed, envy, wrath, lust, sloth | Харьцангуй чухал биш серверүүд. IT хэлтэст харъяалагддаг бүх хүмүүс нэвтрэх эрхтэй. |
| one, two, three, four, ... | Ердийн ажлын машинууд. Зөвхөн *үндсэн* ажилчид нэвтрэх эрхтэй. |
| trashcan | Чухал зүйл байхгүй маш хуучин машин. Дадлагажигчид хүртэл нэвтрэх эрхтэй. |

Хэрэв та дээрх хязгаарлалтуудыг тус бүрд нь хэрэглэгчийг хаах замаар хийх гэж оролдвол бүх машин дээр хаах хэрэглэгч тус бүрийн хувьд `-user` мөрийг passwd файл дотор нэмж өгөх ёстой болно. Хэрэв нэг л мөрийг нэмэхээ мартвал асуудалд орно гэсэн үг.

Энэ байдалд сүлжээний бүлгийг ашиглах нь нилээд олон давуу талтай. Хэрэглэгч бүрийг тус тусад нь авч үзнэ; нэг хэрэглэгчийг нэг болон түүнээс дээш тооны сүлжээний бүлэгт оноож, тухайн сүлжээний бүлгийн бүх гишүүдийн хувьд нэвтрэхийг эсвэл зөвшөөрч эсвэл хаана. Хэрэв та шинэ машин нэмбэл, зөвхөн сүлжээний бүлгүүдийн хувьд л нэвтрэх эрхийг зааж өгнө. Хэрэв шинэ хэрэглэгч нэмбэл, тухайн хэрэглэгчийг нэг болон түүнээс дээш тооны сүлжээний бүлэгт нэмэхэд л хангалттай. Эдгээр өөрчлөлтүүд нь нэг нэгнээсээ хамааралгүй: "хэрэглэгч ба машины бүх хувилбарт нэмэх..." шаардлагагүй болно. Хэрэв та NIS-г анхнаас нь бодлоготой хийх юм бол, машинууд руу нэвтрэх эрхийг хянахдаа зөвхөн ганцхан тохиргооны файлыг өөрчлөхөд хангалттай.

Хамгийн эхний алхам бол NIS сүлжээний бүлгийн буулгалтыг эхлүүлэх юм. FreeBSD-н ypinit(8) нь энэ буулгалтыг анхдагч байдлаар үүсгэдэггүй, гэвч хэрэв нэгэнт үүсгэчихвэл түүний NIS-тэй ажиллах хэсэг нь энэ буулгалт дээр ажиллах чадвартай. Хоосон буулгалт үүсгэхийн тулд:

```
ellington# vi /var/yp/netgroup
```

гэж бичээд дараах зүйлсийг нэмж бичнэ. Манай жишээний хувьд, бидэнд дор хаяж дөрвөн сүлжээний бүлэг хэрэгтэй: IT ажилчид, IT дагалдангууд, ердийн ажилчид болон дадлагажигчид.

```
IT_EMP  (,alpha,test-domain)    (,beta,test-domain)
IT_APP  (,charlie,test-domain)  (,delta,test-domain)
USERS   (,echo,test-domain)     (,foxtrott,test-domain) \
```

```
        (,golf,test-domain)
INTERNS (,able,test-domain)     (,baker,test-domain)
```

`IT_EMP`, `IT_APP` гэх мэт нь сүлжээний бүлгийн нэр. Хаалтан дотор байгаа бүлэг нь хэрэглэгч нэмж байгаа нь. Бүлэг доторх гурван талбар нь:

1. Дараах зүйлүүд хүчинтэй байх хостын нэр. Хэрэв хостын нэр зааж өгөхгүй бол, бүх хостын хувьд хүчинтэй гэсэн үг. Хэрэв хостын нэр зааж өгвөл, та үл ойлгогдох, толгой эргүүлсэн хачин зүйлстэй тулгарах болно.
2. Энэ сүлжээний бүлэгт хамаарах дансны нэр.
3. Тухайн дансны NIS домэйн. Хэрэв та нэгээс олон NIS домэйнд харъяалагддаг азгүй залуусын нэг бол, өөрийн сүлжээний бүлэгт өөр NIS домэйноос данс импортолж болно.

Эдгээр талбаруудын алинд ч орлуулагддаг тэмдэгт ашиглаж болно. Дэлгэрэнгүй мэдээллийг netgroup(5) заавар хуудаснаас үзнэ үү.

Сүлжээний бүлгүүдийн нэр 8-с дээш тэмдэгт байж болохгүй, ялангуяа тухайн NIS домэйнд өөр үйлдлийн системтэй машинууд ажиллаж байгаа бол. Нэрүүд нь том жижиг үсгийн ялгаатай; сүлжээний бүлгийн нэрийг том үсгээр бичих нь хэрэглэгчийн нэр, машины нэр болон сүлжээний бүлгийн нэрийг хооронд нь ялгахад хялбар болгодог.

Зарим NIS харилцагчид (FreeBSD-с бусад) олон тооны гишүүдтэй сүлжээний бүлэгтэй ажиллаж чаддаггүй. Жишээлбэл, SunOS™-н зарим хуучин хувилбарууд сүлжээний бүлэг 15-с дээш тооны *гишүүн*-тэй бол асуудалтай байдаг. Энэ хязгаарыг давахын тулд 15 ба түүнээс доош тооны хэрэглэгчтэй дэд сүлжээний бүлгүүд үүсгээд, дараа нь эдгээр дэд сүлжээний бүлгүүдээс тогтсон жинхэнэ сүлжээний бүлэг үүсгэх замаар үүсгэж болно:

```
BIGGRP1  (,joe1,domain)  (,joe2,domain)  (,joe3,domain) [...]
BIGGRP2  (,joe16,domain)  (,joe17,domain) [...]
BIGGRP3  (,joe31,domain)  (,joe32,domain)
BIGGROUP  BIGGRP1 BIGGRP2 BIGGRP3
```

Хэрэв танд нэг сүлжээний бүлэгт 225-с дээш хэрэглэгч хэрэгтэй бол, дээрх үйлдлийг давтах маягаар цааш үргэлжлүүлж болно.

Шинээр үүсгэсэн NIS буулгалтаа идэвхжүүлэх болон тараах нь амархан:

```
ellington# cd /var/yp
ellington# make
```

Ингэснээр netgroup, netgroup.byhost ба netgroup.byuser гэсэн гурван NIS буулгалт үүсэх болно. Дээрх шинэ буулгалтууд идэвхтэй болсон эсэхийг ypcat(1) ашиглан шалгаарай:

```
ellington% ypcat -k netgroup
ellington% ypcat -k netgroup.byhost
ellington% ypcat -k netgroup.byuser
```

Эхний тушаалын үр дүн /var/yp/netgroup файл доторхтой төстэй байх ёстой. Хэрэв та хостоор тусгайлан сүлжээний бүлэг үүсгээгүй бол хоёр дахь тушаалын үр дүнд юу ч гарах ёсгүй. Гурав дахь тушаалын тусламжтай тухайн хэрэглэгчийн сүлжээний бүлгүүдийн жагсаалтыг харахад хэрэглэгдэнэ.

Харилцагчийг тохируулахад нилээд хялбар. `war` нэртэй серверийг тохируулахын тулд, vipw(8)-г ажиллуулаад

```
+:::::::::
```

гэсэн мөрийг

```
+@IT_EMP:::::::::
```

гэсэн мөрөөр сольж бичих хэрэгтэй. Ингэснээр, зөвхөн `IT_EMP` сүлжээний бүлэгт заагдсан хэрэглэгчдийн мэдээлэл `war`-н нэвтрэх үгийн санд импортлогдож, зөвхөн эдгээр хэрэглэгчид л энэ машин руу нэвтрэх эрхтэй боллоо.

Харамсалтай нь, энэ хязгаарлалт нь бүрхүүлийн `~` функцад, мөн хэрэглэгчийн нэр ба тоон дугаарыг хооронд нь хөрвүүлдэг бүх дэд программуудад хамаатай. Өөрөөр хэлбэл, `cd ~user` тушаал ажиллахгүй, `ls -l` тушаал хэрэглэгчийн нэрийн оронд түүний тоон дугаарыг харуулах ба `find . -user joe -print` тушаал `Тийм хэрэглэгч байхгүй` гэсэн алдааны мэдээлэл өгч амжилттүй болох болно. Үүнийг засахын тулд, бүх хэрэглэгчдийн бүртгэлийг *сервер рүү нэвтрэх эрхгүйгээр* импортлох хэрэгтэй болно.

Үүний тулд өөр нэг мөрийг /etc/master.passwd файлд нэмж өгөх хэрэгтэй. Энэ мөр нь:

`+::::::::::/sbin/nologin` гэсэн бичлэгийг агуулж байх ёстой бөгөөд, энэ нь "бүх бүртгэлийг импортол, гэхдээ импортлогдож байгаа бүртгэлүүдийн бүрхүүлийг /sbin/nologin-р соль" гэсэн утгатай. Үүнтэй адилаар `passwd` файлын ямар ч талбарыг /etc/master.passwd файл дахь анхдагч утгыг сольж бичсэнээр өөрчилж болно.

`::::::::::/sbin/nologin` гэсэн мөр `@IT_EMP:::::::::` гэсэн мөрийн дараа бичигдсэн эсэхийг сайтар нягтлаарай. Үгүй бол, NIS-с импортлогдсон бүх хэрэглэгчдийн бүрхүүл /sbin/nologin болчихно шүү.

Дээрх өөрчлөлтийг хийсний дараа, хэрэв IT хэлтэст шинэ ажилчин орвол, зөвхөн ганцхан NIS буулгалтыг өөрчлөх боллоо. Чухал бус бусад серверийн хувьд ижилхэн арга хэрэглэж, тэдгээрийн өөрийн /etc/master.passwd файл дотор байгаа хуучин `+:::::::::` мөрийг:

```
+@IT_EMP:::::::::
+@IT_APP:::::::::
```

```
+::::::::::/sbin/nologin
```

гэсэн мөрөөр сольж бичих хэрэгтэй. Ердийн ажлын машины хувьд:

```
+@IT_EMP:::::::::
+@USERS:::::::::
+::::::::::/sbin/nologin
```

байх ёстой. Ингээд бүх зүйл асуудалгүй ажиллах болно. Гэтэл хэдэн долоо хоногийн дараа дүрэм, журманд өөрчлөлт орлоо: IT хэлтэс дадлагажигч авч эхэллээ. IT хэлтсийн дадлагажигчид ердийн ажлын машин болон чухал бус серверүүдэд нэвтрэх эрхтэй; IT дагалдангууд гол сервер рүү нэвтрэх эрхтэй болжээ. Одоо `IT_INTERN` гэсэн шинэ сүлжээний бүлэг нэмж, энэ бүлэгт шинэ IT дадлагажигчдийг нэмээд, энэ өөрчлөлтийг бүх машины тохиргоонд оруулж эхлэх хэрэгтэй... Бидний хэлж заншсанаар: "Төвлөрсөн төлөвлөгөөн дээрх алдаа, бүх юмыг орвонгоор нь эргүүлнэ".

Энэ мэт тохиолдолуудад NIS-н өөр сүлжээний бүлгээс шинэ сүлжээний бүлэг үүсгэх боломж нь тус болно. Нэг боломж нь үүрэг дээр үндэслэсэн сүлжээний бүлэг юм. Жишээ нь, чухал серверүүд рүү нэвтрэх эрхийг хянахын тулд `BIGSRV` гэсэн нэртэй сүлжээний бүлэг үүсгэж болох ба, чухал бус серверүүдийн хувьд өөр `SMALLSRV` гэсэн бүрэг үүсгэж, `USERBOX` гэсэн гурав дахь бүлгийг ердийн ажлын машинуудад зориулж үүсгэж болох юм. Эдгээр сүлжээний бүлэг тус бүр дээрх гурван төрлийн машинд нэвтрэх эрхтэй сүлжээний бүлгүүдийг агуулна. NIS сүлжээний бүлгийн буулгалт дараах байдалтай байна:

```
BIGSRV    IT_EMP  IT_APP
SMALLSRV  IT_EMP  IT_APP  ITINTERN
USERBOX   IT_EMP  ITINTERN USERS
```

Нэвтрэх эрхийг хязгаарлах энэ арга нь ижил төрлийн хязгаарлалттай машинуудыг нэг бүлэг болговол илүү үр дүнтэй ажиллана. Харамсалтай нь, заавал тийм байх албагүй. Ихэнх тохиолдолд, машин тус бүрээр нэвтрэх эрхийг хязгаарлах боломжтой байх шаардлага зайлшгүй тулгардаг.

Машин дээр үндэслэсэн сүлжээний бүлэг тодорхойлох нь дээрх мэтийн дүрэм журамд өөрчлөлт ороход хэрэглэж болох хоёр дахь боломж юм. Энэ тохиолдолд, машин бүрийн /etc/master.passwd файл дотор "+"-р эхэлсэн хоёр мөр бичлэг байна. Эхнийх нь энэ машин руу нэвтрэх эрхтэй дансуудаас бүрдсэн сүлжээний бүлгийг нэмж өгнө, хоёр дахь нь бусад дансуудыг /sbin/nologin бүрхүүлтэйгээр нэмнэ. Сүлжээний бүлгийн нэрийг машины нэрийг "БҮХ ҮСГИЙГ ТОМООР" байхаар сонгож авах нь тохиромжтой. Өөрөөр хэлбэл, мөрүүд дараах байдалтай харагдах ёстой:

```
+@BOXNAME:::::::::
+::::::::::/sbin/nologin
```

Бүх машины хувьд дээрх үйлдлийг хийж дууссаны дараа, өөрийн /etc/master.passwd файлыг

дахин өөрчлөх шаардлагагүй болно. Бусад бүх өөрчлөлтүүдийг NIS буулгалтыг өөрчилснөөр шийдэх болно. Дээрх асуудалд тохирох сүлжээний бүлгийн буулгалтыг зарим нэмэлт өөрчлөлтүүдийн хамт дор жишээ болгож үзүүлэв:

```
# Define groups of users first
IT_EMP    (,alpha,test-domain)    (,beta,test-domain)
IT_APP    (,charlie,test-domain)  (,delta,test-domain)
DEPT1     (,echo,test-domain)     (,foxtrott,test-domain)
DEPT2     (,golf,test-domain)     (,hotel,test-domain)
DEPT3     (,india,test-domain)    (,juliet,test-domain)
ITINTERN  (,kilo,test-domain)     (,lima,test-domain)
D_INTERNS (,able,test-domain)     (,baker,test-domain)
#
# Now, define some groups based on roles
USERS     DEPT1   DEPT2     DEPT3
BIGSRV    IT_EMP  IT_APP
SMALLSRV  IT_EMP  IT_APP    ITINTERN
USERBOX   IT_EMP  ITINTERN  USERS
#
# And a groups for a special tasks
# Allow echo and golf to access our anti-virus-machine
SECURITY  IT_EMP  (,echo,test-domain)  (,golf,test-domain)
#
# machine-based netgroups
# Our main servers
WAR       BIGSRV
FAMINE    BIGSRV
# User india needs access to this server
POLLUTION  BIGSRV  (,india,test-domain)
#
# This one is really important and needs more access restrictions
DEATH     IT_EMP
#
# The anti-virus-machine mentioned above
ONE       SECURITY
#
# Restrict a machine to a single user
TWO       (,hotel,test-domain)
# [...more groups to follow]
```

Хэрэв та хэрэглэгчдийнхээ дансыг удирдахын тулд ямар нэг өгөгдлийн санг ашигладаг бол, дээрх буулгалтын эхний хэсгийг өгөгдлийн сангийнхаа тайлан бэлтгэх багажуудыг ашиглах үүсгэх боломжтой. Энэ замаар, шинэ хэрэглэгчид машинуудад хандах эрхийг автоматаар олж авах болно.

Эцэст нь анхааруулж хэлэх нэг зүйл байна: Машин дээр үндэслэсэн сүлжээний бүлгийг хэрэглэхийг байнга зөвлөхгүй. Хэрэв оюутны лабораторид зориулсан, хэдэн арван эсвэл хэдэн зуун нэг ижил машинтай ажиллаж байгаа бол, NIS буулгалтыг тодорхой хэмжээнд барьж байхын тулд машин дээр үндэслэсэн сүлжээний бүлгийн оронд үүрэг дээр үндэслэсэн

сүлжээний бүлгийг хэрэглэх хэрэгтэй.

### 30.4.8. Санаж явах чухал зүйлс

NIS орчинд ороод, өөрөөр хийх ёстой хэд хэдэн зүйлс байна.

- Лабораторид шинэ хэрэглэгч нэмэх бүрдээ *зөвхөн* эзэн NIS серверт нэмэх ёстой, ба *NIS буулгалтыг заавал дахин үүсгэх ёстой*. Хэрэв ингэхээ мартвал, шинэ хэрэглэгч эзэн NIS серверээс өөр хаашаа ч нэвтэрч чадахгүй болно. Жишээ нь, бид `jsmith` гэсэн шинэ хэрэглэгчийг лабораторид нэмэх боллоо:

  ```
  # pw useradd jsmith
  # cd /var/yp
  # make test-domain
  ```

  `pw useradd jsmith`-н оронд `adduser jsmith`-г мөн хэрэглэж болно.

- *Администратор эрхтэй дансуудыг NIS буулгалтад оруулах ёсгүй.* Администратор эрхээр орох ёсгүй хэрэглэгчдийн машин дээр администратор эрхтэй дансууд болон нэвтрэх үгүүдийг тараах хүсэлгүй байгаа биз дээ.
- *NIS эзэн болон зарц серверийн аюулгүй байдлыг хангаж, ажиллахгүй байх хугацааг багасгах хэрэгтэй.* Хэрэв хэн нэг нь серверт нууцаар нэвтэрч, эсвэл унтрааж орхивол хүмүүсийг лабораторын машинууд руу нэвтрэх боломжгүй болгож, саад болох болно.

  Энэ нь ямар ч төвлөрсөн удирдах системийн гол сул тал юм. Хэрэв та өөрийн NIS серверийг хамгаалахгүй бол, та маш олон ууртай хэрэглэгчидтэй таарах болно шүү!

### 30.4.9. NIS v1 нийцтэй байдал

FreeBSD-н ypserv нь NIS v1 харилцагчдад үйлчлэх зарим дэмжигчтэй ирдэг. FreeBSD-н NIS нь зөвхөн NIS v2 протоколыг хэрэглэдэг, гэхдээ бусад нь хуучин системүүдтэй нийцтэй ажиллахын тулд v1 протоколыг дэмждэг байхаар бүтээгдсэн байдаг. Эдгээр системтэй хамт ирсэн ypbind дэмонууд хэдийгээр үнэн хэрэг дээрээ хэзээ ч хэрэглэхгүй боловч NIS v1 сервертэй холболт үүсгэхийг оролддог (ба v2 серверээс хариу хүлээж авсан ч өргөн цацалт хийж хайлтаа үргэлжлүүлдэг талтай). Хэдийгээр ердийн харилцагчийн хүсэлтийг дэмждэг боловч, ypserv-н энэ хувилбар v1 буулгалтыг зөөх хүсэлттэй ажиллаж чадахгүй; иймээс, зөвхөн v1 протоколыг дэмждэг хуучин NIS серверүүдтэй холбоотойгоор эзэн эсвэл зарц байдлаар ажиллаж чадахгүй. Аз болоход, ийм серверийг одоо хэрэглэж байгаа газар байхгүй.

### 30.4.10. NIS Сервер мөртлөө NIS Харилцагч

Сервер машин нь мөн NIS харилцагч байдлаар ажилладаг олон сервертэй домэйнд ypserv-г ажиллуулахдаа анхааралтай байх хэрэгтэй. Ийм серверийг өргөн цацалт хийлгэж, өөр нэг сервертэй холбоо тогтоохыг зөвшөөрөхийн оронд өөрөө өөртэй нь хүчээр холбох нь ихэвчлэн дээр байдаг. Хэрэв нэг сервер унтарч, бусад серверүүд түүнээс хамааралтай байх юм бол хачин алдаанууд гарч болзошгүй. Эцэст нь бүх харилцагчдын хүлээх хугацаа дуусаж, бүгд өөр сервертэй холбогдохыг оролдох болно. Хэдийгээр бүх серверүүд

холболтуудаа сэргээж буцаад хэвийн байдалдаа орсон ч, саатлаас болж харилцагчид холбогдож чадахгүй хэвээр байх болно.

Хостыг ямар нэг сервертэй холбогдохыг `ypbind` тушаалыг `-S` тугийн хамт ажиллуулж, урдаас зааж өгч болно. Хэрэв NIS серверийг дахин ачаалах тоолонд энэ тушаалыг гараар оруулах хүсэлгүй байгаа бол, дараах мөрүүдийг өөрийн /etc/rc.conf файл дотор нэмээрэй:

```
nis_client_enable="YES" # run client stuff as well
nis_client_flags="-S NIS domain,server"
```

Дэлгэрэнгүй мэдээллийг ypbind(8) заавар хуудаснаас үзнэ үү.

### 30.4.11. Нэвтрэх үгийн хэлбэр

NIS-г зохион байгуулах явцад ихэвчлэн тохиолддог асуудлуудын нэг бол нэвтрэх үгийн хэлбэрийн нийцгүй байдал юм. Хэрэв таны NIS сервер DES хувиргалттай нэвтрэх үгийг хэрэглэдэг бол, зөвхөн DES хэрэглэдэг харилцагчид үйлчлэх чадвартай. Жишээлбэл, хэрэв сүлжээнд чинь Solaris™ NIS харилцагчид байгаа бол, та бараг л DES хувиргалттай нэвтрэх үг хэрэглэх шаардлагатай гэсэн үг.

Таны сервер болон харилцагчид ямар хэлбэрийн нэвтрэх үг хэрэглэдгийг шалгахдаа /etc/login.conf файлыг үзээрэй. Хэрэв тухайн хост DES хувиргалттай нэвтрэх үг хэрэглэдэг бол, `default` буюу анхдагч ангилал нь дараах мөрүүдийг агуулсан байх болно:

```
default:\
    :passwd_format=des:\
    :copyright=/etc/COPYRIGHT:\
    [Further entries elided]
```

`passwd_format` нь өөр `blf` ба `md5` гэсэн утгуудыг авч болно (Blowfish болон MD5 хувиргалттай нэвтрэх үгийн хувьд).

Хэрэв та /etc/login.conf файлд өөрчлөлт хийсэн бол, нэвтрэх чадварын санг дахин үүсгэх шаардлагатай. Үүний тулд дараах тушаалыг `root` эрхээр өгөх хэрэгтэй:

```
# cap_mkdb /etc/login.conf
```

/etc/master.passwd файл дотор аль хэдийн үүссэн нэвтрэх үгийн хэлбэр нь хэрэглэгч нэвтрэх чадварын сан дахин үүссэнээс хойш анх удаа нэвтрэх үгээ солих хүртэл өөрчлөгдөхгүй.

Мөн, таны сонгосон хэлбэрээр нэвтрэх үгүүдэд хувиргалт хийгддэг болгохын тулд, /etc/auth.conf файл доторх `crypt_default` утга таны сонгосон хэлбэрийг хамгийн түрүүнд оруулсан байгаа эсэхийг шалгах хэрэгтэй. Жишээ нь, DES хувиргалттай нэвтрэх үгийг хэрэглэх үед:

```
crypt_default   =   des blf md5
```

FreeBSD дээр тулгуурласан NIS сервер болон харилцагч бүр дээр дээрх үйлдлүүдийг хийснээр, нэвтрэх үгийн хэлбэр бүгд таарч байгаа гэдэгт санаа амар байж болно. Хэрэв NIS харилцагч дээр нэвтэрч ороход асуудал гарвал, асуудлыг тодруулах нэг газар байна. Хэрэв та холимог сүлжээний хувьд NIS сервер босгох гэж байгаа бол, ихэнх систем дээр зайлшгүй байх хамгийн бага стандарт тул, бүх системүүд дээрээ DES ашиглах хэрэгтэйг санаарай.

# 30.5. Автомат Сүлжээний Тохиргоо (DHCP)

## 30.5.1. DHCP гэж юу вэ?

DHCP, Dynamic Host Configuration Protocol буюу Динамик Хостын Тохиргооны Протокол нь систем ямар байдлаар сүлжээнд холбогдох, тухайн сүлжээнд харилцаанд орохын тулд шаардагдах мэдээллийг хэрхэн олж авахыг зааж өгдөг. FreeBSD нь OpenBSD 3.7-с авсан OpenBSD-н `dhclient`-г хэрэглэдэг. Энэ бүлэгт гарах `dhclient`-р ISC ба OpenBSD DHCP харилцагчийг хоёуланг нь төлөөлүүлсэн болно. DHCP серверийн хувьд ISC тархацын серверийг авч үзэх болно.

## 30.5.2. Энэ хэсэгт авч үзэх зүйлс

Энэ хэсэгт ISC ба OpenBSD DHCP харилцагчийн харилцагч талыг бүтээж байгаа элементүүд, болон ISC DHCP системийн сервер талыг бүтээж байгаа элементүүдийг хоёуланг нь авч үзэх болно. Харилцагч талын програм, `dhclient`, нь FreeBSD-тэй нэгдмэл байдлаар ирдэг бол, сервер талын хэсэг нь net/isc-dhcp42-server портоос суулгах боломжтой байдлаар ирдэг. dhclient(8), dhcp-options(5), ба dhclient.conf(5) заавар хуудсууд болон доор өгөгдсөн зөвлөмжүүд нь хэрэг болно.

## 30.5.3. Хэрхэн ажилладаг вэ?

Харилцагч машин дээр `dhclient` DHCP харилцагчийг ажиллуулахад, тохиргооны мэдээллийг хүссэн хүсэлтийг цацаж эхэлнэ. Анхдагч байдлаар, эдгээр хүсэлтүүд нь UDP 68-р портоос гарч, серверийн UDP 67 порт руу илгээгдэнэ. Сервер харилцагчид IP хаяг болон сүлжээний баг, чиглүүлэгч, DNS серверийн хаяг зэрэг хэрэгтэй мэдээллийг хариу илгээнэ. Энэ бүх мэдээллийг DHCP "түрээслэх" хэлбэрээр өгөх ба зөвхөн тодорхой хугацааны туршид хүчинтэй байна (DHCP серверийг хариуцагч тохируулж өгсөн байна). Ийм байдлаар, сүлжээнд холбогдохоо больсон харилцагчийн ашиглагдаагүй IP хаягуудыг автоматаар буцааж авах боломжтой болно.

DHCP харилцагч серверээс өргөн мэдээллийг авч чадна. Бүрэн жагсаалтыг dhcp-options(5)-с олж үзэж болно.

## 30.5.4. FreeBSD-тэй нэгдмэл байдал

FreeBSD нь OpenBSD DHCP харилцагч, `dhclient`-г өөртэйгөө бүрэн нэгтгэсэн байдаг. DHCP сервер ажиллаж байгаа сүлжээнд сүлжээний тохиргоог хийх нарийн чимхлүүр ажлаас хөнгөвчлөх үүднээс, DHCP харилцагчийг систем суулгагч болон үндсэн системийн аль

алинд хамт оруулж өгсөн байдаг.

sysinstall нь DHCP-г дэмждэг. sysinstall-р сүлжээний интерфэйсийг тохируулахад асуудаг хоёр дахь асуулт бол: "Та энэ интерфэйсийг DHCP-р тохируулахыг хүсэж байна уу?". Зөвшөөрсөн хариулт өгсөн тохиолдолд `dhclient`-г ажиллуулах бөгөөд, хэрэв амжилттай бол сүлжээний тохиргоо автоматаар хийгдэнэ.

Систем ачаалах үед DHCP ашигладаг болгохын тулд, хоёр зүйлийг хийх хэрэгтэй:

- bpf төхөөрөмж цөмтэй хамт эмхэтгэгдсэн байх ёстой. Үүний тулд, `device bpf` мөрийг цөмийн тохиргооны файлд нэмж бичээд цөмийг дахин бүтээх хэрэгтэй. Цөмийг бүтээх талаар дэлгэрэнгүй мэдээллийг FreeBSD цөмийг тохируулах нь хэсгээс авна уу.

  bpf төхөөрөмж нь FreeBSD-н GENERAL цөмийн нэг хэсэг бөгөөд, DHCP-г ажиллуулахын тулд тусгайлан шинээр цөм бүтээх шаардлагагүй.

  Аюулгүй байдлын талаар сэтгэл зовнидог хүмүүст зөвлөхөд, bpf нь пакет шиншлэгчдийг зөв ажиллах боломжийг олгодог төхөөрөмж болохыг анхааралдаа авна уу (хэдийгээр тэдгээр програм ажиллахын тулд `root` эрх хэрэгтэй боловч). DHCP-г ашиглахын тулд bpf*заавал* хэрэгтэй, гэвч хэрэв та аюулгүй байдлыг маш ихээр анхааралдаа авдаг бол, зөвхөн хэзээ нэгэн цагт DHCP-г ашиглахын тулд bpf-г цөмд нэмэх хэрэггүй.

- Анхдагчаар FreeBSD-н DHCP тохиргоо ар талд буюу асинхрон (*asynchronously*) горимд хийгддэг. DHCP дуустал бусад скриптүүд ажилладаг бөгөөд ингэснээр системийн эхлүүлэлтийг хурдасгадаг.

  Ард ажиллах DHCP нь DHCP сервер хүсэлтүүдэд хурдан хариу өгч DHCP тохиргооны процесс түргэн хийгдэх үед сайн ажилладаг. Гэхдээ DHCP зарим системүүд дээр хийгдэж дуустлаа удаан ажиллаж болно. DHCP дуусахаас өмнө сүлжээний үйлчилгээнүүд ажиллахаар оролдвол амжилттүй болно. DHCP-г синхрон (*synchronous*) горимд ашиглах нь DHCP тохиргоог дуустал эхлүүлэлтийг түр зогсоож асуудал гарахаас сэргийлдэг.

  Бусад эхлүүлэлтүүд үргэлжилж байх үед ар талд DHCP сервер рүү холбогдохын тулд (асинхрон горим) /etc/rc.conf файлд “DHCP” гэсэн утгыг ашиглана:

  ```
  ifconfig_fxp0="DHCP"
  ```

  DHCP дуустал эхлэлийг түр зогсоохын тулд синхрон горимыг “SYNCDHCP” утгатайгаар хэрэглэнэ:

  ```
  ifconfig_fxp0="SYNCDHCP"
  ```

  Сүлжээний интерфэйс картууд суулгах нь-д тайлбарласан ёсоор, эдгээр жишээн дээр байгаа `fxp0`-г динамикаар тохируулах гэж байгаа интерфэйсийн нэрээр сольж бичнэ.

Хэрэв таны `dhclient` өөр газар байгаа бол, эсвэл хэрэв та `dhclient`-г нэмэлт тугуудын хамт ажиллуулах хүсэлтэй бол, дараах мөрүүдийг нэмж бичнэ үү (шаардлагатай бол засаж бичнэ үү):

```
dhclient_program="/sbin/dhclient"
dhclient_flags=""
```

DHCP сервер dhcpd нь портуудын цуглуулгад байгаа net/isc-dhcp42-server портын нэг хэсэг байдлаар ирдэг. Энэ порт нь ISC DHCP сервер болон түүний баримтуудыг агуулсан байдаг.

### 30.5.5. Файлууд

- /etc/dhclient.conf

  `dhclient` нь /etc/dhclient.conf гэсэн тохиргооны файлыг шаарддаг. Ихэвчлэн энэ файл зөвхөн тайлбаруудаас бүрдэх ба анхдагч утгууд нь харьцангуй өөрчлөх шаардлагагүйгээр өгөгдсөн байдаг. Энэ тохиргооны файлыг dhclient.conf(5) заавар хуудсанд тайлбарласан байгаа.

- /sbin/dhclient

  `dhclient` нь статикаар холбогдсон байх ба /sbin дотор байрлана. dhclient(8) хуудаснаас `dhclient`-н талаар дэлгэрэнгүй мэдээллийг авна уу.

- /sbin/dhclient-script

  `dhclient-script` нь зөвхөн FreeBSD-д байдаг, DHCP харилцагчийг тохируулах зориулалттай тусгай скрипт юм. Энэ скриптийг dhclient-script(8) заавар хуудсанд тайлбарласан байх ба, ажиллуулахын тулд хэрэглэгч ямар нэг засвар хийх шаардлагагүй.

- /var/db/dhclient.leases.interface

  DHCP харилцагч нь түрээсэлж авсан хаягуудаа агуулсан өгөгдлийн санг энэ файлд хадгалах бөгөөд бүртгэл маягаар бичдэг. dhclient.leases(5) хэсэгт илүү дэлгэрэнгүй тайлбар бий.

### 30.5.6. Гүнзгийрүүлэн унших

DHCP протокол нь бүрэн хэмжээгээр RFC 2131-д тодорхойлогдсон байдаг. Нэмэлт эх үүсвэрүүд http://www.dhcp.org/-д мөн бий.

### 30.5.7. DHCP Серверийг Суулгах болон Тохируулах

#### 30.5.7.1. Энэ хэсэгт авч үзэх зүйлс

Энэ хэсэгт ISC (Internet Systems Consortium) DHCP серверийг ашиглан FreeBSD системийг хэрхэн DHCP сервер байдлаар ажиллуулах талаар авч үзэх болно.

Сервер нь FreeBSD-н нэг хэсэг байдлаар ирдэггүй бөгөөд ийм үйлчилгээ үзүүлэхийн тулд net/isc-dhcp42-server портыг суулгах хэрэгтэй болдог. Портуудын цуглуулгын хэрхэн ашиглах талаар Програм суулгах. Багцууд болон портууд хэсгээс дэлгэрэнгүй мэдээллийг авна уу.

#### 30.5.7.2. DHCP Серверийг суулгах нь

FreeBSD системийг DHCP сервер байдлаар тохируулахын тулд, bpf(4) төхөөрөмж цөмд эмхэтгэгдсэн байх ёстой. Үүний тулд, цөмийн тохиргооны файл дотор `bpf` `төхөөрөмж`ийг нэмээд цөмийг дахин бүтээх хэрэгтэй. Цөмийг бүтээх талаар дэлгэрэнгүй мэдээллийг FreeBSD цөмийг тохируулах нь хэсгээс үзнэ үү.

bpf төхөөрөмж нь FreeBSD-н GENERAL цөмийн нэг хэсэг бөгөөд, DHCP-г ажиллуулахын тулд тусгайлан шинээр цөм бүтээх шаардлагагүй.

Аюулгүй байдлын талаар сэтгэл зовнидог хүмүүст зөвлөхөд, bpf нь пакет шиншлэгчдийг зөв ажиллах боломжийг олгодог төхөөрөмж болохыг анхааралдаа авна уу (хэдийгээр тэдгээр програм ажиллахын тулд `root` эрх хэрэгтэй боловч). DHCP-г ашиглахын тулд bpf*заавал* хэрэгтэй, гэвч хэрэв та аюулгүй байдлыг маш ихээр анхааралдаа авдаг бол, зөвхөн хэзээ нэгэн цагт DHCP-г ашиглахын тулд bpf-г цөмд нэмэх хэрэггүй.

Үүний дараа net/isc-dhcp42-server порттой хамт ирсэн жишээ dhcpd.conf файлыг засах хэрэгтэй. Анхдагч байдлаар, /usr/local/etc/dhcpd.conf.sample гэсэн файл байх ба өөрчлөлт хийхийнхээ өмнө энэ файлыг /usr/local/etc/dhcpd.conf нэртэйгээр хуулж тавих хэрэгтэй.

#### 30.5.7.3. DHCP Серверийг тохируулах

dhcpd.conf нь дэд сүлжээ болон хостуудтай холбоотой өгөгдөл зарлалтаас бүрдэх ба жишээн дээр тайлбарлавал илүү амархан байх болов уу:

```
option domain-name "example.com";①
option domain-name-servers 192.168.4.100;②
option subnet-mask 255.255.255.0;③

default-lease-time 3600;④
max-lease-time 86400;⑤
ddns-update-style none;⑥

subnet 192.168.4.0 netmask 255.255.255.0 {
  range 192.168.4.129 192.168.4.254;⑦
  option routers 192.168.4.1;⑧
}

host mailhost {
  hardware ethernet 02:03:04:05:06:07;⑨
  fixed-address mailhost.example.com;⑩
}
```

① Энэ тохируулга нь анхдагч хайлтын домэйн байдлаар харилцагчид өгөх домэйныг заана.

Энэ талаар дэлгэрэнгүй мэдээллийг resolv.conf(5) хэсгээс үзнэ үү.

② Энэ тохируулга нь харилцагчийн хэрэглэх ёстой DNS серверүүдийг таслалаар холбосон жагсаалт байна.

③ Хэрэглэгчид өгөх сүлжээний багийг заана.

④ Түрээслэлт (lease) хүчинтэй байх тийм тусгай хугацааг харилцагч хүсэж болох юм. Хэрэв харилцагч хүсээгүй бол сервер энд заасан дуусах хугацаагаар (секундээр) түрээс хийх болно.

⑤ Серверийн түрээслүүлэх хамгийн дээд хугацааг заана. Харилцагч үүнээс урт хугацаагаар түрээслэх хүсэлт тавибал хүсэлтийг хүлээж авах боловч зөвхөн `max-lease-time` секундын туршид хүчинтэй байна.

⑥ Түрээслэх болон эргүүлж авахад DHCP сервер DNS-г шинэчлэхийг оролдох шаардлагатай эсэхийг зааж өгнө. ISC шийдлийн хувьд, энэ тохируулга *заавал* байх ёстой.

⑦ Харилцагчид оноох IP хаягуудын хүрээг заана. Энэ хүрээнд багтах IP хаягуудыг харилцагчид өгөх болно.

⑧ Харилцагчид өгөх анхдагч гарцыг заана.

⑨ Хостын MAC хаягийг заана (ингэснээр DHCP сервер тухайн хостыг хүсэлт тавихад таньж чадна).

⑩ Хостод тогтмол IP хаяг оноохыг заана. Энд хостын нэрийг хэрэглэж болохыг тэмдэглэх хэрэгтэй. DHCP сервер IP хаяг түрээслүүлэх хариуг өгөхөөс өмнө хост нэрийг тайлах болно.

dhcpd.conf файлыг бичиж дууссаны дараа, /etc/rc.conf файл дотор DHCP серверийг идэвхжүүлэх хэрэгтэй, өөрөөр хэлбэл доорх мөрүүдийг нэмж бичих хэрэгтэй:

```
dhcpd_enable="YES"
dhcpd_ifaces="dc0"
```

`dc0`-г өөрийн тань DHCP сервер DHCP харилцагчдын хүсэлтийг хүлээж авах ёстой интерфэйсийн нэрээр (эсвэл интерфэйсүүдийг зайгаар тусгаарлан) сольж бичих хэрэгтэй.

Дараа нь, доорх тушаалыг өгөн серверийг ажиллуулах хэрэгтэй:

```
# service isc-dhcpd start
```

Серверийнхээ тохиргооны файлд өөрчлөлт оруулах бүрдээ, `SIGHUP` дохиог dhcpd-д өгөх нь бусад дэмонуудын хувьд тохиргоог дахин дууддаг шиг биш харин тохиргоог дахин ачаалах_гүй_ болохыг анхаарах хэрэгтэй. Процессийг зогсоохын тулд `SIGTERM` дохиог өгөх хэрэгтэй ба дээрх тушаалыг өгөн дахин эхлүүлэх хэрэгтэй.

#### 30.5.7.4. Файлууд

- /usr/local/sbin/dhcpd

  dhcpd нь статикаар холбогдсон байх ба /usr/local/sbin дотор байрлана. Порттой хамт

суусан dhcpd(8) заавар хуудаснаас dhcpd-н талаар дэлгэрэнгүй мэдээллийг авна уу.

- /usr/local/etc/dhcpd.conf

  dhcpd нь /usr/local/etc/dhcpd.conf гэсэн тохиргооны файлыг шаарддаг. Энэ файл дотор харилцагчид өгөх бүх мэдээллээс гадна серверийн өөрийн үйл ажиллагаатай холбоотой мэдээлэл байх ёстой. Энэ тохиргооны файлыг портоос суусан dhcpd.conf(5) заавар хуудсанд тайлбарласан байгаа.

- /var/db/dhcpd.leases

  DHCP сервер нь түрээслүүлсэн хаягуудаа агуулсан өгөгдлийн санг энэ файлд хадгалах бөгөөд бүртгэл маягаар бичдэг. Портоос суусан dhcpd.leases(5) заавар хуудсанд илүү дэлгэрэнгүй тайлбар бий.

- /usr/local/sbin/dhcrelay

  dhcrelay-г нэг DHCP сервер харилцагчаас хүлээн авсан хүсэлтийг өөр сүлжээнд байгаа нөгөө DHCP сервер рүү дамжуулдаг, нарийн бүтэцтэй орчинд хэрэглэнэ. Хэрэв энэ функцыг ашиглах шаардлагатай бол, net/isc-dhcp42-relay портыг суулгаарай. Порттой хамт ирэх dhcrelay(8) заавар хуудаснаас дэлгэрэнгүй мэдээллийг авна уу.

# 30.6. Домэйн Нэрийн Систем (DNS)

## 30.6.1. Удиртгал

FreeBSD анхдагч байдлаар DNS протоколын хамгийн өргөн хэрэглэгддэг хэрэгжүүлэлт болох BIND (Berkeley Internet Name Domain)-н аль нэг хувилбарыг агуулсан байдаг. DNS нь нэрүүдийг IP хаягууд руу, мөн эсрэгээр нь буулгахад хэрэглэгддэг протокол юм. Жишээ нь, `www.FreeBSD.org`-г асуусан DNS асуулга явуулахад, хариуд нь FreeBSD Төсөлийн вэб серверийн IP хаяг ирэх бол, `ftp.FreeBSD.org`-н хувьд асуулга явуулахад, хариуд нь харгалзах FTP машины IP хаяг ирэх болно. Яг үүнтэй адилаар эсрэгээр нь хийж болно. Ямар нэг IP-р асуулга явуулахад түүний хост нэрийг олж болно. DNS хайлт хийхийн тулд тухайн системд домэйн нэрийн сервер ажиллаж байх ёстой.

FreeBSD нь одоо BIND9 DNS сервер програмын хамт ирдэг болсон. Бидний суулгац нь файл системийн шинэчилсэн зохион байгуулалт, автомат chroot(8) тохиргоо зэрэг аюулгүй байдлыг дээд зэргээр хангах функцүүдтэй ирдэг.

DNS бол Интернэт дээр тулгуурласан, бүрэн эрхт root буюу эх сервер, Top Level Domain буюу Дээд Түвшний Домэйн (TLD) сервер, болон домэйн тус бүрийн мэдээллийг агуулж байдаг бусад жижиг нэрийн серверүүдээс бүтсэн нарийн төвөгтэй систем юм.

BIND одоо Internet Systems Consortium http://www.isc.org/-н мэдэлд байдаг.

## 30.6.2. Нэр Томъёо

Энэ баримтыг ойлгохын тулд, DNS-тэй холбоотой зарим нэр томъёог ойлгосон байх шаардлагатай.

| Нэр | Тайлбар |
| --- | --- |
| Forward буюу Ердийн DNS | Хост нэрийг IP хаяг руу буулгана. |
| Origin буюу Үүсэл | Тухайн бүсийн файлд хамрагдаж байгаа домэйныг заана. |
| named, BIND | FreeBSD-н BIND нэрийн серверийг нэрлэх түгээмэл нэршил. |
| Resolver буюу Тайлагч | Машин, бүсийн мэдээллийн талаар нэрийн серверээс асуулга явуулахын тулд ашигладаг системийн процесс. |
| Reverse буюу Урвуу DNS | IP хаягийг хост нэр рүү буулгана. |
| Root zone буюу Эх бүс | Интернэт бүсийн шатлалын эхлэл. Файл системийн бүх файлууд эх санд харъяалагддаг шиг, бүх бүсүүд эх бүсэд харъяалагдана. |
| Zone буюу Бүс | Нэг бүрэн эрхт газраар удирдуулж байгаа домэйн, дэд домэйн, эсвэл DNS-н нэг хэсэг. |

Бүсүүдийн жишээ:

- `.` нь баримтад ихэвчлэн эх бүс гэж заагддаг.
- `org.` бол эх бүсийн доорх Top Level Domain буюу Дээд Түвшний Домэйн (TLD).
- `example.org.` бол `org.` TLD-н доорх бүс.
- `1.168.192.in-addr.arpa` бол `192.168.1.*` IP хаягийн хүрээнд багтаж байгаа бүх IP хаягуудыг агуулсан бүс.

Хост нэр зүүн тал руугаа явах тусам илүү тодорхой болж байгааг та бүхэн анзаарсан байх. Жишээлбэл, `example.org.` нь `org.`-с илүү тодорхой, харин `org.` нь эх бүсээс илүү тодорхой байна. Хост нэрийн зохион байгуулалт нь файл системийнхтэй төстэй: /dev директор нь эх директорт харъяалагдана, гэх мэт.

### 30.6.3. Нэрийн Сервер ажиллуулах Шалтгаанууд

Нэрийн Серверүүд ерөнхийдөө хоёр янз байна: authoritative буюу бүрэн эрхт нэрийн сервер, ба caching буюу түр тогтоогч нэрийн сервер.

Бүрэн эрхт нэрийн сервер нь дараах тохиолдлуудад хэрэгтэй:

- DNS мэдээллийг өөртөө агуулж, энэ мэдээллийг нийтэд зарлан, ирсэн асуулгуудад бүрэн эрхтэйгээр хариулах хүсэлтэй үед.
- Бүртгэлтэй домэйны хувьд, жишээлбэл `example.org`, түүний дор орших хост нэрүүдэд IP хаяг оноож өгөх хэрэгтэй үед.
- Бүлэг IP хаягуудад урвуу DNS мэдээлэл хэрэгтэй үед (IP-с хост нэр рүү).
- Нөөц эсвэл хоёрдогч нэрийн сервер, зарц гэж нэрлэнэ, асуулгуудад хариулуулах шаардлагатай үед.

Түр тогтоогч нэрийн сервер дараах тохиолдлуудад хэрэгтэй:

- Дотоод DNS сервер нь асуулгын хариуг түр тогтоосноор гадаад нэрийн серверээс илүү хурдан хариу өгч байгаа үед.

www.FreeBSD.org-р асуулга явуулсан үед, тайлагч ихэвчлэн үйлчилгээ авдаг ISP-нхаа нэрийн серверээс асуугаад хариуг олж авна. Дотоод, түр тогтоогч DNS сервер ажиллуулснаар, асуулгыг гадаад интернэтээс зөвхөн ганц удаа явуулах бөгөөд, хариуг тогтоож авна. Нэмэлт асуулгуудад түр тогтоогч нэрийн сервер хариулах ба гадагшаа дахин асуулга явуулах шаардлага байхгүй.

### 30.6.4. Хэрхэн ажилладаг вэ?

FreeBSD-д BIND дэмонг named гэж нэрлэнэ.

| Файл | Тайлбар |
|---|---|
| named(8) | BIND дэмон. |
| rndc(8) | Нэрийн серверийг хянах хэрэгсэл. |
| /etc/namedb | BIND-н бүсийн мэдээлэл хадгалагдаж байгаа сан. |
| /etc/namedb/named.conf | дэмоны тохиргооны файл. |

Тухайн бүс сервер дээр хэрхэн тохируулагдсанаас хамаарч энэ бүстэй хамааралтай файлууд /etc/namedb директорын master, slave, эсвэл dynamic гэсэн дэд сангуудад байрлана. Эдгээр файлуудад гадны асуулгад хариу болгон өгөх DNS мэдээллүүд байрлана.

### 30.6.5. BIND-г ажиллуулах нь

BIND нь анхдагч байдлаар суучихсан ирдэг тул тохируулахад хялбар байдаг.

named-н анхдагч тохиргоо нь chroot(8) орчинд ажиллах, тайлагч нэрийн сервер байдлаар хийгдсэн байдаг бөгөөд локал IPv4 loopback хаяг (127.0.0.1) дээр ажиллахаар хязгаарлагдсан байдаг. Энэ тохиргоогоор серверийг ажиллуулахын тулд дараах тушаалыг өгөх хэрэгтэй:

```
# service named onestart
```

named дэмонг систем ачаалах үед ажиллуулдаг болгохын тулд /etc/rc.conf дотор дараах мөрүүдийг нэмэх хэрэгтэй:

```
named_enable="YES"
```

Мэдээж /etc/namedb/named.conf файл дотор өөр олон тохируулгууд байгаа боловч энэ баримтын мэдлээс халих тул энд дурдсангүй. Хэрэв FreeBSD дээрх named-н эхлэл тохируулгуудын талаар сонирхож байгаа бол /etc/defaults/rc.conf дотор байгаа named_* тугуудыг нэг ороод үзээрэй. Мөн rc.conf(5) заавар хуудаснаас тусламж авч болно. FreeBSD

дээр rc(8) ашиглах нь хэсгийг уншихад илүүдэхгүй.

### 30.6.6. Тохиргооны файлууд

named-н тохиргооны файлууд нь /etc/namedb директор дотор байрлах ба хэрэв хялбар тайлагчаас өөр түвшинд ажиллах хэрэгтэй бол ажиллуулахаасаа өмнө тохиргооны файлд засвар хийх хэрэгтэй. Ихэнх тохиргоог энэ сан дотор гүйцэтгэнэ.

#### 30.6.6.1. /etc/namedb/named.conf

```
// $FreeBSD$
//
// Refer to the named.conf(5) and named(8) man pages, and the documentation
// in /usr/shared/doc/bind9 for more details.
//
// If you are going to set up an authoritative server, make sure you
// understand the hairy details of how DNS works.  Even with
// simple mistakes, you can break connectivity for affected parties,
// or cause huge amounts of useless Internet traffic.

options {
    // All file and path names are relative to the chroot directory,
    // if any, and should be fully qualified.
    directory   "/etc/namedb/working";
    pid-file    "/var/run/named/pid";
    dump-file   "/var/dump/named_dump.db";
    statistics-file "/var/stats/named.stats";

// If named is being used only as a local resolver, this is a safe default.
// For named to be accessible to the network, comment this option, specify
// the proper IP address, or delete this option.
    listen-on   { 127.0.0.1; };

// If you have IPv6 enabled on this system, uncomment this option for
// use as a local resolver.  To give access to the network, specify
// an IPv6 address, or the keyword "any".
//  listen-on-v6    { ::1; };

// These zones are already covered by the empty zones listed below.
// If you remove the related empty zones below, comment these lines out.
    disable-empty-zone "255.255.255.255.IN-ADDR.ARPA";
    disable-empty-zone
"0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.IP6.ARPA";
    disable-empty-zone
"1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.IP6.ARPA";

// If you've got a DNS server around at your upstream provider, enter
// its IP address here, and enable the line below.  This will make you
// benefit from its cache, thus reduce overall DNS traffic in the Internet.
/*
```

```
    forwarders {
        127.0.0.1;
    };
*/

// If the 'forwarders' clause is not empty the default is to 'forward first'
// which will fall back to sending a query from your local server if the name
// servers in 'forwarders' do not have the answer.  Alternatively you can
// force your name server to never initiate queries of its own by enabling the
// following line:
//  forward only;

// If you wish to have forwarding configured automatically based on
// the entries in /etc/resolv.conf, uncomment the following line and
// set named_auto_forward=yes in /etc/rc.conf.  You can also enable
// named_auto_forward_only (the effect of which is described above).
//  include "/etc/namedb/auto_forward.conf";
```

Тайлбар дээр хэлсэнчлэн дээд гарцын түр тогтоогчоос хүртэхийн тулд `forwarders`-г идэвхжүүлж болох юм. Энгийн үед, нэрийн сервер нь хариултыг олтлоо давталттай байдлаар хэд хэдэн нэрийн серверүүдээр дамжин асууна. Энэ тохируулгыг идэвхжүүлснээр, дээд гарцынхаа нэрийн серверээс (эсвэл зааж өгсөн нэрийн сервер) хамгийн түрүүнд асууж, энэ серверийн түр санах ойд байгаа мэдээллээс хүртэхийг эрмэлзэнэ. Хэрэв дээд гарцын нэрийн сервер нь олон асуулгад хариулдаг, хурдан үйлчилдэг сервер байвал дээрх тохируулгыг идэвхжүүлсний үр ашиг гарна.

`127.0.0.1` энд ажиллах_гүй_. Энэ IP хаягийг өөрийн дээд гарцын нэрийн серверээр сольж бичнэ үү.

```
    /*
       Modern versions of BIND use a random UDP port for each outgoing
       query by default in order to dramatically reduce the possibility
       of cache poisoning.  All users are strongly encouraged to utilize
       this feature, and to configure their firewalls to accommodate it.

       AS A LAST RESORT in order to get around a restrictive firewall
       policy you can try enabling the option below.  Use of this option
       will significantly reduce your ability to withstand cache poisoning
       attacks, and should be avoided if at all possible.

       Replace NNNNN in the example with a number between 49160 and 65530.
    */
    // query-source address * port NNNNN;
};

// If you enable a local name server, don't forget to enter 127.0.0.1
// first in your /etc/resolv.conf so this server will be queried.
// Also, make sure to enable it in /etc/rc.conf.
```

```
// The traditional root hints mechanism. Use this, OR the slave zones below.
zone "." { type hint; file "/etc/namedb/named.root"; };

/*  Slaving the following zones from the root name servers has some
    significant advantages:
    1. Faster local resolution for your users
    2. No spurious traffic will be sent from your network to the roots
    3. Greater resilience to any potential root server failure/DDoS

    On the other hand, this method requires more monitoring than the
    hints file to be sure that an unexpected failure mode has not
    incapacitated your server.  Name servers that are serving a lot
    of clients will benefit more from this approach than individual
    hosts.  Use with caution.

    To use this mechanism, uncomment the entries below, and comment
    the hint zone above.

    As documented at http://dns.icann.org/services/axfr/ these zones:
    "." (the root), ARPA, IN-ADDR.ARPA, IP6.ARPA, and ROOT-SERVERS.NET
    are availble for AXFR from these servers on IPv4 and IPv6:
    xfr.lax.dns.icann.org, xfr.cjr.dns.icann.org
*/
/*
zone "." {
    type slave;
    file "/etc/namedb/slave/root.slave";
    masters {
        192.5.5.241;    // F.ROOT-SERVERS.NET.
    };
    notify no;
};
zone "arpa" {
    type slave;
    file "/etc/namedb/slave/arpa.slave";
    masters {
        192.5.5.241;    // F.ROOT-SERVERS.NET.
    };
    notify no;
};
*/

/*  Serving the following zones locally will prevent any queries
    for these zones leaving your network and going to the root
    name servers.  This has two significant advantages:
    1. Faster local resolution for your users
    2. No spurious traffic will be sent from your network to the roots
*/
// RFCs 1912 and 5735 (and BCP 32 for localhost)
zone "localhost"    { type master; file "/etc/namedb/master/localhost-forward.db"; };
zone "127.in-addr.arpa" { type master; file "/etc/namedb/master/localhost-reverse.db";
```

```
};
zone "255.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// RFC 1912-style zone for IPv6 localhost address
zone "0.ip6.arpa"   { type master; file "/etc/namedb/master/localhost-reverse.db"; };

// "This" Network (RFCs 1912 and 5735)
zone "0.in-addr.arpa"   { type master; file "/etc/namedb/master/empty.db"; };

// Private Use Networks (RFCs 1918 and 5735)
zone "10.in-addr.arpa"     { type master; file "/etc/namedb/master/empty.db"; };
zone "16.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "17.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "18.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "19.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "20.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "21.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "22.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "23.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "24.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "25.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "26.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "27.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "28.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "29.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "30.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "31.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "168.192.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// Link-local/APIPA (RFCs 3927 and 5735)
zone "254.169.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IETF protocol assignments (RFCs 5735 and 5736)
zone "0.0.192.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// TEST-NET-[1-3] for Documentation (RFCs 5735 and 5737)
zone "2.0.192.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "100.51.198.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "113.0.203.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Range for Documentation (RFC 3849)
zone "8.b.d.0.1.0.0.2.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// Domain Names for Documentation and Testing (BCP 32)
zone "test" { type master; file "/etc/namedb/master/empty.db"; };
zone "example" { type master; file "/etc/namedb/master/empty.db"; };
zone "invalid" { type master; file "/etc/namedb/master/empty.db"; };
zone "example.com" { type master; file "/etc/namedb/master/empty.db"; };
zone "example.net" { type master; file "/etc/namedb/master/empty.db"; };
zone "example.org" { type master; file "/etc/namedb/master/empty.db"; };
```

```
// Router Benchmark Testing (RFCs 2544 and 5735)
zone "18.198.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "19.198.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IANA Reserved - Old Class E Space (RFC 5735)
zone "240.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "241.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "242.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "243.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "244.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "245.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "246.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "247.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "248.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "249.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "250.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "251.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "252.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "253.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "254.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Unassigned Addresses (RFC 4291)
zone "1.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "3.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "4.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "5.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "6.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "7.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "8.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "9.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "a.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "b.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "c.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "d.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "e.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "0.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "1.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "2.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "3.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "4.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "5.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "6.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "7.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "8.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "9.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "a.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "b.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "0.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "1.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "2.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "3.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
```

```
zone "4.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "5.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "6.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "7.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 ULA (RFC 4193)
zone "c.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "d.f.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Link Local (RFC 4291)
zone "8.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "9.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "a.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "b.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Deprecated Site-Local Addresses (RFC 3879)
zone "c.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "d.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "e.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };
zone "f.e.f.ip6.arpa"   { type master; file "/etc/namedb/master/empty.db"; };

// IP6.INT is Deprecated (RFC 4159)
zone "ip6.int"      { type master; file "/etc/namedb/master/empty.db"; };

// NB: Do not use the IP addresses below, they are faked, and only
// serve demonstration/documentation purposes!
//
// Example slave zone config entries.  It can be convenient to become
// a slave at least for the zone your own domain is in.  Ask
// your network administrator for the IP address of the responsible
// master name server.
//
// Do not forget to include the reverse lookup zone!
// This is named after the first bytes of the IP address, in reverse
// order, with ".IN-ADDR.ARPA" appended, or ".IP6.ARPA" for IPv6.
//
// Before starting to set up a master zone, make sure you fully
// understand how DNS and BIND work.  There are sometimes
// non-obvious pitfalls.  Setting up a slave zone is usually simpler.
//
// NB: Don't blindly enable the examples below. :-)  Use actual names
// and addresses instead.

/* An example dynamic zone
key "exampleorgkey" {
    algorithm hmac-md5;
    secret "sf87HJqjkqh8ac87a02lla==";
};
zone "example.org" {
    type master;
    allow-update {
```

```
        key "exampleorgkey";
    };
    file "/etc/namedb/dynamic/example.org";
};
*/

/* Example of a slave reverse zone
zone "1.168.192.in-addr.arpa" {
    type slave;
    file "/etc/namedb/slave/1.168.192.in-addr.arpa";
    masters {
        192.168.1.1;
    };
};
*/
```

named.conf доторх эдгээр жишээнүүд нь ердийн болон урвуу бүсийн зарц бүртгэлүүд болно.

Шинэ бүс нэмэхдээ, named.conf файл дотор шинэ бүртгэл оруулах хэрэгтэй.

Жишээ нь, `example.org` домэйны хувьд хамгийн хялбар бүртгэл дараах байдалтай байна:

```
zone "example.org" {
    type master;
    file "master/example.org";
};
```

Энэ бүс нь эзэн бүс болохыг `type` илэрхийллээс харж болно. Мөн бүсийн мэдээллийг /etc/namedb/master/example.org файл дотор агуулж байгааг `file` илэрхийллээс харж болно.

```
zone "example.org" {
    type slave;
    file "slave/example.org";
};
```

Зарц бүсийн хувьд, тухайн бүсийн хувьд бүсийн мэдээлэл эзэн нэрийн серверээс зөөгдөж ирэх ба зааж өгсөн файлд хадгалагдана. Эзэн сервер унтарсан эсвэл холбоо тогтоох боломжгүй болбол, зарц нэрийн серверт бүсийн мэдээлэл байгаа тул асуулгуудад хариулах чадвартай байна.

#### 30.6.6.2. Бүсийн Файлууд

`example.org` домэйны хувьд жишээ эзэн бүсийн файлыг дор үзүүлэв (/etc/namedb/master/example.org файл):

```
$TTL 3600        ; 1 hour default TTL
example.org.    IN      SOA     ns1.example.org. admin.example.org. (
```

```
                              2006051501      ; Serial
                              10800           ; Refresh
                              3600            ; Retry
                              604800          ; Expire
                              300             ; Negative Response TTL
                        )

; DNS Servers
                IN      NS      ns1.example.org.
                IN      NS      ns2.example.org.

; MX Records
                IN      MX 10   mx.example.org.
                IN      MX 20   mail.example.org.

                IN      A       192.168.1.1

; Machine Names
localhost       IN      A       127.0.0.1
ns1             IN      A       192.168.1.2
ns2             IN      A       192.168.1.3
mx              IN      A       192.168.1.4
mail            IN      A       192.168.1.5

; Aliases
www             IN      CNAME   example.org.
```

"." тэмдэгтээр төгссөн хост нэрүүд нь жинхэнэ хост нэрүүд бөгөөд "." тэмдэгтээр төгсөөгүй нэрүүдэд үүсэл залгагдахыг анхаарна уу. Жишээлбэл, `ns1` нь `ns1.example.org.`-руу хөрвүүлэгдэх болно.

Бүсийн файл дараах хэлбэртэй байна:

```
recordname      IN recordtype   value
```

Хамгийн өргөн хэрэглэгддэг DNS бичлэгүүд:

**SOA**

start of zone authority буюу бүсийн бүрэн эрхт мэдээллийн эхлэл

**NS**

бүрэн эрхт нэрийн сервер

**A**

хостын хаяг

**CNAME**

хуурамч дүрд өгөх хүлээн зөвшөөрөгдсөн нэр

**MX**

захидал солилцогч

**PTR**

домэйн нэрийг заагч (урвуу DNS-д хэрэглэнэ)

```
example.org. IN SOA ns1.example.org. admin.example.org. (
                        2006051501      ; Serial
                        10800           ; Refresh after 3 hours
                        3600            ; Retry after 1 hour
                        604800          ; Expire after 1 week
                        300 )           ; Negative Response TTL
```

`example.org.`

домэйн нэр, мөн энэ бүсийн файлын хувьд үүсэл болно.

`ns1.example.org.`

энэ бүсийн гол/бүрэн эрхт нэрийн сервер.

`admin.example.org.`

энэ бүсийг хариуцагч хүн, "@" тэмдэгтийг нь орлуулсан цахим захидлын хаяг. (admin@example.org нь `admin.example.org` болно)

`2006051501`

Файлын сериал дугаар. Бүсийн файлд өөрчлөлт оруулах болгонд энэ дугаарыг нэмэгдүүлэх шаардлагатай. Одоо цагт ихэнх администууд энэ сериал дугаарыг `yyyymmddrr` хэлбэрээр хэрэглэх болсон. `2006051501` гэдэг нь хамгийн сүүлд 05/15/2006-нд засвар хийсэн, хамгийн сүүлийн `01` гэдэг нь энэ өдөр хийгдсэн хамгийн анхны засвар гэдгийг илтгэнэ. Энэ сериал дугаар нь зарц серверүүдэд бүсийн мэдээлэл өөрчлөгдсөн талаар мэдээлэл өгдөг тул их чухал зүйл байгаа юм.

```
       IN NS           ns1.example.org.
```

Энэ бол NS бичлэг. Тухайн бүсийн хувьд бүрэн эрхт хариултыг өгч чадах сервер бүрийн хувьд энэ бичлэг байх ёстой.

```
localhost       IN      A       127.0.0.1
ns1             IN      A       192.168.1.2
ns2             IN      A       192.168.1.3
mx              IN      A       192.168.1.4
mail            IN      A       192.168.1.5
```

A бичлэг нь машины нэрийг заана. Дээр үзүүлсэнчлэн, `ns1.example.org` нь `192.168.1.2`-руу буулгагдана.

```
                IN      A       192.168.1.1
```

Энэ мөр нь 192.168.1.1 гэсэн IP хаягийг үүсэлд оноож байна, бидний жишээн дээр example.org.

```
www             IN CNAME        @
```

Хүлээн зөвшөөрөгдсөн нэрийн бичлэг нь машинд хуурамч дүр өгөхөд хэрэглэгдэнэ. Энэ жишээн дээр, www нь example.org (192.168.1.1) гэсэн домэйн нэртэй "master" машины хуурамч дүрийн нэр юм. CNAME-г тухайн хостын нэрийн хувьд өөр төрлийн бичлэгтэй хэзээ ч цуг хэрэглэж болохгүй.

```
                IN MX   10      mail.example.org.
```

MX бичлэг нь аль захидлын серверүүд тухайн бүсийн захидлыг хүлээж авах үүрэгтэй болохыг зааж өгнө. mail.example.org нь захидлын серверийн хост нэр бөгөөд 10 нь энэ захидлын серверийн зэрэглэлийг зааж байна.

Нэг бүсэд 10, 20 гэх мэт ялгаатай зэрэглэлтэй хэд хэдэн захидлын сервер байж болно. example.org домэйн руу захидал явуулах гэж байгаа сервер эхлээд хамгийн өндөр зэрэглэлтэй MX сервертэй (хамгийн бага зэрэглэлийн дугаартай), дараа нь дараагийн хамгийн өндөр зэрэглэлтэй сервертэй гэх мэтчилэн захидлыг явуулж чадтал дарааллаар нь холбоо тогтооно.

in-addr.arpa бүсийн файл (урвуу DNS) нь ижил хэлбэртэй байна. Ганцхан ялгаа нь A болон CNAME бичлэгийн оронд PTR бичлэгийг хэрэглэнэ.

```
$TTL 3600

1.168.192.in-addr.arpa. IN SOA ns1.example.org. admin.example.org. (
                        2006051501      ; Serial
                        10800           ; Refresh
                        3600            ; Retry
                        604800          ; Expire
                        300 )           ; Negative Response TTL

        IN      NS      ns1.example.org.
        IN      NS      ns2.example.org.

1       IN      PTR     example.org.
2       IN      PTR     ns1.example.org.
3       IN      PTR     ns2.example.org.
4       IN      PTR     mx.example.org.
5       IN      PTR     mail.example.org.
```

Энэ файлд дээрх домэйны IP-с хост нэр рүү буулгасан зохих шаардлагатай буулгалтуудыг

үзүүлсэн байна.

PTR бичлэгийн баруун талын бүх нэрс төгссөн байх ёстой (өөрөөр хэлбэл "."-ээр төгссөн байна).

### 30.6.7. Түр тогтоогч Нэрийн Сервер (Caching Name Server)

Түр тогтоогч нэрийн сервер гэдэг нь рекурсив хүсэлтэд хариу өгөх гол үүрэгтэй нэрийн серверийг хэлнэ. Ийм төрлийн сервер нь зөвхөн асуулга явуулах бөгөөд хариултыг дараа хэрэглэхээр тогтоож авдаг.

### 30.6.8. DNSSEC

Домэйн Нэрийн Системийн Аюулгүй байдлын Өргөтгөлүүд, товчоор DNSSEC, бол нэр тайлагч серверүүдийг залилуулсан DNS бичлэг гэх мэт хуурамч DNS өгөгдлөөс хамгаалах зааврууд иж бүрдэл юм. Электрон гарын үсгийн тусламжтай нэр тайлагч нь бичлэгийн бүрэн бүтэн байдлыг магадлах боломжтой. DNSSEC нь зөвхөн Боломжит Бичлэгүүд дээр (RRs) электрон гарын үсэг зурах замаар өгөгдлийн бүрэн бүтэн байдлыг хангадаг болохыг тэмдэглэн хэлье. Нууцлалыг хангаж, эцсийн хэрэглэгчийн буруу үйлдлээс хамгаалж чадахгүй. Өөрөөр хэлбэл хүмүүсийг `example.com`-н оронд `example.net`-руу орохыг болиулж чадахгүй гэсэн үг юм. DNSSEC-н хийж чадах ганц зүйл бол өгөгдөл замдаа хувиралгүйгээр очсоныг магадлан тогтоох юм. DNS-н аюулгүй байдал бол Интернэтийн аюулгүй байдлыг хангахад чухал алхам болдог. DNSSEC хэрхэн ажилладаг талаар дэлгэрэнгүй мэдээллийг тухайн RFC-үүдээс аваарай. Гүнзгийрүүлэн унших-д байгаа жагсаалтыг үзнэ үү.

Дараах бүлгүүдэд BIND 9 ажиллаж байгаа бүрэн эрхт DNS сервер болон рекурсив (эсвэл түр тогтоогч) DNS сервер дээр DNSSEC-г хэрхэн идэвхжүүлэхийг үзүүлэх болно. BIND 9-н бүх хувилбарууд DNSSEC-г дэмжих боловч, DNS асуулгуудын хүчинтэй эсэхийг шалгахад гарын үсэгтэй эх бүсийг ашиглахын тулд хамгийн багадаа 9.6.2 хувилбарыг суулгах шаардлагатай. Яагаад гэвэл өмнөх хувилбаруудад эх (root) бүсийн түлхүүрийг ашиглах шалгалтыг идэвхжүүлэхэд шаардлагатай алгоритмууд байдаггүй. Эх түлхүүрт зориулж автоматаар түлхүүрийг шинэчлэх боломж болон автоматаар бүсүүдийг гарын үсгээр баталгаажуулж гарын үсгүүдийг байнга шинэ байлгахын тулд BIND-ийн хамгийн сүүлийн хувилбар 9.7 юм уу эсвэл түүний дараагийн хувилбарыг ашиглахыг шаарддаг. 9.6.2 болон 9.7 болон түүнээс хойшхи хувилбаруудын хооронд тохиргооны зөрүү байвал харуулах болно.

#### 30.6.8.1. Рекурсив DNS серверийн тохиргоо

Рекурсив DNS серверийн гүйцэтгэсэн хүсэлтүүдийн DNSSEC шалгалтыг идэвхжүүлэхийн тулд named.conf файлд цөөн өөрчлөлтийг хийх хэрэгтэй. Эдгээр өөрчлөлтүүдийг хийхээс өмнө эх бүсийн түлхүүр эсвэл итгэлцлийн анкорыг (anchor) авсан байх шаардлагатай. Одоогоор эх бүсийн түлхүүр нь BIND ойлгох файлын форматаар байдаггүй бөгөөд зөв хэлбэр рүү гараар хувиргах ёстой байдаг. Түлхүүрийг dig ашиглан эх бүсээс асууж авч болдог. Ингэхийн тулд

```
% dig +multi +noall +answer DNSKEY . > root.dnskey
```

гэж ажиллуулна. Түлхүүр root.dnskey файлд байх болно. Доторх нь иймэрхүү байдалтай

харагдана:

```
. 93910 IN DNSKEY 257 3 8 (
    AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQ
    bSEW0O8gcCjFFVQUTf6v58fLjwBd0YI0EzrAcQqBGCzh
    /RStIoO8g0NfnfL2MTJRkxoXbfDaUeVPQuYEhg37NZWA
    JQ9VnMVDxP/VHL496M/QZxkjf5/Efucp2gaDX6RS6CXp
    oY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3
    LQpzW5hOA2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGIcGO
    Yl7OyQdXfZ57relSQageu+ipAdTTJ25AsRTAoub8ONGc
    LmqrAmRLKBP1dfwhYB4N7knNnulqQxA+Uk1ihz0=
    ) ; key id = 19036
. 93910 IN DNSKEY 256 3 8 (
    AwEAAcaGQEA+OJmOzfzVfoYN249JId7gx+OZMbxy69Hf
    UyuGBbRN0+HuTOpBxxBCkNOL+EJB9qJxt+0FEY6ZUVjE
    g58sRr4ZQ6Iu6b1xTBKgc193zUARk4mmQ/PPGxn7Cn5V
    EGJ/1h6dNaiXuRHwR+7oWh7DnzkIJChcTqlFrXDW3tjt
) ; key id = 34525
```

Олж авсан түлхүүрүүд энэ жишээн дээрхээс өөр байвал сандрах хэрэггүй. Тэдгээр нь энэ зааврыг бичсэнээс хойш өөрчлөгдсөн байж болох юм. Энэ гаралт нь хоёр түлхүүрийг агуулдаг. DNSKEY бичлэгийн төрлийн дараах 257 гэсэн утга бүхий жагсаалтад байгаа эхний түлхүүр нь хэрэгтэй нь юм. Энэ утга нь Аюулгүй Орох Цэг (SEP), түлхүүрийг гарын үсгээр баталгаажуулах түлхүүр гэгддэг (KSK) гэдгийг илэрхийлдэг. 256 гэсэн хоёр дахь түлхүүр нь захирагдагч түлхүүр бөгөөд Бүсийг гарын үсгээр баталгаажуулах түлхүүр (ZSK) гэгддэг. Эдгээр өөр түлхүүрийн төрлүүдийн талаар Бүрэн эрхт DNS серверийн тохиргоо хэсэгт дэлгэрэнгүй байгаа.

Одоо түлхүүрийг шалгаж BIND ашиглаж болох хэлбэрт оруулах ёстой. Түлхүүрийг баталгаажуулахын тулд DSRR-г үүсгэнэ. Эдгээр RR-уудыг агуулсан файлыг дараах тушаалаар үүсгэнэ

```
% dnssec-dsfromkey -f root-dnskey . > root.ds
```

Эдгээр бичлэгүүд нь SHA-1 болон SHA-256-г ашигладаг бөгөөд дараах жишээтэй төстэй харагдах ёстой. Урт нь SHA-256-г ашигладаг.

```
. IN DS 19036 8 1 B256BD09DC8DD59F0E0F0D8541B8328DD986DF6E
. IN DS 19036 8 2 49AAC11D7B6F6446702E54A1607371607A1A41855200FD2CE1CDDE32F24E8FB5
```

SHA-256 RR-г https://data.iana.org/root-anchors/root-anchors.xml дээр байгаа дайжесттай харьцуулж болно. Түлхүүрийг XML файлын өгөгдлөөр өөрчлөгдөөгүйг жинхэнэ утгаар мэдэхийн тулд https://data.iana.org/root-anchors/root-anchors.asc дахь PGP гарын үсгийг ашиглан шалгаж болно.

Дараа нь түлхүүрийг зөв хэлбэрт оруулсан байх ёстой. Энэ нь BIND 9.6.2 болон 9.7 түүнээс хойшхи хувилбаруудын хооронд жаахан ялгаатай байдаг. 9.7 хувилбарт түлхүүрт хийгдэх

өөрчлөлтийг автоматаар хянаж шаардлагатай бол шинэчилдэг дэмжлэг нэмэгдсэн байдаг. Үүнийг доорх жишээн дээр үзүүлсэн шиг `managed-keys` ашиглан хийдэг. Хуучин хувилбар ашиглаж байгаа тохиолдолд түлхүүрийг `trusted-keys` гэдгийг ашиглан нэмдэг бөгөөд шинэчлэлтүүдийг гараар хийх ёстой байдаг. BIND 9.6.2-ийн хувьд формат доорхтой адил хэлбэрийн байна:

```
trusted-keys {
    "." 257 3 8
    "AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQbSEW0O8gcCjF
    FVQUTf6v58fLjwBd0YI0EzrAcQqBGCzh/RStIoO8g0NfnfL2MTJRkxoX
    bfDaUeVPQuYEhg37NZWAJQ9VnMVDxP/VHL496M/QZxkjf5/Efucp2gaD
    X6RS6CXpoY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3LQpz
    W5hOA2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGIcGOYl7OyQdXfZ57relS
    Qageu+ipAdTTJ25AsRTAoub8ONGcLmqrAmRLKBP1dfwhYB4N7knNnulq
    QxA+Uk1ihz0=";
};
```

For 9.7 the format will instead be:

```
managed-keys {
    "." initial-key 257 3 8
    "AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQbSEW0O8gcCjF
    FVQUTf6v58fLjwBd0YI0EzrAcQqBGCzh/RStIoO8g0NfnfL2MTJRkxoX
    bfDaUeVPQuYEhg37NZWAJQ9VnMVDxP/VHL496M/QZxkjf5/Efucp2gaD
    X6RS6CXpoY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3LQpz
    W5hOA2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGIcGOYl7OyQdXfZ57relS
    Qageu+ipAdTTJ25AsRTAoub8ONGcLmqrAmRLKBP1dfwhYB4N7knNnulq
    QxA+Uk1ihz0=";
};
```

Эх түлхүүрийг named.conf файл руу шууд эсвэл түлхүүр бүхий файлыг оруулан нэмж өгч болно. Эдгээр алхмуудын дараа BIND-г хүсэлтүүд дээр DNSSEC шалгалтыг хийдэг болгохын тулд named.conf файлыг засварлан дараах мөрийг `options` хэсэгт нэмж тохиргоог хийнэ:

```
dnssec-enable yes;
dnssec-validation yes;
```

Ажиллаж байгааг шалгахын тулд дөнгөж тохируулсан тайлагчийг ашиглан гарын үсгээр баталгаажсан бүсийг асуусан хүсэлтийг dig ашиглан явуулна. Амжилттай хариулт `AD` тэмдэглэгээтэй байх бөгөөд энэ нь өгөгдлийг таньж зөвшөөрсөн гэсэн үг юм. Доорх хүсэлттэй адил хүсэлтийг ажиллуулбал

```
% dig @resolver +dnssec se ds
```

`.se` бүсийн хувьд DSRR-г буцаах ёстой. `flags:` хэсэг дээр `AD` флаг тохируулагдсан байх ёстой

бөгөөд доорх байдлаар харагдана:

```
...
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1
...
```

Тайлагч одоо DNS хүсэлтүүдийг шалгаж таних чадвартай боллоо.

#### 30.6.8.2. Бүрэн эрхт DNS серверийн тохиргоо

DNSSEC-р баталгаажсан бүсэд үйлчлэх бүрэн эрхт нэрийн сервертэй болохын тулд бага зэргийн зүйлс хийх шаардлагатай. Бүсийг криптограф түлхүүрүүд ашиглан баталгаажуулах ёстой бөгөөд түлхүүрүүдийг үүсгэх ёстой. Энэ зорилгоор зөвхөн нэг түлхүүр ашиглаж болно. Гэхдээ зөвлөдөг арга бол байнга өөрчлөгдөөд байдаггүй, хүчтэй, маш сайн хамгаалагдсан Түлхүүрийг гарын үсгээр баталгаажуулах Түлхүүр (KSK) болон байнга өөрчлөгддөг Бүсийг гарын үсгээр баталгаажуулах Түлхүүртэй (ZSK) байх явдал юм. Үйл ажиллагааны хувьд зөвлөсөн практикуудын талаарх мэдээллийг RFC 4641: DNSSEC үйл ажиллагааны практикууд хаягаас авч болно. Эх бүсийн талаарх практикуудыг Эх бүсийн KSKоператорт зориулсан DNSSEC практик болон Эх бүсийн ZSKоператорт зориулсан DNSSEC практик хаягуудаас олж болно. KSK нь дараалсан бүрэн эрхийг шалгагдах шаардлагатай байгаа өгөгдөлд өгөхөд хэрэглэгддэг бөгөөд бас Secure Entry Point буюу Аюулгүй Орох Цэг (SEP) түлхүүр гэгддэг. Энэ түлхүүрийн зурвасын дайжестийг Delegation Signer буюу Төлөөлөн баталгаажуулагч(DS) бичлэг гэгддэг бөгөөд итгэлцлийн дарааллыг бий болгохын тулд эцэг бүсэд бичигдсэн байх ёстой. Үүнийг хэрхэн хийх нь эцэг бүсийг эзэмшигчээс хамаардаг. ZSK нь бүсийг баталгаажуулахад хэрэглэгддэг бөгөөд тэндээ бичигдсэн байх ёстой байдаг.

Өмнөх жишээн дээр харуулсан `example.com` бүсийн хувьд DNSSEC-г идэвхжүүлэхийн тулд эхний алхам нь KSK болон ZSK түлхүүрийн хослолыг үүсгэх dnssec-keygen-г ашиглах явдал юм. Энэ түлхүүрийн хослол нь өөр өөр криптограф алгоритмуудыг хэрэглэж болно. Түлхүүрүүдийн хувьд RSA/SHA256-г ашиглахыг зөвлөдөг бөгөөд 2048 битийн түлхүүрийн урт хангалттай. `example.com`-н хувьд KSK-г үүсгэхийн тулд дараахийг ажиллуулна

```
% dnssec-keygen -f KSK -a RSASHA256 -b 2048 -n ZONE example.com
```

ZSK-г үүсгэхийн тулд

```
% dnssec-keygen -a RSASHA256 -b 2048 -n ZONE example.com
```

dnssec-keygen хоёр файлыг гаргах бөгөөд нийтийн болон хувийн түлхүүрүүд нь Kexample.com.+005+nnnnn.key (нийтийн) болон Kexample.com.+005+nnnnn.private (хувийн) гэсэн файлуудтай төстэй нэртэйгээр байна. Файлын нэрийн `nnnnn` хэсэг нь таван оронтой түлхүүрийн ID юм. Аль түлхүүрийн ID аль түлхүүрт харгалзаж байгааг хянаж байх хэрэгтэй. Энэ нь ялангуяа бүсэд нэгээс илүү түлхүүр ашиглаж байгаа үед чухал юм. Түлхүүрүүдийн нэрийг бас өөрчилж болно. KSK файл бүрийн хувьд дараахийг ажиллуулна:

```
% mv Kexample.com.+005+nnnnn.key Kexample.com.+005+nnnnn.KSK.key
% mv Kexample.com.+005+nnnnn.private Kexample.com.+005+nnnnn.KSK.private
```

ZSK файлуудын хувьд KSK-г ZSK-р солиорой. Одоо файлуудыг $include ашиглан бүсийн файлд оруулж болно. Иймэрхүү байдалтай харагдана:

```
$include Kexample.com.+005+nnnnn.KSK.key    ; KSK
$include Kexample.com.+005+nnnnn.ZSK.key    ; ZSK
```

Төгсгөлд нь бүсийг баталгаажуулж BIND-д баталгаажуулсан бүсийн файлыг ашиглахыг зааж өгнө. Бүсийг баталгаажуулахын тулд dnssec-signzone-г ашиглана. example.com.db-д байрлах example.com бүсийг баталгаажуулах тушаал иймэрхүү байна

```
% dnssec-signzone -o example.com -k Kexample.com.+005+nnnnn.KSK example.com.db
Kexample.com.+005+nnnnn.ZSK.key
```

-k аргументад өгөгдсөн түлхүүр нь KSK ба нөгөө нэг түлхүүрийн файл нь ZSK бөгөөд баталгаажуулахад хэрэглэгдэх ёстой. Нэгээс илүү KSK болон ZSK өгч болох бөгөөд ингэсэн тохиолдолд бүс бүх өгөгдсөн түлхүүрээр баталгаажна. Энэ нь бүсийн өгөгдлийг нэгээс илүү алгоритмаар баталгаажуулахын тулд хэрэгтэй байж болно. dnssec-signzone-ий гаралт нь бүх RR нь баталгаажсан бүсийн файл байна. Энэ гаралт нь example.com.db.signed мэтийн .signed гэсэн өргөтгөлтэй файлд байх болно. DS бичлэгүүд нь бас тусдаа dsset-example.com файлд бичигддэг. Энэ баталгаажсан бүсийг ашиглахын тулд named.conf файлын бүсийн хэсэгт example.com.db.signed-г ашиглахаар болгож өөрчлөх хэрэгтэй. Анхдагчаар гарын үсгүүд нь 30 хоног хүчинтэй байдаг бөгөөд хүчингүй гарын үсгүүд бүхий бичлэгүүдийг нэр тайлагчдаар хадгалуулахгүй байлгахын тулд бүсийг ядаж ойролцоогоор 15 хоногийн дараа дахин баталгаажуулах хэрэгтэй гэсэн үг юм. Үүнийг хийхийн тулд скрипт бичээд cron-д ажиллуулахаар тохируулж болно. Дэлгэрэнгүйг холбогдох гарын авлагуудаас харна уу.

Бүх криптограф түлхүүрүүдийн адил хувийн түлхүүрүүдийг нууцлан хадгалахыг санаарай. Түлхүүрийг солихдоо шинэ түлхүүрийг бүсэд оруулан хуучнаар эхлээд баталгаажуулах нь зүйтэй бөгөөд дараа нь шинэ түлхүүрийг ашиглан баталгаажуулах хэрэгтэй. Эдгээр алхмуудыг хийсний дараа хуучин түлхүүрийг бүсээс арилгаж болно. Ингэж хийхгүй бол шинэ түлхүүр DNS-н шатлалаар түгээгдэн зарлагдтал DNS-н өгөгдөл нь хүртээмжгүй байх нөхцөлд хүргэж болно. Түлхүүр солих мэдээлэл болон DNSSEC-г ажиллуулахтай холбоотой асуудлуудын талаар дэлгэрэнгүйг RFC 4641: DNSSEC Operational practices хаягаас үзнэ үү.

#### 30.6.8.3. BIND 9.7 болон түүнээс хойшхи хувилбаруудыг ашиглан автоматжуулах

BIND 9.7 хувилбараас эхлээд *Smart Signing* буюу ухаалгаар баталгаажуулах боломж шинээр нэмэгдсэн. Энэ боломж нь түлхүүрийг удирдах болон баталгаажуулах процессын зарим хэсгийг автоматжуулснаар хялбар болгохыг зорьдог. *key repository* санд түлхүүрүүдийг байршуулж auto-dnssec гэсэн шинэ тохиргоог ашиглан шаардлагатай тохиолдолд дахин баталгаажуулагддаг динамик бүсийг үүсгэх боломжтой байдаг. Энэ бүсийг шинэчлэхийн тулд nsupdate-г шинэ -l аргументтай хэрэглэнэ. rndc бас түлхүүр байрлах сан дахь түлхүүрүүдээр бүсүүдийг sign гэсэн тохиргоо ашиглан баталгаажуулах боломжтой болсон.

example.com-н хувьд энэ автоматаар хийх баталгаажуулалт болон бүсийг шинэчлэх боломжийг BIND-д зааж өгөхийн тулд дараахийг named.conf файлд нэмж өгөх хэрэгтэй:

```
zone example.com {
    type master;
    key-directory "/etc/named/keys";
    update-policy local;
    auto-dnssec maintain;
    file "/etc/named/dynamic/example.com.zone";
};
```

Эдгээр өөрчлөлтүүдийг хийсний дараа Бүрэн эрхт DNS серверийн тохиргоо-д тайлбарласны дагуу бүсийн хувьд түлхүүрүүдийг үүсгэж өгнө. Ингэхийн тулд тэр түлхүүрүүдийг түлхүүр байрлах санд хийж бүсийн тохиргооны key-directory гэдэгт уг санг өгөх бөгөөд ингэснээр бүс автоматаар баталгаажуулагдах болно. Ийм замаар тохируулсан бүсэд хийх шинэчлэлтийг nsupdate ашиглан хийх ёстой бөгөөд энэ нь бүсэд шинэ өгөгдөл нэмэн дахин баталгаажуулах ажлыг хийдэг байна. Илүү дэлгэрэнгүйг Гүнзгийрүүлэн унших болон BIND-н баримтаас үзнэ үү.

### 30.6.9. Аюулгүй байдал

Хэдийгээр BIND нь хамгийн өргөн хэрэглэгддэг DNS сервер боловч, аюулгүй байдалтай холбоотой асуудлууд байнга тулгардаг. Гадны халдлагад өртөж болзошгүй аюулгүй байдлын цоорхой заримдаа олддог.

Хэдийгээр FreeBSD named-г автоматаар chroot(8) орчинд оруулдаг боловч; DNS халдлагад ашиглаж болохуйц хэд хэдэн механизмууд байсаар байна.

CERT-с гаргадаг аюулгүй байдлын санамжуудыг уншихыг зөвлөж байна. Мөн FreeBSD аюулгүй байдлын мэдэгдлүүд захидлын жагсаалт-д бүртгүүлж, шинээр гарч байгаа Интернэт болон FreeBSD-н аюулгүй байдлын асуудлуудын талаар мэдээлэлтэй байхыг зөвлөе.

Хэрэв ямар нэгэн асуудал тулгарвал эхийг байнга шинэчилж, named-г шинээр бүтээх нь тусалж болох юм.

### 30.6.10. Гүнзгийрүүлэн унших

BIND/named заавар хуудсууд: rndc(8) named(8) named.conf(8)nsupdate(8) dnssec-signzone(8) dnssec-keygen(8)

- ISC BIND-н Албан ёсны Хуудас
- ISC BIND-н Албан ёсны Хэлэлцүүлэг
- Root DNSSEC
- O'Reilly "DNS ба BIND" 5 дахь Хэвлэлт
- DNSSECЭх бүсэд зориулсан итгэмжит анкор зарлалт (Trust Anchor Publication for the Root

Zone)

- RFC1034 - Домэйн Нэрүүд - Зарчмууд болон Боломжууд
- RFC1035 - Домэйн Нэрүүд - Хэрэгжүүлэлт болон Үзүүлэлтүүд
- RFC4033 - DNS-н аюулгүй байдлын танилцуулга ба шаардлагууд
- RFC4034 - DNS-н аюулгүй байдлын өргөтгөлүүдэд зориулсан Resource Records буюу Нөөцийн Бичлэгүүд
- RFC4035 - DNS-н аюулгүй байдлын өргөтгөлүүдэд зориулсан протоколын өөрчлөлтүүд
- RFC4641 - DNSSEC ажиллуулах практикууд
- RFC 5011 - DNS-н аюулгүй байдлын автомат шинэчлэлтүүд (DNSSEC Trust Anchors

# 30.7. Apache HTTP Сервер

## 30.7.1. Удиртгал

Дэлхийн хамгийн их ачаалалтай ажилладаг зарим вэб сайтууд FreeBSD дээр ажилладаг. Интернэтэд ажиллаж байгаа вэб серверүүдийн олонхи нь Apache HTTP Серверийг ашиглаж байна. Apache програм хангамжийн багц таны FreeBSD суулгах дискэнд орсон байгаа. Хэрэв та FreeBSD-г анх суулгахдаа Apache-г хамт суулгаагүй бол www/apache22 портоос суулгаж болно.

Apache нэгэнт амжилттай суусан бол түүнийг тохируулах шаардлагатай.

Apache HTTP Server-н 2.2.X хувилбар нь FreeBSD-д хамгийн өргөн хэрэглэгддэг тул бид энэ хэсэгт энэ хувилбарыг үзэх болно. Apache 2.X-н талаар энэ баримтын хүрээнээс хальсан дэлгэрэнгүй мэдээллийг http://httpd.apache.org/ хаягаар орж үзнэ үү.

## 30.7.2. Тохиргоо

FreeBSD дээрх Apache HTTP Серверийн гол тохиргооны файл бол /usr/local/etc/apache22/httpd.conf юм. Энэ файлд, UNIX®-н текст тохиргооны файлын нэгэн адил тайлбар мөрүүдийн өмнө # тэмдэгтийг хэрэглэдэг. Бүх боломжит тохируулгуудын талаар дэлгэрүүлж тайлбарлах нь энэ номын хүрээнээс халих тул, хамгийн их өөрчлөлт хийгддэг директивүүдийг энд авч үзье.

`ServerRoot "/usr/local"`

Энэ директив Apache суулгацын анхдагч директор шатлалын эхийг зааж өгнө. Хоёртын файлууд серверийн эх директорын bin ба sbin дэд директоруудад, тохиргооны файлууд etc/apache дэд директорт байрлана.

`ServerAdmin you@your.address`

Сервертэй холбоотой асуудлуудын талаар илгээх цахим захидлын хаягийг заана. Энэ хаяг алдааны хуудсууд гэх зэрэг сервер талаас автоматаар үүсгэгддэг зарим хуудсууд дээр бичигдэх болно.

`ServerName www.example.com`

`ServerName` нь хост дээр тохируулагдсан хост нэрээс өөр нэрийг сервертээ өгөх боломжийг танд олгоно (өөрөөр хэлбэл, хостын жинхэнэ хост нэрийн оронд `www`-г хэрэглэх). Энэ нэрээр таны сервер харилцагч нартай харилцах болно.

`DocumentRoot "/usr/local/www/apache22/data"`

`DocumentRoot`: Энэ директорт байгаа вэб баримтуудыг харилцагч нарт үзүүлэх болно. Анхдагч байдлаар, бүх хүсэлтүүд энэ директорт өгөгдөнө. Гэвч симбол холбоосууд болон хуурамч дүрүүдийг ашиглан өөр газар руу зааж өгч болно.

Apache-н тохиргооны файлд ямар нэг өөрчлөлт хийхээсээ өмнө нөөц хуулбарыг авч үлдэхээ мартуузай. Тохиргоо хийж дууссан бол одоо Apache-г ажиллуулах хэрэгтэй.

### 30.7.3. Apache-г ажиллуулах нь

www/apache22 порт нь Apache-г эхлүүлэх, зогсоох болон дахин ачаалахад хэрэгтэй rc(8) скриптийг суулгадаг бөгөөд энэ нь /usr/local/etc/rc.d/ санд байрладаг.

Систем ачаалах үед Apache-г эхлүүлэхийн тулд дараах мөрүүдийг /etc/rc.conf файлд нэмж бичнэ:

```
apache22_enable="YES"
```

Хэрэв Apache-г анхдагч биш сонголтуудтай ажиллуулах бол дараах мөрийг /etc/rc.conf файлд нэмж тохируулж болно:

```
apache22_flags=""
```

Apache-н тохиргоог `httpd` демонг анх эхлүүлэхээсээ өмнө юм уу эсвэл `httpd` ажиллаж байгаа үед дараалсан тохиргооны өөрчлөлтүүдиийг хийсний дараа алдаа байгаа эсэхийг тест хийж болно. Үүнийг rc(8) скриптээр шууд хийх юм уу эсвэл service(8) хэрэгслийг ашиглан дараах тушаалуудын аль нэгийг ажиллуулж хийнэ:

```
# service apache22 configtest
```

`configtest` нь rc(8)-ий хувьд стандарт биш гэдгийг санаарай, бүх rc(8) эхлүүлэх скриптүүдийн хувьд ажиллахгүй байж болно.

Хэрэв Apache тохиргооны алдаа өгөөгүй бол Apache `httpd`-г адил service(8) механизмаар эхлүүлж болно:

```
# service apache22 start
```

`httpd` үйлчилгээг вэб хөтөч дээр `http://localhost` гэж тест хийж болно. Хэрэв энэ нь локал

машин биш бол `httpd` ажиллаж байгаа машины бүрэн танигдсан домен нэрээр сольж тестлээрэй. Харуулагдах анхдаг вэб хуудас нь /usr/local/www/apache22/data/index.html байна.

### 30.7.4. Давхар байршуулалт

Apache нь хоёр төрлийн давхар байршуулах үйлчилгээг дэмждэг. Эхнийх нь нэр дээр үндэслэсэн давхар байршуулалт юм. Нэр дээр үндэслэсэн давхар байршуулалт дээр хост нэрийг ялгаж мэдэхдээ харилцагчийн HTTP/1.1 толгойн хэсгийг ашигладаг. Иим байдлаар олон өөр домэйнууд нэг IP хаягийг хуваан хэрэглэх боломжтой болдог.

Apache дээр, нэр дээр үндэслэсэн давхар байршуулалтыг хэрэглэхийн тулд доор дурдсантай төстэй бүртгэлийг httpd.conf файл дотор нэмж бичих хэрэгтэй:

```
NameVirtualHost *
```

Таны вэб серверийн нэр `www.domain.tld` бөгөөд `www.someotherdomain.tld` нэртэй домэйныг давхар байршуулах хүсэлтэй бол, та дараах бүртгэлийг httpd.conf файлд нэмэх хэрэгтэй болно:

```
<VirtualHost *>
ServerName www.domain.tld
DocumentRoot /www/domain.tld
</VirtualHost>

<VirtualHost *>
ServerName www.someotherdomain.tld
DocumentRoot /www/someotherdomain.tld
</VirtualHost>
```

Дээрх хаягуудын оронд хэрэгтэй хаягуудыг, замуудын оронд баримтууд байгаа зохих замуудыг сольж бичнэ үү.

Давхар хостуудыг зохион байгуулах талаар дэлгэрэнгүй мэдээллийг Apache-н албан ёсны баримтжуулалт: http://httpd.apache.org/docs/vhosts/-с олж үзнэ үү.

### 30.7.5. Apache Модулиуд

Үндсэн серверийн үүрэг функцыг сайжруулахын тулд бүтээгдсэн Apache-н олон модулиуд байдаг. FreeBSD Портуудын Цуглуулга нь Apache-г түүний өргөн хэрэглэгддэг зарим модулиудын хамт хялбар суулгах боломжийг олгодог.

#### 30.7.5.1. mod_ssl

mod_ssl модуль нь Secure Sockets Layer (SSL v2/v3) ба Transport Layer Security (TLS v1) протоколоор дамжуулан өндөр нууцлалыг хангахын тулд OpenSSL санг ашигладаг. Энэ модуль нь батламж олгодог итгэмжлэгдсэн байгууллагаас батламж авахын тулд шаардлагатай бүх зүйлсээр хангадаг тул та үүнийг ашиглан FreeBSD дээр аюулгүй вэб

сервер ажиллуулж чадна.

mod_ssl модуль нь анхдагчаар бүтээгдсэн байдаг боловч бүхээх үедээ `-DWITH_SSL` сонголт ашиглан идэвхжүүлж болно.

#### 30.7.5.2. Хэлний холболтууд

Ихэнх гол скрипт хэлнүүдэд зориулсан Apache-ийн модулиуд байдаг. Эдгээр модулиуд нь Apache-ийн модулиудыг бүхэлд нь скрипт хэл дээр бичих боломжийг ихэвчлэн бүрдүүлдэг. Эдгээр нь бас гадаад тайлбарлагчийг эхлүүлэх нэмэлт зардал болон димамик вэб сайтуудын хувьд байдаг эхлүүлэх хугацааны алдагдлыг тойрон гарах, серверт багтааж хийгдсэн байнгын тайлбарлагч болон дараагийн хэсэгт тайлбарлагдсан шигээр ихэвчлэн ашиглагддаг.

### 30.7.6. Динамик вэб сайтууд

Сүүлийн 10 жилд, өөрийн ашиг орлогыг нэмэгдүүлэх, хүмүүст хүрэх зорилгоор илүү олон компаниуд бизнесээ Интернэтээр явуулах болжээ. Энэ нь динамик агуулгатай вэб хуудсууд төрөн гарах хэрэгцээ шаардлагыг улам нэмэгдүүлсэн. Microsoft® гэх мэт зарим компаниуд ч өөрийн бүтээгдэхүүнүүдэд тэдгээрээс оруулах болсон хэдий ч, нээлттэй эхийн нэгдэл энэ асуудалд хариу өгсөн юм. Динамик вэб агуулгыг бий болгох орчин үеийн боломжууд бол Django, Ruby on Rails, mod_perl2 болон mod_php юм.

#### 30.7.6.1. Django

Django нь өндөр ажиллагаатай, гоёмсог вэб програмыг хурдан бичих боломжийг хөгжүүлэгчдэд олгохоор хийгдсэн, BSD лицензтэй тогтолцоо юм. Энэ нь өгөгдлийн төрлүүд Python обьект хэлбэрээр хөгжүүлэгддэг байхаар болгосон обьектийн харилцааг оноогчтой бөгөөд тэдгээр обьектуудад зориулсан хөгжүүлэгчдэд SQL бичих шаардлагагүй болгож өгдөг, баялаг динамик өгөгдлийн сангийн хандалтын API-тай юм. Энэ нь бас програмын логикийг HTML үзүүлбэрээс тусгаарлах боломжийг бүрдүүлэх нэмэлт загварын системтэй байдаг.

Django нь mod_python, Apache, болон таны сонгосон SQL өгөгдлийн сангийн хөдөлгүүрээс хамаардаг. FreeBSD порт нь эдгээр бүх хамаарлуудыг тохирсон сонголтуудтай нь танд суулгаж өгөх болно.

*Жишээ 38. Django-г Apache2, mod_python3, болон PostgreSQL суулгах нь*

```
# cd /usr/ports/www/py-django; make all install clean -DWITH_MOD_PYTHON3
-DWITH_POSTGRESQL
```

Django болон бусад хамаарлууд суулгагдсаны дараа та Django төслийн санг үүсгэх хэрэгтэй бөгөөд өөрийн сайт дээрх тухайн URL дээр өөрийн програмыг дуудахын тулд суулгагдсан Python тайлбарлагчийг ашиглахаар болгож Apache-г тохируулах хэрэгтэй.

*Жишээ 39. Django/mod_python-д зориулсан Apache-ийн тохиргоо*

Та өөрийн вэб програм руу тодорхой URL-уудад зориулсан хүсэлтүүдийг дамжуулахаар

Apache-г тохируулахын тулд apache-ийн httpd.conf файлд мөр нэмэх шаардлагатай:

```
<Location "/">
    SetHandler python-program
    PythonPath "['/dir/to/your/django/packages/'] + sys.path"
    PythonHandler django.core.handlers.modpython
    SetEnv DJANGO_SETTINGS_MODULE mysite.settings
    PythonAutoReload On
    PythonDebug On
</Location>
```

#### 30.7.6.2. Ruby on Rails

Ruby on Rails нь бүрэн гүйцэд хөгжүүлэлтийн стекийн боломжийг олгодог бөгөөд вэб хөгжүүлэгчдийг хүчирхэг програмыг хурдан шуурхай, илүү үр бүтээлтэй бичдэг байхаар оновчлогдсон, нээлттэй эхийн вэб тогтолцоо юм. Үүнийг портын системээс хялбараар суулгаж болно.

```
# cd /usr/ports/www/rubygem-rails; make all install clean
```

#### 30.7.6.3. mod_perl2

Apache/Perl нэгтгэх төсөл Perl програмчлалын хэл ба Apache HTTP Серверийн бүх хүч чадлыг нэгтгэсэн юм. mod_perl2 модулийн тусламжтай Apache модулиудыг тэр чигээр нь Perl дээр бичих боломжтой. Дээр нь, серверт суулгасан шургуу хөрвүүлэгч, гадны хөрвүүлэгч ашиглах илүү ажил болон Perl эхлүүлэх хугацааны алдагдлаас зайлсхийж чадсан юм.

mod_perl2 нь www/mod_perl2 портод байдаг.

#### 30.7.6.4. mod_php

PHP буюу "PHP:Hypertext Preprocessor" бол вэб хөгжүүлэлтэд тусгайлан тохируулсан, энгийн хэрэглээний скрипт хэл юм. HTML дотор суулгах боломжтой түүний синтакс C, Java™, ба Perl-с гаралтай. Энэ нь вэб хөгжүүлэгчдэд динамикаар үүсгэгдэх вэб хуудсыг хурдан бичих боломжтой болгох үүднээс тэгсэн хэрэг.

Apache вэб серверийг PHP5-г дэмждэг болгохын тулд, lang/php5 портыг суулгаж эхлэх хэрэгтэй.

Хэрэв lang/php5 портыг анх удаа суулгаж байгаа бол, боломжит `ТОХИРУУЛГУУД` автоматаар дэлгэцэн дээр гарч ирнэ. Хэрэв цэс гарч ирэхгүй бол, өөрөөр хэлбэл lang/php5 портыг өмнө нь хэзээ нэгэн цагт суулгаж байсан бол, тохируулгуудын харилцах цонхыг гаргаж ирэхийн тулд дараах тушаалыг:

```
# make config
```

порт директор дотор өгөх хэрэгтэй.

Тохируулгуудын харилцах цонхонд, mod_php5-г Apache-н ачаалах боломжтой модуль байдлаар бүтээхийн тулд `APACHE` тохируулгыг идэвхжүүлнэ.

Олон сайтууд PHP4-г янз бүрийн шалтгааны улмаас (өөрөөр хэлбэл, нийцтэй байдал эсвэл аль хэдийн үйлчилгээнд гаргачихсан вэб програмууд) ашигласаар байна. Хэрэв mod_php4-г mod_php5-н оронд ашиглах шаардлагатай бол, lang/php4 портыг ашиглаарай. lang/php4 порт нь lang/php5 портод байдаг тохиргооны болон бүтээх үеийн олон тохируулгуудыг дэмждэг.

Энэ хэсэг код динамик PHP програмыг дэмждэг болгоход шаардлагатай модулиудыг суулгаж тохируулах болно. Доорх мөрүүд /usr/local/etc/apache22/httpd.conf файл дотор нэмэгдсэн эсэхийг шалгаарай:

```
LoadModule php5_module        libexec/apache/libphp5.so
```

```
AddModule mod_php5.c
    <IfModule mod_php5.c>
        DirectoryIndex index.php index.html
    </IfModule>
    <IfModule mod_php5.c>
        AddType application/x-httpd-php .php
        AddType application/x-httpd-php-source .phps
    </IfModule>
```

Үүний дараа, PHP модулийг ачаалахын тулд, дараах тушаалыг өгч серверийг дахин ачаалах хэрэгтэй:

```
# apachectl graceful
```

Дараа, PHP-н хувилбарыг дээшлүүлэх үедээ, `make config` тушаалыг өгөх шаардлагагүй; идэвхжүүлсэн `ТОХИРУУЛГУУД` FreeBSD Портуудын тогтолцоонд автоматаар хадгалагдсан байгаа.

FreeBSD-н PHP дэмжлэг нь дээд зэргээр модульчлагдсан тул үндсэн суулгац нь маш хязгаарлагдмал байдаг. lang/php5-extensions портыг ашиглан дэмжлэг нэмэх нь үнэхээр амархан асуудал. PHP өргөтгөлийг суулгах явцад, энэ порт танд цэсээс тогтсон интерфэйсийг санал болгоно. Өөрөөр, өргөтгөлүүдийг нэг нэгээр нь харгалзах портуудаас суулгаж болно.

Жишээлбэл, PHP5-д MySQL өгөгдлийн сангийн серверийн дэмжлэгийг нэмэхийн тулд, databases/php5-mysql портыг суулгахад хангалттай.

Ямар нэг өргөтгөл суулгасны дараа, тохиргооны өөрчлөлтийг хүчин төгөлдөр болгохын тулд Apache серверийг дахин ачаалах шаардлагатайг анхаарна уу:

```
# apachectl graceful
```

# 30.8. Файл Дамжуулах Протокол (FTP)

## 30.8.1. Удиртгал

File Transfer Protocol буюу Файл Дамжуулах Протокол (FTP) нь хэрэглэгчдэд FTP серверээс файлыг авах болон тавих хялбар замыг бий болгодог. FreeBSD үндсэн систем дотроо FTP сервер програм ftpd-г агуулж байдаг. Энэ нь FreeBSD дээр FTP серверийг босгох, удирдах ажлыг төвөггүй болгодог.

## 30.8.2. Тохиргоо

Тохиргоо хийхийн өмнөх хамгийн чухал алхам бол ямар дансууд FTP серверт хандах эрхтэй байх вэ гэдгийг шийдэх байдаг. Ердийн FreeBSD систем нь янз бүрийн дэмонуудад хэрэглэгддэг олон тооны системийн дансуудтай байдаг ба гадны хэрэглэгчид эдгээр дансыг ашиглан нэвтрэх ёсгүй. /etc/ftpusers файл дотор FTP хандалт зөвшөөрөгдөөгүй хэрэглэгчдийн жагсаалтыг хадгална. Анхдагч байдлаар, дээр дурдсан системийн дансууд энэ файлд байна. FTP хандалтыг зөвшөөрөх ёсгүй өөр хэрэглэгчдийг ч мөн энэ файлд нэмж болно.

Зарим хэрэглэгчдийн FTP хэрэглэхийг нь бүр болиулчихалгүйгээр, зөвхөн зарим нэг эрхийг нь хязгаарлаж бас болно. Үүнийг /etc/ftpchroot файлын тусламжтай гүйцэтгэж болно. Энэ файл дотор FTP хандалтыг нь хязгаарлах хэрэглэгчид болон бүлгүүдийн жагсаалт байна. ftpchroot(5) заавар хуудсанд бүх мэдээлэл байгаа тул энд дурдсангүй.

Хэрэв сервертээ нийтийн FTP хандалтыг зөвшөөрөх хүсэлтэй байгаа бол, FreeBSD систем дээрээ `ftp` нэртэй хэрэглэгч нэмэх хэрэгтэй. Ингэснээр хэрэглэгчид таны FTP сервер рүү `ftp` эсвэл `anonymous` гэсэн нэрээр ямар ч нэвтрэх үг шаардагдахгүйгээр (тогтсон заншил ёсоор хэрэглэгч цахим шуудангийн хаягаа нэвтрэх үгийн оронд хэрэглэх шаардлагатай) нэвтрэн орох болно. Нийтийн хэрэглэгч системд орж ирэхэд FTP сервер түүний эрхийг зөвхөн `ftp` хэрэглэгчийн гэрийн сан дотор хязгаарлахын тулд chroot(2)-г дуудна.

FTP харилцагчдад зориулсан мэндчилгээний үгнүүдийг агуулсан хоёр текст файл байдаг. /etc/ftpwelcome файл дотор байгааг нэвтрэлт хүлээх мөр гарахаас өмнө хэрэглэгчдэд дэлгэцэн дээр хэвлэнэ. Амжилттай нэвтэрч орсны дараа /etc/ftpmotd файл дотор байгааг дэлгэцэн дээр хэвлэнэ. Энэ файлын зам нь нэвтэрч орсон орчинтой харьцангуйгаар авсан зам гэдгийг анхаарна уу, тиймээс нийтийн хэрэглэгчдийн хувьд ~ftp/etc/ftpmotd файлыг хэвлэх болно.

FTP серверийн тохиргоог зохих ёсоор хийсний дараа, /etc/inetd.conf файл дотор идэвхжүүлэх хэрэгтэй. Үүний тулд, ftpd гэсэн мөрний өмнөх "#" тэмдэгтийг арилгахад хангалттай:

```
ftp stream  tcp nowait  root    /usr/libexec/ftpd   ftpd -l
```

inetd-н тохиргооны файлыг дахин ачаалах нь хэсэгт тайлбарласан ёсоор энэ тохиргооны файлд өөрчлөлт оруулсны дараа inetd-г дахин ачаалах шаардлагатай. Өөрийн систем дээр inetd-г идэвхжүүлэх талаар дэлгэрэнгүйг Тохиргоо-с үзнэ үү.

Мөн ftpd-ийг дангаар нь ажиллуулахаар тохируулж болно. Энэ тохиолдолд /etc/rc.conf файлд тохирох хувьсагчийг тохируулахад хангалттай байдаг:

```
ftpd_enable="YES"
```

Дээрх хувьсагчийг тохируулсны дараа сервер дараачийн ачаалалт хийхэд ажиллах боломжтой болох бөгөөд эсвэл дараах тушаалыг `root` эрхээр ажиллуулан эхлүүлж болно:

```
# service ftpd start
```

Одоо та дараах тушаалыг өгөн FTP сервер рүү нэвтрэн орж болно:

```
% ftp localhost
```

### 30.8.3. Арчилгаа

ftpd дэмон бүртгэл хөтлөхдөө syslog(3)-г ашигладаг. Анхдагч байдлаар, системийн бүртгэлийн дэмон FTP-тэй холбоотой зурвасуудыг /var/log/xferlog файлд бичнэ. FTP бүртгэлийн файлын байршлыг өөрчлөхийн тулд /etc/syslog.conf файл дотор, дараах мөрийг засах хэрэгтэй:

```
ftp.info      /var/log/xferlog
```

Нийтийн FTP сервер ажиллуулахад тохиолдох болзошгүй асуудлуудын талаар мэдлэгтэй байгаарай. Ялангуяа, нийтийн хэрэглэгчдэд файл байршуулахыг зөвшөөрөх тухайд сайн бодох хэрэгтэй. Таны FTP сайт лицензгүй програм хангамжуудыг наймаалцдаг талбар болох, эсвэл түүнээс ч муу зүйл тохиолдохыг үгүйсгэхгүй. Хэрэв нийтийн FTP байршуулалтыг зөвшөөрөх шаардлагатай бол, файлуудыг нягталж үзэхээс нааш бусад нийтийн хэрэглэгчид тэдгээр файлыг унших эрхгүй байхаар тохируулж өгөх хэрэгтэй.

## 30.9. Microsoft® Windows® харилцагчдад зориулсан Файл болон Хэвлэх Үйлчилгээ (Samba)

### 30.9.1. Ерөнхий Агуулга

Samba бол Microsoft® Windows® харилцагчдад файл болон хэвлэх үйлчилгээг үзүүлдэг, өргөн хэрэглэгддэг нээлттэй эхийн програм хангамжийн багц юм. Ийм төрлийн харилцагчид FreeBSD файлын орчинд холбогдож, файлуудыг өөрийн дискэн дээр байгаа юм шиг, эсвэл FreeBSD хэвлэгчийг өөрийн дотоод хэвлэгч шиг хэрэглэх боломжтой болдог.

Samba програм хангамжийн багцууд таны FreeBSD суулгах дискэнд орсон байгаа. Хэрэв та анх FreeBSD суулгахдаа Samba-г хамт суулгаагүй бол, net/samba34 порт эсвэл багцаас суулгаж болно.

### 30.9.2. Тохиргоо

Samba-н анхдагч тохиргооны файл /usr/local/shared/examples/samba34/smb.conf.default гэж суугдсан байдаг. Энэ файлыг /usr/local/etc/smb.conf нэртэй хуулаад, Samba-г ашиглаж эхлэхээсээ өмнө өөртөө тааруулан засварлах ёстой.

smb.conf файл нь Windows® харилцагчтай хуваалцах хүсэлтэй "файл системийн хэсэг" ба хэвлэгчийн тодорхойлолт гэх зэрэг Samba-н ажиллах үеийн тохиргооны мэдээллийг агуулж байдаг. Samba багц дотор smb.conf файл дээр ажиллах хялбар арга замыг хангасан swat нэртэй вэб дээр суурилсан хэрэгсэл хамт ирдэг.

#### 30.9.2.1. Samba-г Вэбээр Удирдах Хэрэгсэл (SWAT)

Samba Web Administration Tool буюу Samba-г Вэбээр Удирдах Хэрэгсэл (SWAT) нь inetd-н дэмон хэлбэрээр ажиллана. Тиймээс inetd"Супер-Сервер" дээр харуулсан шиг inetd-г идэвхжүүлж Samba-г swat ашиглан тохируулахын өмнө /etc/inetd.conf доторх дараах мөрийг ил гаргах шаардлагатай:

```
swat   stream  tcp     nowait/400      root    /usr/local/sbin/swat    swat
```

inetd-н тохиргооны файлыг дахин ачаалах нь хэсэгт тайлбарласан ёсоор, энэ тохиргооны файлд өөрчлөлт оруулсны дараа inetd-ийн тохиргоог дахин ачаалах шаардлагатай.

swat-г inetd.conf дотор идэвхжүүлсний дараа, вэб хөтөч ашиглан http://localhost:901 хаяганд холбогдоно. Та эхлээд системийн `root` дансаар нэвтэрч орох ёстой.

Samba-н тохиргооны үндсэн хуудсанд амжилттай нэвтэрч орсон бол, системийн баримтуудаар аялах, эсвэл **Globals** цэсэн дээр дарж тохиргоог хийх боломжтой болно. **Globals** хэсэг /usr/local/etc/smb.conf файлын `[global]` хэсэгт байгаа хувьсагчдад харгалзана.

#### 30.9.2.2. Глобал тохиргоо

swat-г хэрэглэж байгаа эсвэл /usr/local/etc/smb.conf-г гараараа засаж байгаа аль нь ч бай, Samba-г тохируулах явцад тааралдах хамгийн эхний директивууд бол:

`workgroup`

Энэ нь сервер рүү хандах компьютеруудын NT Домэйн-Нэр эсвэл Ажлын бүлгийн-Нэр.

`netbios name`

Энэ директив Samba серверийн NetBIOS нэрийг заана. Анхдагч байдлаар, хостын DNS нэрийн эхний хэсэгтэй адил байна.

`серверийн мөр`

Энэ директив `net view` тушаалын хариуд гарч ирэх эсвэл зарим сүлжээний хэрэгслүүд дээр энэ серверийг төлөөлж гарах мөрийг заана.

#### 30.9.2.3. Ашуулгүй байдлын Тохиргоо

/usr/local/etc/smb.conf доторх хамгийн чухал хоёр тохиргоо бол ашуулгүй байдлын загвар,

болон харилцагчдын нэвтрэх үгийн арын шугамны хэлбэр юм. Дараах директивүүд эдгээр тохируулгуудыг хянана:

`security`

Энд хамгийн элбэг хэрэглэгддэг хоёр сонголт бол `security = share` ба `security = user` юм. Хэрэв танай харилцагч нар FreeBSD машин дээр хэрэглэдэг хэрэглэгчийн нэртэй ижил нэрийг ашигладаг бол, user түвшний аюулгүй байдлыг сонгохыг хүсэж байж магадгүй. Энэ бол аюулгүй байдлын анхдагч бодлого бөгөөд эх үүсвэрт хандахаас өмнө харилцагчийг системд нэвтэрч орохыг шаардана.

share түвшний аюулгүй байдалд, харилцагчид эх үүсвэрт хандахаас өмнө хүчин төгөлдөр хэрэглэгчийн нэр болон нэвтрэх үгээр сервер рүү нэвтрэн орох шаардлагагүй байдаг. Энэ бол Samba-н хуучин хувилбаруудын хувьд аюулгүй байдлын анхдагч загвар байсан.

`passdb backend`

+ Samba-д хэд хэдэн төрлийн арын шугамны магадлах загварууд байдаг. Харилцагчдыг LDAP, NIS, SQL өгөгдлийн сан, эсвэл хувиргасан нэвтрэх үгийн файлаар магадлаж болно. Анхдагч магадлах арга бол `smbpasswd` бөгөөд бид зөвхөн энэ талаар авч үзэх болно.

Анхдагч `smbpasswd` арын шугамыг хэрэглэж байгаа гэж үзвэл, Samba харилцагчдыг магадлахын тулд /usr/local/etc/samba/smbpasswd файлыг эхлээд үүсгэх ёстой. Хэрэв UNIX® хэрэглэгчийн эрхээр Windows® харилцагчаас ханддаг байх шаардлагатай бол, дараах тушаалыг хэрэглэнэ:

```
# smbpasswd -a username
```

Энэ үед санал болгодог арын мэдээллийн сан нь `tdbsam` бөгөөд хэрэглэгчийн бүртгэлийг нэмэхийн тулд дараах тушаалыг ашиглах ёстой:

```
# pdbedit -a -u username
```

Тохируулгуудын талаар нэмэлт мэдээллийг Албан ёсны Samba HOWTO-с олж авна уу. Энд цухас дурдсан үндсэн мэдлэгтэйгээр Samba-г ажиллуулж эхлэх чадвартай байх ёстой.

### 30.9.3. Samba-г Эхлүүлэх нь

net/samba34 портод Samba-г удирдахад зориулсан шинэ эхлэл скрипт орсон байгаа. Энэ скриптийг идэвхжүүлэхийн тулд, өөрөөр хэлбэл энэ скриптийг ашиглан Samba-г эхлүүлэх, зогсоох болон дахин эхлүүлдэг болохын тулд, /etc/rc.conf файл дотор дараах мөрийг нэмж бичих хэрэгтэй:

```
samba_enable="YES"
```

Эсвэл илүү нарийнаар доор дурдсан шиг тохируулж болно:

```
nmbd_enable="YES"
```

```
smbd_enable="YES"
```

Ингэснээр мөн Samba-г систем ачаалах үед автоматаар эхлүүлдэг болгоно.

Үүний дараа хүссэн үедээ Samba-г эхлүүлэхийн тулд дараах тушаалыг өгөхөд хангалттай:

```
# service samba start
Starting SAMBA: removing stale tdbs :
Starting nmbd.
Starting smbd.
```

rc скриптийг ашиглах талаар дэлгэрэнгүй мэдээллийг FreeBSD дээр rc(8) ашиглах нь хэсгээс авна уу.

Samba нь үнэн хэрэгтээ гурван тусдаа дэмоноос тогтоно. nmbd ба smbd дэмонууд samba скриптээр эхлүүлдэг болохыг та анзаарах болно. Хэрэв smb.conf дотор winbind нэр тайлах үйлчилгээг идэвхжүүлсэн бол winbindd дэмон бас ажиллаж эхэлсэн болохыг харж болно.

Samba-г хүссэн үедээ зогсоохын тулд дараах тушаалыг өгөхөд хангалттай:

```
# service samba stop
```

Samba бол Microsoft® Windows® сүлжээтэй өргөн хүрээнд нэгдмэл ажиллах боломжийг олгодог нарийн төвөгтэй програмын цогц юм. Энд тайлбарласан үндсэн суулгацаас хальсан функцуудын талаар дэлгэрэнгүй мэдээллийг http://www.samba.org хаягаар орж авна уу.

# 30.10. ntpd-р Цаг Тааруулах нь

## 30.10.1. Ерөнхий Агуулга

Цаг хугацаа өнгөрөхөд компьютерийн цаг зөрөх хандлагатай байдаг. Network Time Protocol буюу Сүлжээний Цагийн Протоколыг(NTP) цагийг зөв байлгах, зөв ажиллуулахад хэрэглэдэг.

Олон тооны Интернэт үйлчилгээнүүд компьютерийн цагаас хамаарч, эсвэл хүртэж ажилладаг. Жишээлбэл, вэб сервер тодорхой цагаас хойш өөрчлөлт орсон файлуудыг илгээх хүсэлт хүлээн авсан байж болох юм. Дотоод сүлжээний орчинд, нэг файл серверээр үйлчлүүлж байгаа компьютеруудын хувьд файлын цагийн тамга дүйж байхын тулд тэдгээрийн цагууд хоорондоо тохирч байх ёстой. cron(8) зэрэг үйлчилгээнүүд тодорхой цагт тушаалыг гүйцэтгэхийн тулд системийн цагт бүрэн найдаж ажилладаг.

FreeBSD ntpd(8) NTP серверийн хамт ирдэг. ntpd(8) NTP нь таны машины цагийг тааруулахын тулд бусад NTP серверүүдээс асуух эсвэл бусдад цагийн мэдээллийг түгээх

үйлчилгээг үзүүлдэг.

### 30.10.2. Зохимжтой NTP Серверийг Сонгох нь

Цагаа тааруулахын тулд, та нэг болон түүнээс дээш тооны NTP серверийг хэрэглэх хэрэгтэй болно. Танай сүлжээний администратор эсвэл ISP үүнд зориулсан NTP сервертэй байж болох юм-тийм эсэхийг тэдний зааврааc шалгана уу. нийтэд зориулсан NTP серверүүдийн онлайн жагсаалтыг ашиглан өөртөө ойрхон байгаа NTP серверийг олно уу. Сонгож авсан серверийнхээ ашиглах журмыг судлаарай. Мөн хэрэв шаардлагатай бол зөвшөөрөл аваарай.

Таны сонгосон сервер холбогдох боломжгүй, эсвэл цаг нь бүрэн итгэж болохооргүй үе гарах тул, хоорондоо хамааралгүй хэд хэдэн NTP серверүүдийг сонгох нь хамгийн зөв сонголт болдог. ntpd(8) бусад серверээс хүлээн авсан хариултуудыг маш ухаалгаар хэрэглэдэг-итгэж болох серверүүдийг илүү авч үздэг.

### 30.10.3. Өөрийн Машиныг Тохируулах нь

#### 30.10.3.1. Үндсэн Тохиргоо

Хэрэв та машин асахад цагаа тааруулах хүсэлтэй байгаа бол, ntpdate(8)-г ашиглаж болно. Энэ нь олон дахин тааруулах шаардлагагүй, ойр ойрхон асааж унтраадаг ширээний компьютерийн хувьд зохимжтой байж болох юм. Гэхдээ ихэнх машины хувьд ntpd(8)-г ажиллуулах нь зүйтэй.

Систем ачаалах үед ntpdate(8)-г ашиглах нь ntpd(8) ажиллаж байгаа машинуудын хувьд зөв санаа юм. Учир нь ntpd(8) програм нь цагийг алгуур өөрчилдөг байхад, ntpdate(8) машины одоогийн цаг болон зөв цагын хооронд хир их ялгаа байгааг үл хайхран цагийг тааруулдаг.

ntpdate(8)-г систем ачаалах үед идэвхжүүлэхийн тулд, `ntpdate_enable="YES"` гэсэн мөрийг /etc/rc.conf файлд нэмэх хэрэгтэй. Мөн цаг авах гэж байгаа бүх серверүүд болон ntpdate(8)-д өгөх тугуудыг `ntpdate_flags`-д зааж өгөх хэрэгтэй.

#### 30.10.3.2. Ерөнхий Тохиргоо

NTP-г /etc/ntp.conf файлын тусламжтай, ntp.conf(5)-д заасан хэлбэрээр тохируулна. Доор хялбар жишээг үзүүлэв:

```
server ntplocal.example.com prefer
server timeserver.example.org
server ntp2a.example.net

driftfile /var/db/ntp.drift
```

`server` тохируулгаар ямар серверүүдийг ашиглахыг заана. Нэг мөрөнд нэг серверийг бичнэ. Хэрэв аль нэг серверийг `prefer` гэсэн аргументаар онцолсон бол, `ntplocal.example.com` шиг, тэр серверийг бусдаас илүүд үзнэ. Илүүд үзсэн серверээс ирсэн хариу бусад серверүүдийн хариунаас мэдэгдэхүйцээр зөрж байгаа үед хариуг тоохгүй өнгөрөөнө. Түүнээс бусад тохиолдолд бусад серверийн хариуг үл харгалзан тэр серверийн хариуг хэрэглэх болно.

`prefer` аргументийг ер нь өндөр нарийвчлалтай, тусгай цаг хянадаг тоног төхөөрөмж дээр тулгуурласан NTP серверийн хувьд хэрэглэнэ.

`driftfile` тохируулгаар ямар файлд системийн цагийн алдах зөрүү утгыг хадгалж байгааг заана. ntpd(8) програм энэ утгыг ашиглан цагийн алдсан зөрүүг автоматаар нөхнө. Ингэснээр цагийн бүх гадаад эх үүсвэрүүдтэй холбоо тогтоох боломжгүй болсон үед, хэсэг хугацааны туршид ч гэсэн цагийг харьцангуй зөв ажиллуулах боломжийг олгоно.

`driftfile` тохируулгаар ямар файлд таны зааж өгсөн NTP серверүүдийн өмнөх хариунуудын тухай мэдээллийг хадгалж байгааг заана. Энэ файлд NTP-н дотоод үйл ажиллагааны мэдээллийг хадгалдаг. Энэ мэдээллийг өөр ямар ч процесс өөрчлөх ёсгүй.

#### 30.10.3.3. Өөрийн Сервер рүү Хандах Хандалтыг Хянах нь

Анхдагч байдлаар, таны NTP сервер рүү Интернэтэд байгаа бүх хост хандах боломжтой. /etc/ntp.conf файл дотор `restrict` тохируулгаар ямар машинууд таны сервер рүү хандаж болохыг хянаж болно.

Хэрэв та өөрийн NTP сервер рүү хэнийг ч хандуулахыг хүсэхгүй байгаа бол /etc/ntp.conf файл дотор дараах мөрийг нэмэх хэрэгтэй:

```
restrict default ignore
```

Энэ нь таны серверээс өөрийн чинь локал тохиргоонд жагсаагдсан аль ч сервер үрүү хандах боломжийг бас хаана. Хэрэв та өөрийн NTP серверийг гадаад NTP сервертэй синхрончлох хэрэгтэй бол ямар нэг серверийг зөвшөөрөх ёстой. Дэлгэрэнгүй мэдээллийг ntp.conf(5) гарын авлагаас үзнэ үү.

Хэрэв та зөвхөн өөрийн сүлжээнд байгаа машинуудыг таны сервертэй цагаа тааруулахыг зөвшөөрөөд, гэхдээ таны серверийн тохиргоог өөрчлөх болон тэгш эрхтэй серверүүд шиг цагийн мэдээллийг хуваахыг зөвшөөрөхгүй бол дээр дурдсаны оронд:

```
restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap
```

гэсэн мөрийг бичнэ үү. Энд `192.168.1.0` нь таны сүлжээний IP хаяг, `255.255.255.0` нь таны сүлжээний баг болно.

/etc/ntp.conf дотор олон тооны `restrict` тохируулгууд байж болно. Илүү дэлгэрэнгүй мэдээллийг ntp.conf(5)-н `Хандалтыг Удирдах Дэмжлэг` дэд хэсгээс үзнэ үү.

### 30.10.4. NTP Серверийг Ажиллуулах нь

NTP серверийг систем ачаалах үед эхлүүлэхийн тулд, `ntpd_enable="YES"` гэсэн мөрийг /etc/rc.conf файлд нэмж бичих хэрэгтэй. Хэрэв ntpd(8)-д нэмэлт тугуудыг өгөх хүсэлтэй бол, /etc/rc.conf файлд байгаа `ntpd_flags` параметрийг засах хэрэгтэй.

Машиныг дахин ачаалалгүйгээр серверийг эхлүүлэхийн тулд, `ntpd` тушаалыг /etc/rc.conf-д

заасан `ntpd_flags` нэмэлт параметрүүдийн хамтаар өгөх хэрэгтэй. Жишээлбэл:

```
# ntpd -p /var/run/ntpd.pid
```

### 30.10.5. ntpd-г Түр зуурын Интернэт Холболттой үед Хэрэглэх нь

ntpd(8) програм зөв ажиллахын тулд байнгын Интернэт холболт шаардлагагүй. Гэхдээ, хэрэгцээтэй үедээ гадагшаа залгадаг тийм төрлийн түр зуурын холболттой бол, NTP трафикийг гадагшаа залгах болон холболтыг бариад байхаас сэргийлэх нь чухал. Хэрэв та PPP хэрэглэдэг бол, /etc/ppp/ppp.conf файл дотор байгаа `filter` директивийг ашиглаж болно. Жишээ нь:

```
set filter dial 0 deny udp src eq 123
# Prevent NTP traffic from initiating dial out
set filter dial 1 permit 0 0
set filter alive 0 deny udp src eq 123
# Prevent incoming NTP traffic from keeping the connection open
set filter alive 1 deny udp dst eq 123
# Prevent outgoing NTP traffic from keeping the connection open
set filter alive 2 permit 0/0 0/0
```

Дэлгэрэнгүй мэдээллийг ppp(8)-н `PACKET FILTERING` хэсгээс болон /usr/shared/examples/ppp/-д байгаа жишээнүүдээс авч болно.

Зарим Интернэт үйлчилгээ үзүүлэгчид бага дугаартай портуудыг хаасан байдаг бөгөөд ингэснээр хариу нь таны машинд хэзээ ч хүрэхгүй болж NTP ажиллахгүй болдог.

### 30.10.6. Цааших Мэдээлэл

NTP серверийн баримтжуулалтыг HTML хэлбэрээр /usr/shared/doc/ntp/-с олж үзэж болно.

## 30.11. `syslogd` ашиглан алсын хост руу бүртгэх нь

Системийн бүртгэлтэй ажиллах нь аюулгүй байдлын болоод системийг удирдах ажиллагааны чухал асуудал юм. Хостууд дунд зэргийн эсвэл том сүлжээнд тархсан эсвэл тэдгээр нь төрөл бүрийн олон янзын сүлжээний хэсэг болсон байх тохиолдолд эдгээр олон хостын бүртгэлийн файлуудыг монитор хийх нь ихээхэн төвөгтэй болдог. Энэ тохиолдолд алсаас бүртгэхийг тохируулах нь бүх л процессийг илүү тухтай болгодог.

Тусгайлан заасан бүртгэх хост руу төвлөрүүлэн бүртгэх нь бүртгэлийн файлын удирдлагатай холбоотой зарим хүндрэлүүдийг багасгаж чаддаг. syslogd(8) болон newsyslog(8) зэрэг FreeBSD-ийн эх хэрэгслүүдийг ашиглан бүртгэлийн файлын цуглуулга, нийлүүлэлт болон багасгалтыг нэг газар тохируулж болдог. Дараах жишээ тохиргоонд `logserv.example.com` гэж нэрлэгдсэн хост `A` локал сүлжээнээс бүртгэлийн мэдээллийг цуглуулах болно. `logclient.example.com` гэж нэрлэгдсэн хост `B` бүртгэлийн мэдээллийг сервер систем рүү

дамжуулах болно. Жинхэнэ тохиргоонд эдгээр хостууд зохих дамжуулах болон буцах DNS эсвэл /etc/hosts файлд оруулгууд шаардана. Тэгэхгүй бол өгөгдлийг сервер хүлээн авахгүй татгалзах болно.

### 30.11.1. Бүртгэлийн серверийн тохиргоо

Бүртгэлийн серверүүд нь алсын хостуудаас бүртгэлийн мэдээллийг хүлээн авахаар тохируулагдсан машинууд юм. Ихэнх тохиолдолд энэ нь тохиргоог хялбар болгох зорилготой бөгөөд зарим тохиолдолд энэ нь удирдлагыг арай сайжруулж байгаа хэлбэр байж болох юм. Аль ч шалтгаан байсан гэсэн үргэлжлүүлэхээсээ өмнө цөөн хэдэн шаардлагыг дурдъя.

Зөв тохируулсан бүртгэлийн сервер дараах хамгийн бага шаардлагыг хангасан байх шаардлагатай:

- Клиент болон сервер дээр 514-р порт руу UDP-г дамжуулах боломжийг бүрдүүлэх галт хананы дүрэм;
- Клиент машинаас алсын мэдэгдлүүдийг хүлээн авахаар syslogd тохируулагдсан байх;
- syslogd сервер болон бүх клиент машинууд нь дамжуулах болон буцах DNS-ийн хувьд зөв оруулгуудтай эсвэл /etc/hosts файлд зөв тохируулсан байх шаардлагатай.

Бүртгэлийн серверийг тохируулахын тулд клиент нь /etc/syslog.conf-д нэмэгдсэн байх ёстой бөгөөд бүртгэх боломжийг зааж өгсөн байх шаардлагатай:

```
+logclient.example.com
*.*     /var/log/logclient.log
```

Төрөл бүрийн дэмжигдсэн, байгаа *facility* буюу *боломжуудын* талаарх дэлгэрэнгүй мэдээллийг syslog.conf(5) гарын авлагын хуудаснаас олж болно.

Нэмсэний дараа бүх `facility` мэдэгдлүүд өмнө заасан /var/log/logclient.log файл руу бүртгэгдэх болно.

Сервер машин дараах тохиргоог бас /etc/rc.conf файлдаа хийсэн байх шаардлагатай:

```
syslogd_enable="YES"
syslogd_flags="-a logclient.example.com -v -v"
```

Эхний тохиргоо нь `syslogd` демоныг эхлүүлэхийг заах бөгөөд хоёр дахь нь клиетийн өгөгдлийг энэ сервер дээр хүлээн авахыг зөвшөөрнө. Сүүлийн `-v -v` хэсэг нь бүртгэж байгаа мэдэгдлүүдийн гаралтыг илүү дэлгэрэнгүй болгоно. Энэ нь facility-г тохируулахад ихээхэн ашигтай байдаг. Администраторууд ямар төрлийн мэдэгдлүүд ямар facility-р бүртгэгдэж байгааг хянах боломжийг энэ нь бүрдүүлдэг.

Олон клиентээс бүртгэлийг хүлээн авахын тулд олон `-a` сонголтыг зааж өгч болно. IP хаягууд болон бүхэл сүлжээний блокийг бас зааж өгч болох бөгөөд боломжит сонголтуудын бүх

жагсаалтыг syslog(3) гарын авлагын хуудаснаас үзнэ үү.

Төгсгөлд нь бүртгэлийн файлыг үүсгэх хэрэгтэй. Хэрэглэгсэн арга нь хамаагүй боловч touch(1) үүнтэй адил тохиолдлуудад сайн ажилладаг:

```
# touch /var/log/logclient.log
```

Энэ үед `syslogd` демоныг дахин ажиллуулж шалгах ёстой:

```
# service syslogd restart
# pgrep syslog
```

Хэрэв PID буцаагдвал сервер нь амжилттай дахин эхэлсэн гэсэн үг бөгөөд клиентийн тохиргоо ажиллаж эхэлнэ. Хэрэв сервер дахин эхлээгүй бол ямар нэг зүйл болсон эсэхийг /var/log/messages файл дахь мэдэгдлүүдээс шалгаарай.

### 30.11.2. Клиентийн бүртгэлийн тохиргоо

Бүртгэл илгээгч клиент нь өөр дээрээ хуулбараа үлдээхээс гадна бас бүртгэлийн сервер рүү бүртгэлийн мэдээллийг явуулдаг машин юм.

Бүртгэлийн серверүүдийн нэгэн адил клиентүүд нь бас хамгийн бага шаардлагыг хангасан байх ёстой:

- syslogd(8) нь бүртгэлийн сервер хүлээн авах ёстой заасан төрлийн мэдэгдлүүдийг бүртгэлийн сервер рүү илгээхээр тохируулагдсан байх ёстой;
- Галт хана UDP пакетуудыг 514-р порт руу зөвшөөрөх ёстой;
- Дамжуулах болон буцах DNS тохируулагдсан эсвэл /etc/hosts файл зохих оруулгуудтай байх шаардлагатай.

Клиентийн тохиргоо нь серверийнхтэй харьцуулах юм бол арай зөөлөн байдаг. Клиент машин нь /etc/rc.conf файлдаа дараахийг нэмж өгсөн байх шаардлагатай байдаг:

```
syslogd_enable="YES"
syslogd_flags="-s -v -v"
```

Өмнө дурдсаны адил эдгээр тохиргоонууд нь `syslogd` демоныг ачаалж эхлэхэд эхлүүлэхийг заах бөгөөд бүртгэх мэдэгдлүүдийг дэлгэрэнгүйгээр харуулах болно. `-s` сонголт нь бусад хостуудаас бүртгэлийг энэ клиент хүлээн авахаас сэргийлдэг.

Facility нь мэдэгдэл үүсгэгдэж байгаа тэр системийн хэсгийг тайлбарладаг. Жишээ нь ftp болон ipfw нь хоёулаа facility юм. Эдгээр хоёр үйлчилгээний хувьд бүртгэлийн мэдэгдлүүд үүсэхэд ихэвчлэн дээрх хоёр хэрэгслийг бүртгэлийн мэдэгдэл бүртээ агуулсан байдаг. Facility нь бүртгэлийн мэдэгдэл ямар чухлыг тэмдэглэхэд хэрэглэгдэх дараалал эсвэл түвшинтэй байдаг. Хамгийн түгээмэл нь `warning` ба `info` юм. Боломжит бүх facilty болон дарааллуудын жагсаалтыг syslog(3) гарын авлагын хуудаснаас үзнэ үү.

Бүртгэлийн серверийг клиентийн /etc/syslog.conf файлд заасан байх шаардлагатай. Энэ жишээн дээр алсын сервер рүү бүртгэлийн өгөгдлийг илгээхийн тулд `@` тэмдгийг ашигласан бөгөөд доор дурдсан мөртэй төстэй харагдана:

```
*.*     @logserv.example.com
```

Нэмсэний дараа өөрчлөлтийг хүчинтэй болгохын тулд `syslogd`-г дахин эхлүүлэх шаардлагатай:

```
# service syslogd restart
```

Сүлжээгээр бүртгэлийн мэдэгдлүүдийг илгээж байгаа эсэхийг тест хийхийн тулд клиент дээр logger(1)-г ашиглаж мэдэгдлийг `syslogd` руу илгээнэ:

```
# logger "Test message from logclient"
```

Энэ мэдэгдэл клиент дээрх /var/log/messages болон сервер дээрх /var/log/logclient.log файлд одоо орсон байх ёстой.

### 30.11.3. Бүртгэлийн серверүүдийг дибаг хийх

Зарим тохиолдолд хэрэв бүртгэлийн сервер дээр мэдэгдлүүд нь хүлээн авагдаагүй бол дибаг хийх шаардлагатай байж болох юм. Хэд хэдэн шалтгаанаас болж ийм байдалд хүрч болох юм. Хамгийн түгээмэл хоёр нь сүлжээний холболтын болон DNS-тэй холбоотой асуудлууд юм. Эдгээр тохиолдлуудыг тест хийхийн тулд хоёр хост хоёулаа /etc/rc.conf файлд заагдсан хостын нэрээрээ нэг нэгэн рүүгээ хүрч чадаж байгааг шалгах хэрэгтэй. Хэрэв энэ зөв ажиллаж байгаа бол /etc/rc.conf файлд `syslogd_flags` тохиргоог өөрчлөх шаардлагатай болно.

Дараах жишээн дээр /var/log/logclient.log нь хоосон бөгөөд /var/log/messages файл нь амжилттүй болсон шалтгааныг харуулна. Дибаг хийж байгаа гаралтыг илүү дэлгэрэнгүй харуулахын тулд дараах жишээтэй төстэйгөөр `syslogd_flags` тохируулгыг өөрчилж дахин ачаалах хэрэгтэй:

```
syslogd_flags="-d -a logclien.example.com -v -v"
```

```
# service syslogd restart
```

Доор дурдсантай төстэй дибаг өгөгдөл дахин ачаалсны дараа дэлгэц дээр хурдан гарч өнгөрнө:

```
logmsg: pri 56, flags 4, from logserv.example.com, msg syslogd: restart
syslogd: restarted
logmsg: pri 6, flags 4, from logserv.example.com, msg syslogd: kernel boot file is
```

```
/boot/kernel/kernel
Logging to FILE /var/log/messages
syslogd: kernel boot file is /boot/kernel/kernel
cvthname(192.168.1.10)
validate: dgram from IP 192.168.1.10, port 514, name logclient.example.com;
rejected in rule 0 due to name mismatch.
```

Мэдэгдлүүд нэр зөрснөөс болоод дамжихгүй байгааг эндээс харж болно. Тохиргоог алхам алхмаар дахин шалгасны дараа /etc/rc.conf дахь дараах мөр буруу бичигдсэн бөгөөд асуудалтай байгааг олж харна:

```
syslogd_flags="-d -a logclien.example.com -v -v"
```

Энэ мөр `logclien` биш `logclient` гэдгийг агуулсан байх ёстой. Зөв болгож засан дахин ачаалсны дараа хүлээж байсан үр дүнгээ харах болно:

```
# service syslogd restart
logmsg: pri 56, flags 4, from logserv.example.com, msg syslogd: restart
syslogd: restarted
logmsg: pri 6, flags 4, from logserv.example.com, msg syslogd: kernel boot file is
/boot/kernel/kernel
syslogd: kernel boot file is /boot/kernel/kernel
logmsg: pri 166, flags 17, from logserv.example.com,
msg Dec 10 20:55:02 <syslog.err> logserv.example.com syslogd: exiting on signal 2
cvthname(192.168.1.10)
validate: dgram from IP 192.168.1.10, port 514, name logclient.example.com;
accepted in rule 0.
logmsg: pri 15, flags 0, from logclient.example.com, msg Dec 11 02:01:28 trhodes: Test
message 2
Logging to FILE /var/log/logclient.log
Logging to FILE /var/log/messages
```

Энэ үед мэдэгдлүүдийг зөв хүлээн аван зөв файлд бичих болно.

### 30.11.4. Аюулгүй байдлын хувьд бодолцох зүйлс

Сүлжээний аль ч үйлчилгээний нэгэн адил энэ тохиргоог хийхээсээ өмнө аюулгүй байдлын шаардлагуудыг бодолцох ёстой. Заримдаа бүртгэлийн файлууд нь локал хост дээр идэвхжүүлсэн үйлчилгээнүүд, хэрэглэгчдийн бүртгэл болон тохиргооны өгөгдлийн талаарх эмзэг өгөгдлүүдийг агуулсан байж болох юм. Клиентээс сервер рүү илгээсэн сүлжээний өгөгдөл нь шифрлэгдээгүй эсвэл нууц үгээр хамгаалагдаагүй байдаг. Хэрэв шифрлэх шаардлагатай бол өгөгдлийг шифрлэсэн хоолойгоор дамжуулах security/stunnel хэрэгслийг ашиглаж болох юм.

Локал аюулгүй байдал нь бас л асуудал юм. Бүртгэлийн файлууд нь хэрэглэж байхад юм уу эсвэл бүртгэлийн багасгах үед шифрлэгддэггүй. Локал хэрэглэгчид эдгээр файлуудад хандаж системийн тохиргооны талаар нэмэлт мэдээлэл олж авч болох юм. Ийм тохиолдолд эдгээр

файлууд дээр зөв зөвшөөрлүүдийг тавих нь чухал юм. newsyslog(8) хэрэгсэл нь шинээр үүсгэгдсэн болон багасгагдсан бүртгэлийн файлууд дээр зөвшөөрөл тавихыг дэмждэг. Бүртгэлийн файлууд дээр `600` горимыг тавьснаар хүсээгүй локал хэрэглэгчид тэдгээрийг шиншлэх боломжийг хаах юм.

# Chapter 31. Галт хана

## 31.1. Танилцуулга

Галт ханын тусламжтайгаар систем уруу орж байгаа болон түүнээс гарч байгаа өгөгдлийн урсгалыг шүүн нэвтрүүлэх боломжтой болдог. Галт хана нь сүлжээгээр дамжин өнгөрч байгаа пакетуудыг, "дүрмүүдэд" заасны дагуу эсвэл нэвтрүүлэх, эсвэл хаах үүргийг гүйцэтгэдэг. Галт ханын дүрмүүд нь пакетийг протоколын төрөл, эхлэл хост хаяг, очих хост хаяг, эхлэл порт хаяг, очих порт хаяг зэрэг хэд хэдэн шинжээр нь шинжлэх боломжийг олгодог.

Галт ханыг ашигласнаар тухайн хостын болон сүлжээний аюулгүй байдлыг нилээд нэмэгдүүлж чадна. Галт ханын тусламжтайгаар дараах зүйлсийг хийх боломжтой :

- Дотоод сүлжээнд байрлаж байгаа сервер машин, түүн дээр ажиллаж байгаа програм үйлчилгээг Интернэтээр дамжин орж ирж буй гадны урсгалаас хамгаалах, тусгаарлах.
- Дотоод сүлжээнд байрлаж байгаа хостоос Интернэт уруу хандах хандалтыг хаах, хязгаарлах.
- Network address translation буюу Сүлжээний Хаягийн Хөрвүүлэлтийг (NAT) дэмжих. Өөрөөр хэлбэл дотоод сүлжээндээ хувийн IP хаяг хэрэглэж, Интернэтэд гарахдаа дундаа нэг холболтыг (нэг IP хаяг эсвэл автоматаар оноосон бүлэг хаягаар) хуваан хэрэглэх.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- пакетийг шүүн нэвтрүүлэх дүрмүүдийг хэрхэн оновчтойгоор тодорхойлох.
- FreeBSD-тэй хамт суусан галт хананууудын ялгаа.
- OpenBSD-н PF галт ханыг хэрхэн тохируулах болон хэрэглэх.
- IPFILTER-г хэрхэн тохируулах болон хэрэглэх.
- IPFW-г хэрхэн тохируулах болон хэрэглэх.

Энэ бүлгийг уншихаасаа өмнө, та дараах зүйлсийг мэдсэн байх шаардлагатай:

- FreeBSD болон Интернэтийн тухай үндсэн ойлголт.

## 31.2. Галт ханын тухай ойлголтууд

Галт ханын дүрмүүдийг дараах үндсэн хоёр янзаар үүсгэж болно: "inclusive буюу хамааруулсан" эсвэл "exclusive буюу хамааруулаагүй". Хамааруулаагүй галт хана нь дүрмэнд тохирсон урсгалаас бусдыг нэвтрүүлнэ. Харин хамааруулсан галт хана бол эсрэгээр нь, дүрмэнд тохирсон урсгалыг нэвтрүүлж бусдыг хаана.

Хамааруулсан галт хана нь Интернэтэд үйлчилгээнүүдийг санал болгодог системүүдийн хувьд илүү сайн сонголт болдог бөгөөд гарч байгаа урсгалыг илүү сайн хянах боломжийг олгодог. Энэ нь Интернэтээс таны хувийн сүлжээ рүү хандах урсгалыг бас хянадаг. Дүрэмд харгалзаж тохирохгүй бүх урсгалыг хааж бүртгэдэг. Болохгүй урсгалыг галт ханаар нэвтрүүлэх эрсдэлийг багасгадаг учраас хамааруулсан галт хана нь хамааруулаагүй галт

ханыг бодвол илүүтэйгээр аюулгүй байдлыг хангаж чаддаг.

Зааж хэлээгүй л бол энэ бүлгийн бүх тохиргоо болон дүрмүүд нь хамааруулсан галт ханыг үүсгэдэг.

"Төлөвт галт ханыг" ашиглан аюулгүй байдлыг цааш илүү сайжруулах боломжтой. Энэ төрлийн галт хана нь галт ханаар дамжин тогтсон холболтуудыг бүртгэж, зөвхөн таарч байгаа тогтсон холболтоор эсвэл шинэ холболт үүсгэн урсгалыг нэвтрүүлдэг. Төлөвт галт ханын нэг дутагдалтай тал гэвэл олон шинэ холболтууд нэг дор тогтох үед Denial of Service буюу Үйлчилгээг Зогсоох(DoS) халдлагад өртөмтгий болдог. Иймээс галт ханыг зохион байгуулахдаа төлөвт ба төлөвт-бус байдлыг хослуулан хэрэглэх нь хамгийн оновчтой байдаг.

## 31.3. Галт ханын багцууд

FreeBSD дээр гурван янзын галт ханын багцууд хамрагдсан байдаг. Нэрлэвэл: *IPFILTER* (IPF гэж нэрлэх нь элбэг), *IPFIREWALL* (IPFW гэж нэрлэх нь элбэг), ба *OpenBSD-н PacketFilter* (PF гэж нэрлэх нь элбэг). FreeBSD нь мөн урсгалыг хязгаарлах(үндсэндээ зурвасын өргөнийг хязгаарлах) хоёр багцын хамт ирдэг: altq(4) болон dummynet(4). Dummynet нь анхнаасаа IPFW-тай, харин ALTQ нь PF-тэй нягт холбоотой ажилладаг. IPFILTER-ийн хувьд урсгал хязгаарлалтыг хийхдээ NAT болон шүүлтэд IPFILTER-ийг ба IPFW-ийг dummynet(4)-тэй цуг юм уу *эсвэл*PF-ийг ALTQ-тай цуг ашиглан хийж болно. IPFW, ба PF нь бүгд систем уруу орж байгаа болон гарч байгаа урсгалыг дүрмүүдийн тусламжтай удирдах боловч синтаксын хувьд ч, арга замын хувьд ч өөр өөр байдаг.

FreeBSD дээр олон галт ханын багцууд хамт ирдэг нь өөр өөр хэрэгцээ шаардлагатай хүмүүст хүртээмжтэй байхыг гол зорилгоо болгосонд оршино. Түүнээс аль ч галт хана нь нөгөөгөөсөө илүү, эсвэл дутуу гэсэн үг биш юм.

Зохиогч IPFILTER-г сонгон авсан нь түүний төлөвт дүрмүүд нь NAT орчинд хэрэглэхэд төвөг багатай, мөн дотроо ftp proxy агуулсан байдгаас болсон хэрэг. Энэхүү ftp proxy-г ашиглан гадагшаа гарах FTP урсгалыг зөвшөөрсөн дүрмүүдийг бичихэд хялбар байдаг.

Бүх галт ханууд пакет удирдах талбарын утгыг шинжлэх зарчмаар ажиллах тул галт ханын дүрмүүдийг бичихийн өмнө TCP/IP протокол хэрхэн ажилладаг талаар болон пакет удирдах талбарын утгууд, энэ утгууд session буюу сесс үүсэхэд хэрхэн хэрэглэгддэг талаар үндсэн ойлголттой байх шаардлагатай болдог. Дээрх ойлголтуудын талаар дараах хаягаар орж уншина уу: http://www.ipprimer.com/overview.cfm.

## 31.4. OpenBSD Пакет шүүгч (PF) ба ALTQ

2003 оны 7 сард OpenBSD-н галт ханын програм болох PF FreeBSD уруу шилжиж, FreeBSD Портын Цуглуулгад орсон. 2004 онд гарсан FreeBSD 5.3 нь PF-г үндсэн системийн багцын нэг хэсэг болгон оруулсан анхны хувилбар юм. PF нь бүрэн хэмжээнд ажиллах чадвартай галт хана бөгөөд ALTQ-тай (Alternate Queuing буюу Ээлжлэн солигдох дараалал) хамтран ажиллах боломжтой. ALTQ нь Quality of Service буюу Үйлчилгээний Чанарын (QoS) боломжоор хангадаг. OpenBSD Төсөл нь PF FAQ-г хөтлөн явуулдаг. Тиймээс гарын авлагын энэ хэсэг нь FreeBSD-д хамаатай PF дээр илүү анхаарлаа хандуулахаас гадна хэрэглээний талаар зарим

нэг ерөнхий мэдээллийг өгнө. Хэрэглээний мэдээллийн талаар илүү дэлгэрэнгүйг PF FAQ-с үзнэ үү.

FreeBSD-д зориулсан PF-ийн талаар илүү дэлгэрэнгүй мэдээллийг http://pf4freebsd.love2party.net/ хаягаас үзэж болно.

### 31.4.1. Цөмийн дуудагдах PF модулиудыг ашиглах нь

PF цөмийн модулийг дуудахдаа /etc/rc.conf файлд дараах мөрийг нэмнэ:

```
pf_enable="YES"
```

Дараа нь модулийг дуудахдаа эхлүүлэх скриптийг ажиллуулна:

```
# /etc/rc.d/pf start
```

PF модуль нь дүрмийн тохиргооны файлаа олж чадахгүй бол дуудагдахгүйг санаарай. Анхдагч байрлал нь /etc/pf.conf байна. Хэрэв PF дүрмийн олонлог өөр хаа нэгтээ байгаа бол /etc/rc.conf файлд доор дурдсантай адил мөрийг нэмж PF дүрмийн олонлогоо тэр газраас хайхаар зааж өгч болно:

```
pf_rules="/path/to/pf.conf"
```

Жишээ pf.conf файлыг /usr/shared/examples/pf/ сангаас олж болно.

PF модулийг тушаалын мөрөөс бас дуудан ажиллуулж болно:

```
# kldload pf.ko
```

PF-д зориулсан бүртгэл хөтлөх дэмжлэгийг `pflog.ko` хангадаг бөгөөд /etc/rc.conf файлд дараах мөрийг нэмж дуудаж болно:

```
pflog_enable="YES"
```

Дараа нь модулийг дуудахдаа эхлүүлэх скриптийг ажиллуулна:

```
# /etc/rc.d/pflog start
```

Хэрэв танд PF-ийн өөр боломжууд хэрэгтэй бол PF-ийн дэмжлэгийг цөмд оруулан эмхэтгэх хэрэгтэй.

### 31.4.2. PF цөмийн тохиргоонууд

PF дэмжлэгийг FreeBSD цөмд оруулж эмхэтгэх нь шаардлагагүй боловч дуудагдах модульд ороогүй байдаг PF-ийн нэмэлт боломжуудын нэг бөгөөд PF-ийн ашигладаг төлвийн хүснэгтэд зарим өөрчлөлтүүдийг ил гаргадаг псевдо төхөөрөмж болох pfsync(4) гэгддэг боломжийг ашиглахын тулд та тэгж хийж өгч болох юм. Үүнийг carp(4)-тэй хослуулан ажиллагаа доголдоход тойрон гарах галт ханаыг PF ашиглан бүтээж болно. CARP-ийн талаар дэлгэрэнгүйг гарын авлагын Common Address Redundancy Protocol (CARP)-с үзэж болно.

Цөмийн PF тохиргоонуудыг /usr/src/sys/conf/NOTES-с олж болох бөгөөд доор үзүүлэв:

```
device pf
device pflog
device pfsync
```

`device pf` тохиргоо "Packet Filter" галт ханыг (pf(4)) дэмждэг болгоно.

`device pflog` тохиргоо псевдо буюу хуурамч pflog(4) сүлжээний төхөөрөмжийг идэвхжүүлнэ. Энэ хуурамч төхөөрөмжийн тусламжтайгаар bpf(4) дескриптор уруу урсгалыг бүртгэх боломжтой. pflogd(8) дэмонг бүртгэлийг дискэн дээр хадгалахад хэрэглэнэ.

`device pfsync` тохиргоо псевдо буюу хуурамч pfsync(4) сүлжээний төхөөрөмжийг идэвхжүүлнэ. Энэ хуурамч төхөөрөмжийн тусламжтайгаар "төлвийн өөрчлөлтүүдийг" хянах боломжтой.

### 31.4.3. rc.conf боломжууд

Дараах rc.conf(5) илэрхийллүүд PF болон pflog(4)-ийг ачаалах үед тохируулна:

```
pf_enable="YES"                 # Enable PF (load module if required)
pf_rules="/etc/pf.conf"         # rules definition file for pf
pf_flags=""                     # additional flags for pfctl startup
pflog_enable="YES"              # start pflogd(8)
pflog_logfile="/var/log/pflog"  # where pflogd should store the logfile
pflog_flags=""                  # additional flags for pflogd startup
```

Хэрвээ энэ галт ханын цаана LAN байгаа бөгөөд LAN-д байгаа компьютерууд уруу пакет дамжуулах шаардлагатай бол эсвэл NAT ашиглах бодолтой байгаа бол дараах илэрхийлэл танд бас хэрэгтэй:

```
gateway_enable="YES"            # Enable as LAN gateway
```

### 31.4.4. Шүүгч дүрмүүдийг үүсгэх нь

PF нь pf.conf(5)-с (анхдагчаар /etc/pf.conf) өөрийн тохиргооны дүрмүүдийг унших бөгөөд тэнд заагдсан дүрмүүд буюу тодорхойлолтуудын дагуу пакетуудыг өөрчлөх, орхих буюу эсвэл

дамжуулдаг. FreeBSD суулгацад /usr/shared/examples/pf/-д байрлах хэд хэдэн жишээ файлууд байдаг. PF-ийн дүрмийн олонлогуудын талаар бүрэн мэдээллийг PF FAQ-с лавлана уу.

PF FAQ-г үзэж байхдаа FreeBSD-ийн хувилбар бүр өөр өөр PF хувилбартай байж болохыг анхаарах хэрэгтэй. Одоогоор FreeBSD 8.*X* болон түүнээс өмнөх хувилбарууд OpenBSD 4.1-ийн нэгэн адил PF-ийн хувилбарыг ашигладаг. FreeBSD 9.*X* болон түүнээс хойшхи хувилбарууд OpenBSD 4.5-ийн нэгэн адил PF-ийн хувилбарыг ашигладаг.

FreeBSD пакет шүүгчийн захидлын жагсаалт нь PF галт ханыг тохируулж ажиллуулах талаар асуухад тохиромжтой газар юм. Асуулт асуухаасаа өмнө захидлын жагсаалтын архиваас шалгахаа мартуузай!

### 31.4.5. PF-тэй ажиллах нь

PF-ийг хянахдаа pfctl(8)-г ашиглана. Зарим нэг хэрэгтэй тушаалуудыг доор жагсаав (Бүх боломжит тохиргоонуудын талаар pfctl(8) гарын авлагын хуудаснаас лавлахаа мартуузай):

| Тушаал | Зорилго |
|---|---|
| `pfctl -e` | PF-г идэвхжүүлэх |
| `pfctl -d` | PF-г болиулах |
| `pfctl -F all -f /etc/pf.conf` | Бүх дүрмүүдийг арилгаж (nat, шүүх, төлөв, хүснэгт, гэх мэт.) /etc/pf.conf файлаас дахин ачаалах |
| `pfctl -s [ rules \| nat \| state ]` | Шүүх дүрмүүд, nat дүрмүүд, эсвэл төлвийн хүснэгтийн талаар тайлан гаргах |
| `pfctl -vnf /etc/pf.conf` | Дүрмийн олонлогийг ачаалалгүйгээр /etc/pf.conf-д алдаа байгаа эсэхийг шалгах |

### 31.4.6. ALTQ-г идэвхжүүлэх

ALTQ-г идэвхжүүлэх ганц арга зам бол түүний боломжуудыг FreeBSD цөмтэй хамт хөрвүүлэн эмхэтгэх юм. Мөн сүлжээний картын драйвер болгон ALTQ-г дэмждэггүй тул өөрийн тань хэрэглэж буй FreeBSD хувилбарын хувьд дэмжигддэг драйверуудын жагсаалтыг altq(4) гарын авлагын хуудаснаас үзнэ үү.

Дараах тохируулгууд ALTQ-г идэвхжүүлж нэмэлт үүргүүдийг оруулдаг.

```
options         ALTQ
options         ALTQ_CBQ        # Class Bases Queuing (CBQ)
options         ALTQ_RED        # Random Early Detection (RED)
options         ALTQ_RIO        # RED In/Out
options         ALTQ_HFSC       # Hierarchical Packet Scheduler (HFSC)
options         ALTQ_PRIQ       # Priority Queuing (PRIQ)
options         ALTQ_NOPCC      # Required for SMP build
```

`options ALTQ` мөр ALTQ -г бүхэлд нь идэвхжүүлнэ.

`options ALTQ_CBQ` мөр *Class Based Queuing* буюу Ангиллаас Хамаарсан Дараалал Үүсгэх(CBQ) боломжийг идэвхжүүлнэ. CBQ нь шүүгч дүрмүүд дээр үндэслэн урсгалуудад эрэмбэ тогтоох зорилгоор зурвасын өргөнийг өөр өөр ангиллуудад болон дарааллуудад хуваах боломжийг олгоно.

`options ALTQ_RED` мөр *Random Early Detection* буюу Санамсаргүй Эрт Илрүүлэлт(RED)-г идэвхжүүлнэ. RED-г сүлжээний даац хэтрэхээс сэргийлэхэд хэрэглэдэг. RED дарааллын уртыг хэмжиж, түүнийг байх ёстой дээд ба доод хэмжээтэй жиших байдлаар ажилладаг. Хэрэв дараалал дээд хэмжээнээс урт болбол шинэ пакетууд орхигдох болно. Нэртэйгээ адилаар, RED нь холболтуудаас пакетийг санамсаргүйгээр орхигдуулдаг.

`options ALTQ_RIO` мөр нь *Random Early Detection In and Out* буюу Орох ба Гарах Санамсаргүй Эрт Илрүүлэлтийг идэвхжүүлнэ.

`options ALTQ_HFSC` мөр нь *Hierarchical Fair Service Curve Packet Scheduler* буюу Ye давхаргат Зохимжит Үйлчилгээний Муруйн Пакет Хуваарилагчийг идэвхжүүлнэ. HFSC талаар илүү дэлгэрэнгүй мэдээллийг дараах хаягаас үзнэ үү: http://www-2.cs.cmu.edu/~hzhang/HFSC/main.html.

`options ALTQ_PRIQ` мөр нь *Priority Queuing* буюу Эрэмбэт Дараалал Үүсгэх (PRIQ)-г идэвхжүүлнэ. PRIQ нь эрэмбэ өндөртэй дараалалд байгаа урсгалыг эхэнд нэвтрүүлэх зарчмаар ажилладаг.

`options ALTQ_NOPCC` мөр нь ALTQ-г SMP-тай хамт ажиллах боломжтой болгоно. SMP системийн хувьд энэ боломжийг заавал идэвхжүүлэх хэрэгтэй.

## 31.5. IPFILTER (IPF) Галт хана

IPFILTER-г зохиосон хүн бол Даррин Рид билээ. IPFILTER нь үйлдлийн системээс хамааралгүй: нээлттэй эхийн програм бөгөөд FreeBSD, NetBSD, OpenBSD, SunOS™, HP/UX, ба Solaris™ зэрэг олон үйлдлийн систем уруу шилжүүлэгдсэн юм. IPFILTER эрчимтэй дэмжигдэж, сайжруулсан хувилбарууд нь тогтмол гарсаар байгаа.

IPFILTER нь цөмийн талд ажиллах галт хана болон NAT механизм дээр суурилсан бөгөөд түүнийг удирдах, хянахын тулд хэрэглэгчийн интерфэйс програмыг ашиглана. Галт ханын дүрмүүдийг нэмэх болон хасахдаа ipf(8) хэрэгслийг хэрэглэнэ. NAT дүрмүүдийг нэмэх болон хасахдаа ipnat(8) хэрэгслийг хэрэглэнэ. ipfstat(8) хэрэгсэл нь IPFILTER-н цөмийн талд ажиллаж байгаа хэсгийн статистикийг хэвлэхэд зориулагдсан. ipmon(8) програм харин IPFILTER-н үйлдлүүдийг системийн бүртгэлийн файлд бүртгэнэ.

IPF-г анх зохиохдоо "сүүлд тохирсон дүрэм дийлнэ" гэсэн логикийн дагуу бүтээсэн ба зөвхөн төлөвт-бус дүрмүүдийг ашигладаг байлаа. Цаг хугацаа өнгөрөхөд IPF илүү хөгжиж, "quick" тохируулга болон төлөвт "keep state" тохируулгуудыг агуулах болсон нь орчин үеийн хэрэгцээ шаардлагад илүү нийцэх болжээ. IPF-н албан ёсны баримтжуулалтанд хуучин уламжлалт дүрмүүдийг бичих параметрүүд болон файлтай ажиллах логикууд багтсан байдаг. Харин шинэ функцуудыг нь зөвхөн нэмэлт боломж байдлаар оруулсан нь аюулгүй байдлыг хавьгүй илүү хангасан аюулгүй галт хана бий болгож байгаа тэднийг хэт доогуур тавьсан санагддаг.

Энэ бүлэгт байгаа зааврууд нь "quick" болон төлөвт "keep state" тохируулгуудыг агуулсан дүрмүүдээс үндсэндээ бүрдсэн байгаа. Энэ бол хамааруулсан галт ханын дүрмүүдийг бичих үндсэн арга барил юм.

Хуучин уламжлалт дүрмүүдтэй ажиллах аргуудын талаар дэлгэрэнгүй тайлбарыг: http://www.munk.me.uk/ipf/ipf-howto.html ба http://coombs.anu.edu.au/~avalon/ip-filter.html хаягаар орж үзнэ үү.

IPF FAQ-г http://www.phildev.net/ipf/index.html хаягаар орж үзнэ үү.

Нээлттэй эхийн IPFilter програмын захидлын жагсаалтын архивыг http://marc.theaimsgroup.com/?l=ipfilter хаягаар орж үзнэ үү.

### 31.5.1. IPF-г идэвхжүүлэх

IPF нь FreeBSD үндсэн суулгацтай хамт, ажиллах үеийн ачаалах боломжтой модуль байдлаар ирдэг. rc.conf тохиргооны файл дотор ipfilter_enable="YES" илэрхийлэл байгаа үед систем IPF цөмийн модулийг динамикаар ачаална. Энэ ачаалах боломжтой модуль нь бүртгэх боломжтойгоор, анхдагч `default pass all` тохируулгын хамт бүтээгдсэн байдаг. Анхдагч дүрмийг `block all` болгохын тулд IPF-г цөмд эмхэтгэх шаардлага байхгүй. Зөвхөн дүрмүүдийнхээ төгсгөлд бүгдийг хаах дүрмийг бичиж өгөхөд хангалттай.

### 31.5.2. Цөмийн тохируулгууд

FreeBSD цөм уруу дараах боломжуудыг эмхэтгэн IPF-г идэвхжүүлэх албагүй боловч, суурь мэдлэг болгон энд үзүүллээ. IPF-г цөм уруу хөрвүүлэн эмхэтгэснээр ачаалах боломжтой модулийг хэрэглэх боломжгүй болдог.

Цөмийн тохиргоон дахь жишээ IPF илэрхийллүүд /usr/src/sys/conf/NOTES гэсэн цөмийн эх файлд байх ба доор сийрүүлбэл:

```
options IPFILTER
options IPFILTER_LOG
options IPFILTER_DEFAULT_BLOCK
```

`options IPFILTER` мөр нь "IPFILTER" галт ханыг идэвхжүүлнэ.

`options IPFILTER_LOG` мөр нь `log` гэсэн түлхүүр үг орсон дүрмүүдийн хувьд урсгалыг ipl пакет бүртгэх хуурамч-төхөөрөмж уруу бүртгэх боломжтой болгоно.

`options IPFILTER_DEFAULT_BLOCK` мөр нь галт ханын `pass` дүрмэнд тохироогүй пакетийг хаах анхдагч чанарыг зааж өгнө.

Эдгээр тохируулгууд нь зөвхөн тэдгээрийг тохируулан, тусгайлан цөм бүтээж суулгасны дараа идэвхждэг.

### 31.5.3. rc.conf тохируулгууд

IPF-г систем ачаалах үед идэвхтэй болгохын тулд /etc/rc.conf дотор дараах илэрхийллүүд

байх ёстой:

```
ipfilter_enable="YES"             # Start ipf firewall
ipfilter_rules="/etc/ipf.rules"   # loads rules definition text file
ipmon_enable="YES"                # Start IP monitor log
ipmon_flags="-Ds"                 # D = start as daemon
                                  # s = log to syslog
                                  # v = log tcp window, ack, seq
                                  # n = map IP & port to names
```

Хэрэв энэ галт ханын цаана хувийн IP хаяг хэрэглэдэг LAN байгаа бол NAT функцыг идэвхжүүлэхийн тулд дараах мөрүүдийг нэмэх хэрэгтэй:

```
gateway_enable="YES"              # Enable as LAN gateway
ipnat_enable="YES"                # Start ipnat function
ipnat_rules="/etc/ipnat.rules"    # rules definition file for ipnat
```

### 31.5.4. IPF

Таны бичсэн дүрмүүдийг ачаалахад ipf(8) тушаалыг хэрэглэнэ. Та өөрийн дүрмүүдээс бүтсэн файлыг үүсгээд, түүнийгээ дээрх тушаалын ашиглан галт ханын одоо ажиллаж байгаа дотоод дүрмүүдтэй сольж тавьна гэсэн үг юм:

```
# ipf -Fa -f /etc/ipf.rules
```

`-Fa` нь бүх дотоод дүрмүүдийн хүснэгтийг цэвэрлэ гэсэн үг.

`-f` нь ачаалах дүрмүүдээ энэ файлаас унш гэсэн үг.

Ийм байдлаар та өөрийн хүссэн дүрмүүдийн файлыг үүсгээд, дээрх IPF тушаалыг ажиллуулан системийг шинээр ачаалахгүйгээр ажиллаж байгаа галт ханын дүрмүүдийг шинээр өөрчлөх боломжтой болж байна. Дээрх аргаар галт ханын дүрмүүдийг хэдэн ч удаа сольж болох тул энэ арга нь шинэ дүрмүүдийг туршихад тохиромжтой арга юм.

Энэ тушаалтай ажиллах боломжтой бусад тугуудын талаар дэлгэрэнгүйг ipf(8) заавар хуудаснаас үзнэ үү.

ipf(8) тушаал дүрмүүдийн файлыг стандарт текст файл гэж тооцдог. Симбол орлуулалттай скрипт байдлаар бичигдсэн файлыг ойлгохгүй.

Гэвч скрипт симбол орлуулалтын хүчийг ашиглан IPF дүрмүүдийг бүтээх арга зам байгаа. Илүү дэлгэрэнгүй мэдээллийг Симбол орлуулалттай скриптийг үүсгэх нь хэсгээс үзнэ үү.

### 31.5.5. IPFSTAT

ipfstat(8)-н анхдагч чанар бол галт ханыг хамгийн сүүлд асааснаас хойших, эсвэл `ipf -Z` тушаалыг өгөн хуримтлуулагчийг хамгийн сүүлд тэглэснээс хойших галт ханаар орж байгаа

болон гарч байгаа пакетуудыг хэрэглэгчийн тодорхойлж өгсөн дүрмүүдээр шүүсэн үр дүнд бий болсон статистик тоог гаргаж ирэн, дэлгэцэнд харуулах юм.

Дэлгэрэнгүйг ipfstat(8) заавар хуудаснаас үзнэ үү.

ipfstat(8) тушаалын анхдагч үр дүн дараах байдалтай байна:

```
input packets: blocked 99286 passed 1255609 nomatch 14686 counted 0
 output packets: blocked 4200 passed 1284345 nomatch 14687 counted 0
 input packets logged: blocked 99286 passed 0
 output packets logged: blocked 0 passed 0
 packets logged: input 0 output 0
 log failures: input 3898 output 0
 fragment state(in): kept 0 lost 0
 fragment state(out): kept 0 lost 0
 packet state(in): kept 169364 lost 0
 packet state(out): kept 431395 lost 0
 ICMP replies: 0 TCP RSTs sent: 0
 Result cache hits(in): 1215208 (out): 1098963
 IN Pullups succeeded: 2 failed: 0
 OUT Pullups succeeded: 0 failed: 0
 Fastroute successes: 0 failures: 0
 TCP cksum fails(in): 0 (out): 0
 Packet log flags set: (0)
```

Дотогшоо урсгалын хувьд `-i`, гадагшаа урсгалын хувьд `-o` тохируулгыг өгөхөд кернелийн ашиглаж буй дүрмүүдийн жагсаалтыг гаргаж харуулна.

`ipfstat -in` нь дотогшоо урсгалын дотоод дүрмүүдийн хүснэгтийг дүрмүүдийн дугаарын хамт харуулна.

`ipfstat -on` нь гадагшаа урсгалын дотоод дүрмүүдийн хүснэгтийг дүрмүүдийн дугаарын хамт харуулна.

Үр дүн нь дараах байдалтай байна:

```
@1 pass out on xl0 from any to any
@2 block out on dc0 from any to any
@3 pass out quick on dc0 proto tcp/udp from any to any keep state
```

`ipfstat -ih` нь дотогшоо урсгалын дотоод дүрмүүдийн хүснэгтийг дүрмүүдийн хэдэн удаа тохирсон тооны хамт харуулна.

`ipfstat -oh` нь гадагшаа урсгалын дотоод дүрмүүдийн хүснэгтийг дүрмүүдийн хэдэн удаа тохирсон тооны хамт харуулна.

Үр дүн нь дараах байдалтай байна:

```
2451423 pass out on xl0 from any to any
354727 block out on dc0 from any to any
430918 pass out quick on dc0 proto tcp/udp from any to any keep state
```

`ipfstat` тушаалын хамгийн чухал функцуудын нэг бол, FreeBSD-н ажиллаж байгаа процессийн хүснэгтийг top(1) харуулдаг шиг төлвийн хүснэгтийг `-t` туг харуулдаг явдал юм. Таны галт хана гадны халдлагад өртөх үед энэ функц түүнийг илрүүлэх, шинжлэх, халдлагад оролцож буй пакетуудыг харах боломжийг олгоно. Нэмэлт дэд тугууд нь хяналт хийх эхлэл болон очих IP хаяг, порт, эсвэл протоколыг сонгох боломжийг олгодог. Дэлгэрэнгүйг ipfstat(8) заавар хуудаснаас үзнэ үү.

### 31.5.6. IPMON

`ipmon` тушаал зохистой ажиллахын тулд цөмийн `IPFILTER_LOG` тохируулга идэвхжсэн байх ёстой. Энэ тушаал хоёр өөр горимд ажиллах чадвартай. Төрөлх горим нь энэ тушаалыг тушаал мөрөн дээр `-D` туггүйгээр оруулахад ажиллах анхдагч горим юм.

Демон горим нь болж өнгөрсөн үйл явцын бүртгэлийг эргэж харахын тулд системийн бүртгэлийг тасралтгүй хөтлөн явуулахад тохиромжтой горим юм. FreeBSD болон IPFILTER энэ горимд ажиллахаар тохируулагдсан байдаг. FreeBSD нь системийн бүртгэлийг автоматаар тойруулах чадвартай. Тиймээс бүртгэлийн мэдээллийг syslogd(8) процесс уруу гаргах нь энгийн файл уруу гаргах анхдагч аргаас дээр байдаг. Анхдагч rc.conf файл дотор `ipmon_flags` илэрхийлэл `-Ds` тугуудыг хэрэглэдэг:

```
ipmon_flags="-Ds" # D = start as daemon
                  # s = log to syslog
                  # v = log tcp window, ack, seq
                  # n = map IP & port to names
```

Бүртгэл хөтлөн явуулахын давуу талыг дурдахад илүүц биз. Бүртгэлийн тусламжтай ямар пакетууд орхигдсон, тэдгээр пакетууд хаанаас ирсэн, хаашаа явж байсан зэрэг мэдээллийг эргэн харах боломжтой болдог. Энэ бүх мэдээлэл гадны халдлагыг мөрдөхөд чухал түлхэц болно.

Хэдийгээр бүртгэх боломжоор хангагдсан боловч, IPF дангаараа бүртгэлийг үүсгэж чадахгүй. Галт ханын администратор аль дүрмийн бүртгэлийг бичихийг шийдэн, тэдгээр дүрмүүдэд log түлхүүр үгийг нэмж өгнө. Ер нь, зөвхөн deny дүрмүүдийн бүртгэл бичигддэг.

Бүгдийг хориглосон анхдагч дүрмийг log түлхүүр үгийн хамт дүрмүүдийнхээ хамгийн төгсгөлд бичиж өгөх нь нилээд өргөн хэрэглэгддэг арга юм. Ингэснээр таны дүрмүүдийн алинтай ч тохироогүй пакетуудыг мэдэх боломжтой болно.

### 31.5.7. IPMON бүртгэл хөтлөлт

Syslogd нь бүртгэлийн мэдээллийг дотор нь ангилах өөрийн тусгай аргатай. "facility" ба "түвшин" гэсэн тусгай ангилалаар ялгадаг. `-Ds` горимон дахь IPMON нь анхдагчаар "facility"-аар `local0`-г хэрэглэдэг. Хэрэв хүсвэл доорх түвшнүүдийг ашиглан бүртгэгдсэн мэдээллийг

илүү ангилж болно:

```
LOG_INFO - нэвтрүүлэх, хаахаас үл хамааран "log" түлхүүрийг үйлдэл ашиглан пакетуудыг
бүртгэх.
LOG_NOTICE - нэвтэрсэн пакетуудыг бүртгэх.
LOG_WARNING - хаагдсан пакетуудыг бүртгэх.
LOG_ERR - бүртгэсэн пакетууд болон богино гэгдсэн пакетууд
```

IPFILTER-н бүх бүртгэлийн мэдээллийг /var/log/ipfilter.log файл дотор бичихийн тулд, файл эхлээд үүссэн байх хэрэгтэй. Үүний тулд дараах тушаалыг өгөх хэрэгтэй:

```
# touch /var/log/ipfilter.log
```

syslogd(8)-н функцуудыг /etc/syslog.conf файл доторх тодорхойлох илэрхийллүүдээр удирдаж болно. syslog.conf файл нь IPF мэт програмуудын үүсгэсэн системийн мэдэгдлүүдтэй syslog-г ажиллахад уян хатан болгодог.

Дараах илэрхийллүүдийг /etc/syslog.conf файл дотор нэмж бичнэ үү:

```
local0.* /var/log/ipfilter.log
```

`local0.*` нь бүх бүртгэгдсэн мэдэгдлүүдийг дурдсан файлд бичихийг хэлж өгч байна.

/etc/syslog.conf файлд хийсэн өөрчлөлтүүдийг идэвхжүүлэхийн тулд та системээ дахин ачаалах эсвэл `/etc/rc.d/syslogd reload` тушаалыг ашиглан syslogd(8) демонд /etc/syslog.conf файлыг дахин уншуулах хэрэгтэй.

Дээр шинээр үүсгэсэн бүртгэлийг тойруулахын тулд /etc/newsyslog.conf файл дотор өөрчлөлт оруулахаа мартуузай.

### 31.5.8. Бүртгэгдсэн мэдэгдлийн формат

`ipmon`-ы үүсгэсэн мэдэгдэл зайгаар тусгаарлагдсан өгөгдлийн талбаруудаас бүрдэнэ. Бүх мэдэгдэлд байдаг гол талбарууд гэвэл:

1. Пакетийг хүлээж авсан огноо.
2. Пакетийг хүлээж авсан цаг. Цаг, минут, секунд, бутархай секундэд (олон орны нарийвчлалтай) харгалзан HH:MM:SS.F форматтай байна.
3. Пакеттай ажилласан интерфэйсийн нэр, жишээлбэл dc0.
4. Дүрмийн бүлэг болон дүрмийн дугаар, жишээлбэл `@0:17`.

Эдгээрийг `ipfstat-in` тушаалын тусламжтай үзэж болно.

1. Үйлдэл: нэвтрүүлсэн бол p, хаасан бол b, богино пакет бол S, аль ч дүрмэнд тохироогүй бол n, бүртгэх дүрэм бол L. Эдгээр тугуудыг дараах эрэмбээр харуулна: S, p, b, n, L. Том P эсвэл B үсэг нь тухайн пакет ямар нэг дүрмээс биш, глобал тохиргооноос хамааран

бүртгэгдсэн болохыг заана.

2. Хаягууд. Үндсэндээ гурван талбар байна: эхлэл хаяг болон порт (таслалаар тусгаарлагдсан), → тэмдэг, ба очих хаяг болон порт. Жишээ нь `209.53.17.22,80 → 198.73.220.17,1722`.
3. `PR`-н дараа протоколын нэр болон дугаар, жишээлбэл `PR tcp`.
4. `len`-ы дараа толгойн урт болон пакетийн нийт урт, жишээлбэл `len 20 40`.

Хэрэв TCP пакет бол зураасаар эхэлж тугуудаар удаалсан нэмэлт талбар байна. Үсгүүд болон түүнд харгалзах тугуудын талаар ipf(5) заавар хуудаснаас үзнэ үү.

Хэрэв ICMP пакет бол, төгсгөлд нь хоёр талбар байна. Эхнийх нь үргэлж "ICMP" утгатай байна, дараагийнх нь налуу зураасаар тусгаарлагдсан ICMP мэдэгдэл болон дэд мэдэгдлийн төрөл, жишээлбэл портод хандаж чадсангүй гэсэн мэдэгдлийн хувьд ICMP 3/3 байна.

### 31.5.9. Симбол орлуулалттай скриптийг үүсгэх нь

Зарим туршлагатай IPF хэрэглэгчид дүрмүүдийг агуулсан файл үүсгээд түүнийгээ симбол орлуулалттай скрипт байдлаар ажиллуулах боломжтой болгон бичдэг. Үүний гол давуу тал нь зөвхөн симбол нэрд харгалзах утгыг өөрчлөх хэрэгтэй бөгөөд, скриптийг ажиллуулахад уг симбол орлуулалт орсон дүрэм бүр шинэ утгыг авах болно. Скриптийн хувьд, олон дахин хэрэглэгддэг утгуудыг бичихэд симбол орлуулалтыг ашиглаж, тэдгээрийг олон дүрмэнд орлуулж өгнө гэсэн үг юм. Дараах жишээн дээрээс харна уу.

Энд хэрэглэгдсэн скриптийн синтакс нь sh(1), csh(1), ба tcsh(1) бүрхүүл дээр ажиллах боломжтой.

Симбол орлуулалтын талбарууд нь урдаа долларын тэмдэгтэй байна: `$`.

Симбол талбарууд нь $ тэмдэг урдаа байхгүй.

Симбол талбарыг орлох утга нь давхар хашилтан(") дотор байрлана.

Дүрмийн файлаа дараах байдалтай үүсгэж эхлэх хэрэгтэй:

```
############## Start of IPF rules script ###########################

oif="dc0"            # name of the outbound interface
odns="192.0.2.11"    # ISP's DNS server IP address
myip="192.0.2.7"     # my static IP address from ISP
ks="keep state"
fks="flags S keep state"

# You can choose between building /etc/ipf.rules file
# from this script or running this script "as is".
#
# Uncomment only one line and comment out another.
#
# 1) This can be used for building /etc/ipf.rules:
#cat > /etc/ipf.rules << EOF
```

```
#
# 2) This can be used to run script "as is":
/sbin/ipf -Fa -f - << EOF

# Allow out access to my ISP's Domain name server.
pass out quick on $oif proto tcp from any to $odns port = 53 $fks
pass out quick on $oif proto udp from any to $odns port = 53 $ks

# Allow out non-secure standard www function
pass out quick on $oif proto tcp from $myip to any port = 80 $fks

# Allow out secure www function https over TLS SSL
pass out quick on $oif proto tcp from $myip to any port = 443 $fks
EOF
################### End of IPF rules script ##########################
```

Байх ёстой бүх зүйлс бүгд байна. Энэ жишээн дээр дүрмүүд чухал биш, харин симбол орлуулалт хэрхэн ажилладгыг харуулсан байна. Хэрэв дээрх жишээ /etc/ipf.rules.script нэртэй файл дотор байсан бол, эдгээр дүрмүүдийг дараах тушаалыг өгч дахин ачаалах боломжтой:

```
# sh /etc/ipf.rules.script
```

Суулгагдсан симболтой дүрмийн файлыг хэрэглэхэд нэг асуудал тулгардаг: IPF симбол орлуулалтыг ойлгохгүй, ийм скриптийг шууд уншиж чаддаггүй.

Дараах скриптийг хоёр аргын нэгээр хэрэглэж болно:

- `cat`-р эхэлсэн мөрийг ил гарга, харин `/sbin/ipf`-р эхэлсэн мөрүүдийг далдал. `ipfilter_enable="YES"`-г /etc/rc.conf файл дотор байрлуул, дараа нь өөрчлөлт бүрийн дараа скриптийг ажиллуулан /etc/ipf.rules файлыг үүсгэ эсвэл өөрчлөлт оруул.
- /etc/rc.conf файл дотор `ipfilter_enable="NO"` (энэ анхдагч утга) мөрийг нэмэн системийн эхлэл скриптэд IPFILTER-г идэвхгүй болго.

  Дээрхтэй адил скриптийг өөрийн /usr/local/etc/rc.d/ эхлэл хавтаст байрлуул. Энэ скрипт ipf.loadrules.sh ч юм уу ойлгомжтой нэртэй байх ёстой. .sh гэсэн өргөтгөлтэй байх ёстой.

  ```
  #!/bin/sh
  sh /etc/ipf.rules.script
  ```

  Энэ скриптийн эрхүүд эзэмшигч `root`-н хувьд унших, бичих, ажиллах эрхтэй байх ёстой.

  ```
  # chmod 700 /usr/local/etc/rc.d/ipf.loadrules.sh
  ```

Одоо систем ачаалсны дараа таны IPF дүрмүүд ачаалагдсан байх болно.

### 31.5.10. IPF Дүрмүүдийн олонлог

Пакет дотор агуулагдаж байгаа утгуудыг үндэслэн тухайн пакетийг нэвтрүүлэх болон хаахыг хэлж байгаа IPF дүрмүүдийг дүрмийн олонлог гэнэ. Хостуудын хоорондох хоёр чиглэлтэй пакет солилцоо нь сесс харилцааг бүрдүүлнэ. Галт ханын дүрмийн олонлог нь Интернэтээс ирж байгаа пакетуудыг болон систем буцааж тэдэнд хариу өгсөн пакетуудыг боловсруулдаг. Бүх TCP/IP үйлчилгээнүүд (жишээ нь: telnet, www, mail, г.м.) өөрийн протокол болон зөвшөөрөгдсөн (сонсож байгаа) портоороо тодорхойлогддог. Тухайн нэг үйлчилгээ рүү зорисон пакетууд нь тусгай зориулалтаар ашиглагддаггүй порт ашиглан эх хаягаас гарч очих хаягийн тухайн үйлчилгээний порт руу чиглэдэг. Дээрх бүх параметрууд (өөрөөр хэлбэл: портууд болон хаягууд) дээр үндэслэн нэвтрүүлэх болох хаах дүрмүүдийг зохиодог.

IPF-г анх зохиохдоо "сүүлд тохирсон дүрэм дийлнэ" логикийн дагуу бүтээсэн ба зөвхөн төлөвт-бус дүрмүүдийг ашигладаг байлаа. Цаг хугацаа өнгөрөхөд IPF илүү хөгжиж, "quick" тохируулга болон төлөвт "keep state" тохируулгуудыг агуулах болсон нь орчин үеийн хэрэгцээ шаардлагад илүү нийцэх болжээ.

Энэ бүлэгт байгаа зааврууд нь "quick" болон төлөвт "keep state" тохируулгуудыг агуулсан дүрмүүдээс үндсэндээ бүрдсэн байгаа. Энэ бол хамааруулсан галт ханын дүрмүүдийг бичих үндсэн арга барил юм.

Галт ханын дүрмүүдтэй ажиллахдаа *маш анхааралтай* байх хэрэгтэй. Зарим тохиргоо серверээс *бүх холбоог тань тасалж* мэднэ. Ийм аюулаас хол байхын тулд, галт ханын тохиргоог анхлан хийхдээ ssh зэрэг алсын хандалтаас илүүтэйгээр ойрын удирдлагыг сонгоорой.

### 31.5.11. Дүрмийн синтакс

Энд дурдах дүрмийн синтакс нь орчин үеийн төлөвт дүрмүүдийн хүрээнд, "сүүлд тохирсон дүрэм дийлнэ" логикоор ажиллахаар хялбаршуулан бичигдсэн байгаа. Хуучин уламжлалт дүрмүүдийн синтаксын бүрэн тайлбарыг ipf(8) заавар хуудаснаас үзнэ үү.

# гэсэн тэмдэгт тайлбарын эхлэлийг заах ба дүрэм бичсэн мөрийн төгсгөлд эсвэл өөрөө бүтэн мөр байдлаар байрлана. Хоосон мөрийг тооцохгүй.

Дүрмүүд нь түлхүүр үгүүдийг агуулна. Эдгээр түлхүүр үгүүд нь тухайн мөрөнд зүүнээс баруун тийш тодорхой дэс дараагаар бичигдэнэ. Түлхүүр үгүүд тодоор бичигдэх тул түүгээр нь ялгаж танина. Зарим түлхүүр үгүүд дэд-тохируулгатай байна. Тэр нь өөрөө түлхүүр үг байж болохоос гадна цааш дэд-тохируулгуудыг агуулсан байж болно. Доорх синтаксын үг бүр нь дор байрлах мөрүүдэд задаргааны хамт байгаа.

*ACTION IN-OUT OPTIONS SELECTION STATEFUL PROTO SRC_ADDR,DST_ADDR OBJECT PORT_NUM TCP_FLAG STATEFUL*

*ACTION* = block | pass

*IN-OUT* = in | out

*OPTIONS* = log | quick | on interface-name

*SELECTION* = proto value | source/destination IP | port = number | flags flag-value

*PROTO* = tcp/udp | udp | tcp | icmp

*SRC_ADD,DST_ADDR* = all | from object to object

*OBJECT* = IP address | any

*PORT_NUM* = port number

*TCP_FLAG* = S

*STATEFUL* = keep state

#### 31.5.11.1. ACTION

Тухайн дүрмэнд тохирч байгаа пакетийг хэрхэхийг action буюу үйлдэл зааж өгнө. Бүх дүрэм үйлдэлтэй *байх ёстой*. Дараах үйлдлүүдийг хэрэглэж болно:

`block` гэдэг нь пакеттай selection буюу сонголтын параметрүүд тохирч байвал тухайн пакетийг орхигдуулахыг зааж өгнө.

`pass` гэдэг нь пакеттай selection буюу сонголтын параметрүүд тохирч байвал тухайн пакетийг нэвтрүүлэхийг зааж өгнө.

#### 31.5.11.2. IN-OUT

Дүрэм нь орох болон гарах урсгалын алинд үйлчлэхийг заавал зааж өгөх ёстой. Энэ нь `in` эсвэл `out` түлхүүр үгийн аль нэг нь заавал бичигдсэн байх ёстой гэсэн үг юм. Үгүй бол синтаксын алдаа өгч, танигдахгүй.

`in` гэдэг нь Интернэт уруу харж байгаа интерфэйс дээр хүлээж авсан дотогшоо ирж байгаа пакетийн хувьд энэ дүрэм үйлчлэхийг зааж өгнө.

`out` гэдэг нь Интернэт уруу харж байгаа интерфэйс уруу чиглэсэн гадагшаа явж байгаа пакетийн хувьд энэ дүрэм үйлчлэхийг зааж өгнө.

#### 31.5.11.3. OPTIONS

Эдгээр options буюу тохируулгуудыг энд үзүүлсэн дэс дарааллын дагуу хэрэглэх ёстой.

`log` гэдэг нь пакеттай selection буюу сонголтын параметрүүд тохирч байвал пакетийн толгой ipl бүртгэл уруу (дор Бүртгэл Хөтлөх хэсэгт заасны дагуу) бичигдэхийг зааж өгнө.

`quick` гэдэг нь пакеттай selection буюу сонголтын параметрүүд тохирч байвал энэ дүрэм нь хамгийн сүүлийн дүрэм болохыг зааж өгнө. Ингэснээр "short-circuit" замыг тухайн пакетийн хувьд дараагийн дүрмүүдийг шалгахыг болиулна. Орчин үеийн дүрэмтэй ажиллах логикийн хувьд энэ тохируулгыг заавал хэрэглэнэ.

`on` гэдэг нь selection буюу сонголтын параметрүүдийн ажиллах интерфэйсийг зааж өгнө.

Интерфэйсүүдийн нэрийг ifconfig(8)-н тусламжтай харж болно. Энэ тохируулгыг хэрэглэснээр, тухайн дүрэм зөвхөн энэ интерфэйсээр зохих чиглэлд(in/out) явж байгаа пакетийн хувьд үйлчилнэ. Орчин үеийн дүрэмтэй ажиллах логикийн хувьд энэ тохируулгыг заавал хэрэглэнэ.

Пакетийг бүртгэхэд, түүний толгойг IPL пакет бүртгэх хуурамч-төхөөрөмж уруу бичнэ. `log` түлхүүр үгийн дараа шууд залгаад, дараах тодотгогчдыг(дараах дэс дарааллаар) хэрэглэж болно:

`body` гэдэг нь пакетийн толгойн дараа пакетийн агуулгын эхний 128 байтыг бүртгэхийг зааж өгнө.

`first` Хэрэв `log` түлхүүр үг `keep state` тохируулгын хамт хэрэглэгдсэн бол, түүний араас ирэх "keep state"-д тохирч байгаа бүх пакетийг биш зөвхөн энэ тохируулгыг идэвхжүүлсэн эхний пакетийг бүртгэхийн тулд энэ тохируулгыг хэрэглэнэ.

#### 31.5.11.4. SELECTION

Энэ бүлэгт танилцуулж байгаа түлхүүр үгүүд тухайн пакетийг дүрмэнд тохирсон эсэхийг тогтоохын тулд шалгадаг пакетийн онцлогийг тодорхойлоход хэрэглэгддэг. Мөн subject түлхүүр үг байх ба дэд-тохируулга түлхүүр үгийн аль нэгийг сонгон хэрэглэнэ. Дараах ерөнхий онцлогуудыг хэрэглэх боломжтой, гэхдээ доорх дэс дарааллаар хэрэглэх хэрэгтэй:

#### 31.5.11.5. PROTO

`proto` гэдэг нь subject түлхүүр үг бөгөөд өөрийн харгалзах дэд-тохируулгын хамт хэрэглэгдэх ёстой. Утга нь ямар протокол дээр ажиллахыг хэлж өгнө. Орчин үеийн дүрэмтэй ажиллах логикийн хувьд энэ тохируулгыг заавал хэрэглэнэ.

`tcp/udp | udp | tcp | icmp` эсвэл /etc/protocols файл дотор байгаа протоколуудыг хэрэглэж болно. Тусгай `tcp/udp` гэсэн түлхүүр үг TCP эсвэл UDP пакетийг сонгоход хэрэглэгддэг ба, давхар эсвэл төстэй дүрмүүдийг арилгах үүднээс нэмэгдсэн байгаа.

#### 31.5.11.6. SRC_ADDR/DST_ADDR

`all` гэсэн түлхүүр үг нь өөр ямар ч параметргүй "from any to any" гэдэгтэй адил юм.

`from src to dst`: `from` ба `to` гэсэн түлхүүр үгүүд IP хаягийг шүүхэд хэрэглэгдэнэ. Дүрэмд хэрэглэхдээ эхлэл ба очих параметрийг *хоёуланг* зааж өгөх ёстой. `any` гэсэн тусгай түлхүүр үгийн тусламжтай бүх IP хаягийг зөвшөөрч өгч болно. Хэрэглэх жишээ: `from any to any` эсвэл `from 0.0.0.0/0 to any` эсвэл `from any to 0.0.0.0/0` эсвэл `from 0.0.0.0 to any` эсвэл `from any to 0.0.0.0`.

Цэгээр тусгаарлагдсан тоо/баг хэлбэрээр хялбархан илэрхийлэх боломжгүй IP хаягуудыг хэрэглэх боломжгүй. net-mgmt/ipcalc порт ашиглан үүнийг хялбарчилж болох юм. Нэмэлт мэдээллийг дараах вэб хуудсаар орж үзнэ үү: http://jodies.de/ipcalc.

#### 31.5.11.7. PORT

Хэрэв эхлэл эсвэл очих порт, эсвэл хоёулангаар нь тохируулах бол энэ нь зөвхөн TCP ба UDP

пакетуудад хамаарна. Порт жишсэн дүрэм бичихдээ /etc/services файл доторх үйлчилгээний нэр эсвэл бүхэл тоон портын дугаарыг хэрэглэнэ. Портыг `from` обьекттой хамт хэрэглэх үед энэ нь эхлэл портын дугаарыг, `to` обьекттой хамт хэрэглэх үед энэ нь очих портын дугаарыг заана. Орчин үеийн дүрэмтэй ажиллах логикийн хувьд port тохиргоог `to` обьекттой заавал хамт хэрэглэнэ. Хэрэглэх жишээ: `from any to any port = 80`

Портын харьцуулалтыг төрөл бүрийн жиших операторуудыг ашиглан хэд хэдэн аргаар хийж болно. Портын зурвасыг бас зааж өгч болно.

port "=" | "!=" | "<" | ">" | "⇐" | ">=" | "eq" | "ne" | "lt" | "gt" | "le" | "ge".

Порт зурвасыг зааж өгөхдөө, port "<>" | "><" гэж хэрэглэнэ.

Орчин үеийн дүрэмтэй ажиллах логикийн хувьд эхлэл болон очих порт тохируулах параметрүүдийн дараа, дараах хоёр параметрийг заавал хэрэглэнэ.

#### 31.5.11.8. TCP_FLAG

Тугуудыг зөвхөн TCP шүүлтийн үед хэрэглэнэ. Үсгүүдээр нь TCP пакетийн толгойтой таарч байгаа эсэхийг шалгах боломжит тугуудыг үзүүлсэн байна.

Орчин үеийн дүрэмтэй ажиллах логик нь `flags S` параметрийг tcp сесс эхлүүлэх хүсэлтийг тэмдэглэхдээ хэрэглэдэг.

#### 31.5.11.9. STATEFUL

`keep state` гэдэг нь нэвтрүүлэх төрлийн дүрмийн хувьд сонгох параметрүүдтэй тохирсан ямар ч пакет төлөвт шүүх нэмэлт боломжийг идэвхжүүлэх ёстойг зааж өгнө.

Орчин үеийн дүрэмтэй ажиллах логикийн хувьд энэ тохируулгыг заавал хэрэглэнэ.

### 31.5.12. Төлөвт шүүлт

Хостуудын хоорондох хоёр чиглэлтэй пакет солилцоо сесс харилцаанаас бүрддэг. Төлөвт шүүлт нь урсгалыг сесс харилцаанаас бүрдэж буй хоёр чиглэлтэй пакет солилцоо гэж үздэг. keep-state-г идэвхжүүлсэн үед, keep-state нь хоёр чиглэлтэй сесс харилцааны үед солилцсон бүх пакетуудын хувьд дотоод дүрмүүдийг динамик байдлаар үүсгэдэг. Мөн энэ нь сесс эхлүүлсэн болон хүлээн авсан хостуудын хоорондох сесс харилцаа хоёр чиглэлтэй пакет солилцох процедуртай зохицож байгаа эсэхийг магадлах чадвартай байдаг. Сесс харилцааны загварт зохицоогүй пакетийг хуурамч гэж үзэн хүлээж авахгүй.

TCP эсвэл UDP сесстэй холбоотой ICMP пакетуудыг keep state нэвтрүүлнэ. Тэгэхээр, keep state дүрмээр зөвшөөрөгдсөн вэбээр хийх аялалын хариуд ICMP type 3 code 4 хариуг хүлээн авбал галт хана үүнийг автоматаар нэвтрүүлнэ гэсэн үг юм. Хэрэв IPF хүлээн авсан пакетийг идэвхтэй байгаа сессийн нэг хэсэг гэж баттай итгэж байвал, өөр протокол дээр байсан ч пакетийг нэвтрүүлнэ.

Үүний цаана юу болох вэ гэвэл:

Интернэт уруу холбогдсон интерфэйсээр гарч байгаа пакетуудыг хамгийн түрүүнд динамик төлвийн хүснэгтэнд шалгана. Хэрэв тухайн пакет ямар нэг идэвхтэй байгаа сесс харилцаанд оролцож байгаа зөвшөөрөгдсөн дараагийн пакет байх юм бол, галт ханаар нэвтрүүлэх ба динамик төлвийн хүснэгтэн дэх сесс харилцааны төлөв шинэчлэгдэнэ. Идэвхтэй сесс харилцаанд хамааралгүй пакетуудыг гадагшаа урсгалын дүрмээр шалгах болно.

Интернэт уруу холбогдсон интерфэйсээс ирж байгаа пакетуудыг хамгийн түрүүнд динамик төлвийн хүснэгтэнд шалгана. Хэрэв тухайн пакет ямар нэг идэвхтэй байгаа сесс харилцаанд оролцож байгаа зөвшөөрөгдсөн дараагийн пакет байх юм бол, галт ханаар нэвтрүүлэх ба динамик төлвийн хүснэгтэн дэх сесс харилцааны төлөв шинэчлэгдэнэ. Идэвхтэй сесс харилцаанд хамааралгүй пакетуудыг дотогшоо урсгалын дүрмээр шалгах болно.

Харилцаа дуусахад динамик төлвийн хүснэгтээс зохих бичлэг устгагдана.

Төлөвт шүүлтийн тусламжтайгаар та шинэ сесс зөвшөөрөх/хаах үйл ажиллагаан дээр төвлөрч ажиллаж чадна. Хэрэв шинэ сесс зөвшөөрөгдсөн бол түүний дараагийн бүх пакетуудыг автоматаар нэвтрүүлэх ба хуурамч пакетууд автоматаар буцаагдана. Хэрэв шинэ сесс хаагдсан бол түүний дараагийн ямар ч пакет нэвтэрч чадахгүй. Төлөвт шүүлт нь сүүлийн үеийн халдлагуудад ашиглагдаж байгаа аргуудын эсрэг хамгаалах чадвартай, техникийн хувьд өндөр түвшний шүүлт хийх чадвартай юм.

## 31.5.13. Хамааруулсан дүрмийн олонлогийн жишээ

Дараах дүрмийн олонлог нь аюулгүй байдлыг маш сайн хангасан, хамааруулсан галт ханын дүрмүүдийг хэрхэн бичих жишээ юм. Хамааруулсан галт хана нь зөвхөн `pass` дүрмүүдэд тохирсон үйлчилгээг зөвшөөрч, бусдыг бүгдийг хаана. Бусад машинуудыг хамгаалах ёстой галт хананууд буюу "сүлжээний галт хананууд" нь багаар бодоход хоёр интерфэйстэй байх ёстой бөгөөд ерөнхийдөө нэг талд (LAN) итгэж нөгөөд (Интернэт) итгэхгүй байхаар тохируулагдсан байдаг. Мөн ажиллаж байгаа системээ зөвхөн хамгаалахаар галт хана тохируулагдсан байж болох бөгөөд ийм галт ханыг "хостын галт хана" гэх бөгөөд энэ нь итгэлгүй сүлжээн дэх серверүүдийн хувьд ялангуяа тохиромжтой байдаг.

FreeBSD-г оролцуулаад бүх UNIX® төрлийн систем нь үйлдлийн систем дэх дотоод харилцаандаа lo0 интерфэйс болон `127.0.0.1` гэсэн IP хаягийг хэрэглэхээр бүтээгдсэн байдаг. Галт ханын дүрмүүд нь дотооддоо хэрэглэж байгаа эдгээр пакетуудыг чөлөөтэй нэвтрүүлэх дүрмүүдийг агуулсан байх ёстой.

Интернэттэй холбогдож байгаа интерфэйс дээр Интернэт уруу гарч байгаа болон Интернэтээс ирж байгаа хандалтыг удирдах болон хянах дүрмүүдийг байрлуулна. Энэ нь таны PPP tun0 интерфэйс эсвэл таны DSL эсвэл кабель модемд холбогдсон NIC байж болно.

Галт ханын цаана байгаа хувийн сүлжээнд нэг болон түүнээс дээш тооны NIC-ууд холбогдсон тохиолдолд, тэдгээр интерфэйсүүдийн хувьд тэдгээр LAN интерфэйсүүдийн нэгээс нөгөө рүү ба/эсвэл гадагш гарсан пакетуудыг чөлөөтэй нэвтрүүлэх дүрмийг агуулсан байх ёстой.

Дүрмүүд гурван үндсэн хэсэгт хуваагдан зохион байгуулагдсан байх ёстой: эхлээд итгэлтэй интерфэйсүүд, дараа нь нийтийн гадагшаа интерфэйс, төгсгөлд нь нийтийн итгэлтэй дотогшоо интерфэйс.

Нийтийн интерфэйс хэсэгт байгаа дүрмүүд тухайн интерфэйс болон чиглэлийн хувьд хамгийн олон тохиолддог дүрмүүд нь хамгийн түрүүнд, цөөн тохиолддог дүрмүүдээс өмнө байхаар, хаах болох бүртгэх дүрмүүд нь хамгийн сүүлд байхаар бичигдсэн байна.

Дараах жишээн дээрх Гадагшаа хэсэг нь Интернэт хандалтыг зөвшөөрч өгөх үйлчилгээг тодорхойлох сонголтын утгуудыг агуулсан зөвхөн `pass` дүрмүүдээс бүрдэж байна. Бүх дүрмүүд `quick`, `on`, `proto`, `port`, болон `keep state` тохируулгуудыг агуулсан байгаа. `proto tcp` дүрмүүд нь төлөвт байдлыг идэвхжүүлж байгаа пакетийг тодорхойлох байдлаар, сесс эхлүүлэх хүсэлтийг тодорхойлох зорилгоор `flag` тохируулгыг агуулсан байна.

Дотогшоо хэсэгт эхлээд хүсээгүй пакетуудыг хаах дүрмүүдийг бичсэн байна. Энэ нь хоёр өөр шалтгаантай. Эхнийх нь хортой пакетуудын зарим нь зөвшөөрсөн урсгалын хэсэг байж болох юм. `allow` дүрэмд таарч байгаа тэр хэсэг дээр тулгуурлан эдгээр пакетуудыг зөвшөөрөлгүйгээр хаах ёстой. Хоёр дахь шалтгаан нь тухайн хэсэгт байгаа хамгийн сүүлийн дүрмээр хааж бүртгэхийн оронд тэдгээр мэдэгдэж байгаа сонирхолгүй татгалзалтуудыг чимээгүйгээр хааж болох юм. Хэсэг бүр дэх сүүлийн дүрэм бүх пакетуудыг хааж бүртгэдэг бөгөөд таны систем рүү халдаж байгаа хүмүүсийг шүүхэд шаардагдах баримтыг бий болгоход ашиглагдаж болох юм.

Өөр нэг санаа тавин тэмдэглэн хэлэх зүйл бол хүсээгүй урсгалын хариуд ямар ч хариу явуулахгүй байх явдал юм. Буруу пакетуудыг зүгээр орхиж тэд алга болох ёстой юм. Ингэснээр халдлага явуулагч түүний явуулсан пакетууд таны системд хүрсэн эсэх талаар юу ч мэдэхгүй үлдэх болно. Таны системийн талаар хэдий хэр бага мэднэ, тэд ямар нэг муу зүйл хийж чадах хүртэл төдий чинээ урт хугацаа зарцуулна гэсэн үг юм. `log first` тохируулгыг агуулах дүрмүүд анхны удаа таарахад бүртгэж авдаг. Энэ тохируулга нь жишээ `nmap OS fingerprint` дүрэмд орсон байдаг. security/nmap хэрэгслийг халдагчид таны серверийн үйлдлийн системийг танихын тулд ихэвчлэн ашигладаг.

`log first` дүрмийн хувьд бүртгэлийн мэдээлэл бүртгэгдэх бүрт `ipfstat -hio` тушаалаар энэ дүрэм хэдэн удаа тохирсон байгааг шалгаж болно. Ихээхэн хэмжээний бүртгэлийн мэдээлэл нь таныг халдлагад өртөж байгааг ихэвчлэн илэрхийлдэг.

/etc/services файлыг ашиглан мэдэгдэхгүй портын дугаарыг хайж олж болох юм. Мөн http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers хаягаар тухайн порт ямар зориулалтаар ашиглагддагийг орж шалгаж болох юм.

Троянуудын хэрэглэдэг портын дугааруудыг http://www.sans.org/security-resources/idfaq/oddports.php хаягаар орж шалгаарай.

Дараах дүрмийн олонлог нь ажиллаж байгаа систем дээр шалгагдсан аюулгүй байдлыг бүрэн хангасан хамааруулсан галт ханын дүрмийн олонлог юм. Үүнийг өөрийн системд хялбарханаар тааруулж болох юм. Зөвшөөрөхийг хүсэхгүй байгаа үйлчилгээнүүдийн хувьд харгалзах `pass` дүрмийг далдлаарай.

Хүсээгүй мэдээллийг бүртгэхгүйн тулд дотогшоо хэсэгт `block` дүрэм нэмж бичээрэй.

Дүрэм бүрт байгаа dc0 гэсэн интерфэйсийн нэрийн оронд таны системийг Интернэт уруу холбож байгаа NIC картны интерфэйсийн нэрээр сольж тавиарай. Хэрэглэгчийн PPP-н хувьд, энэ нь tun0 байна.

Дараах илэрхийллүүдийг /etc/ipf.rules дотор бичих хэрэгтэй:

```
######################################################################
# No restrictions on Inside LAN Interface for private network
# Not needed unless you have LAN
######################################################################

#pass out quick on xl0 all
#pass in quick on xl0 all

######################################################################
# No restrictions on Loopback Interface
######################################################################
pass in quick on lo0 all
pass out quick on lo0 all

######################################################################
# Interface facing Public Internet (Outbound Section)
# Match session start requests originating from behind the
# firewall on the private network
# or from this gateway server destined for the public Internet.
######################################################################

# Allow out access to my ISP's Domain name server.
# xxx must be the IP address of your ISP's DNS.
# Dup these lines if your ISP has more than one DNS server
# Get the IP addresses from /etc/resolv.conf file
pass out quick on dc0 proto tcp from any to xxx port = 53 flags S keep state
pass out quick on dc0 proto udp from any to xxx port = 53 keep state

# Allow out access to my ISP's DHCP server for cable or DSL networks.
# This rule is not needed for 'user ppp' type connection to the
# public Internet, so you can delete this whole group.
# Use the following rule and check log for IP address.
# Then put IP address in commented out rule & delete first rule
pass out log quick on dc0 proto udp from any to any port = 67 keep state
#pass out quick on dc0 proto udp from any to z.z.z.z port = 67 keep state

# Allow out non-secure standard www function
pass out quick on dc0 proto tcp from any to any port = 80 flags S keep state

# Allow out secure www function https over TLS SSL
pass out quick on dc0 proto tcp from any to any port = 443 flags S keep state

# Allow out send & get email function
pass out quick on dc0 proto tcp from any to any port = 110 flags S keep state
pass out quick on dc0 proto tcp from any to any port = 25 flags S keep state

# Allow out Time
pass out quick on dc0 proto tcp from any to any port = 37 flags S keep state
```

```
# Allow out nntp news
pass out quick on dc0 proto tcp from any to any port = 119 flags S keep state

# Allow out gateway & LAN users' non-secure FTP ( both passive & active modes)
# This function uses the IPNAT built in FTP proxy function coded in
# the nat rules file to make this single rule function correctly.
# If you want to use the pkg_add command to install application packages
# on your gateway system you need this rule.
pass out quick on dc0 proto tcp from any to any port = 21 flags S keep state

# Allow out ssh/sftp/scp (telnet/rlogin/FTP replacements)
# This function is using SSH (secure shell)
pass out quick on dc0 proto tcp from any to any port = 22 flags S keep state

# Allow out insecure Telnet
pass out quick on dc0 proto tcp from any to any port = 23 flags S keep state

# Allow out FreeBSD CVSup
pass out quick on dc0 proto tcp from any to any port = 5999 flags S keep state

# Allow out ping to public Internet
pass out quick on dc0 proto icmp from any to any icmp-type 8 keep state

# Allow out whois from LAN to public Internet
pass out quick on dc0 proto tcp from any to any port = 43 flags S keep state

# Block and log only the first occurrence of everything
# else that's trying to get out.
# This rule implements the default block
block out log first quick on dc0 all

######################################################################
# Interface facing Public Internet (Inbound Section)
# Match packets originating from the public Internet
# destined for this gateway server or the private network.
######################################################################

# Block all inbound traffic from non-routable or reserved address spaces
block in quick on dc0 from 192.168.0.0/16 to any    #RFC 1918 private IP
block in quick on dc0 from 172.16.0.0/12 to any     #RFC 1918 private IP
block in quick on dc0 from 10.0.0.0/8 to any        #RFC 1918 private IP
block in quick on dc0 from 127.0.0.0/8 to any       #loopback
block in quick on dc0 from 0.0.0.0/8 to any         #loopback
block in quick on dc0 from 169.254.0.0/16 to any    #DHCP auto-config
block in quick on dc0 from 192.0.2.0/24 to any      #reserved for docs
block in quick on dc0 from 204.152.64.0/23 to any   #Sun cluster interconnect
block in quick on dc0 from 224.0.0.0/3 to any       #Class D & E multicast

##### Block a bunch of different nasty things. ############
# That I do not want to see in the log
```

```
# Block frags
block in quick on dc0 all with frags

# Block short tcp packets
block in quick on dc0 proto tcp all with short

# block source routed packets
block in quick on dc0 all with opt lsrr
block in quick on dc0 all with opt ssrr

# Block nmap OS fingerprint attempts
# Log first occurrence of these so I can get their IP address
block in log first quick on dc0 proto tcp from any to any flags FUP

# Block anything with special options
block in quick on dc0 all with ipopts

# Block public pings
block in quick on dc0 proto icmp all icmp-type 8

# Block ident
block in quick on dc0 proto tcp from any to any port = 113

# Block all Netbios service. 137=name, 138=datagram, 139=session
# Netbios is MS/Windows sharing services.
# Block MS/Windows hosts2 name server requests 81
block in log first quick on dc0 proto tcp/udp from any to any port = 137
block in log first quick on dc0 proto tcp/udp from any to any port = 138
block in log first quick on dc0 proto tcp/udp from any to any port = 139
block in log first quick on dc0 proto tcp/udp from any to any port = 81

# Allow traffic in from ISP's DHCP server. This rule must contain
# the IP address of your ISP's DHCP server as it's the only
# authorized source to send this packet type. Only necessary for
# cable or DSL configurations. This rule is not needed for
# 'user ppp' type connection to the public Internet.
# This is the same IP address you captured and
# used in the outbound section.
pass in quick on dc0 proto udp from z.z.z.z to any port = 68 keep state

# Allow in standard www function because I have apache server
pass in quick on dc0 proto tcp from any to any port = 80 flags S keep state

# Allow in non-secure Telnet session from public Internet
# labeled non-secure because ID/PW passed over public Internet as clear text.
# Delete this sample group if you do not have telnet server enabled.
#pass in quick on dc0 proto tcp from any to any port = 23 flags S keep state

# Allow in secure FTP, Telnet, and SCP from public Internet
# This function is using SSH (secure shell)
```

```
pass in quick on dc0 proto tcp from any to any port = 22 flags S keep state

# Block and log only first occurrence of all remaining traffic
# coming into the firewall. The logging of only the first
# occurrence avoids filling up disk with Denial of Service logs.
# This rule implements the default block.
block in log first quick on dc0 all
#################### End of rules file ########################################
```

### 31.5.14. NAT

NAT нь *Network Address Translation* буюу Сүлжээний хаягийн Хөрвүүлэлтийн товчлол юм. Linux®-н талаар ойлголттой хүмүүсийн хувьд, энэ ойлголтыг IP маскарад гэж нэрлэдэг; NAT ба IP маскарад нь нэг зүйл юм. IPF NAT-н бидэнд олгож байгаа олон зүйлүүдийн нэг бол галт ханын цаана байгаа Local Area Network буюу Ойрын Зайн Сүлжээ(LAN)-н хувьд ISP-с оноож өгсөн ганц IP хаягийг Интернэтэд хуваан хэрэглэх юм.

Ингэх ямар шаардлага байнаа гэж та гайхан асуух байх. ISP-ууд өөрийн ашгийн-бус хэрэглэгчиддээ ихэвчлэн динамик IP хаяг оноодог. Динамик гэдэг нь таныг ISP руу залган нэвтрэн орох болгонд, кабель эсвэл DSL модемтой хэрэглэгчдийн хувьд модемоо асааж унтраах болгонд өөр өөр IP хаяг онооно гэсэн үг юм. Таны системийг Интернэтэд танихад энэ динамик IP хаягийг ашигладаг.

Та гэртээ таван PC-тэй бөгөөд бүгд Интернэт уруу гардаг байх хэрэгтэй гэж бодъё. Тэгвэл та PC тус бүрт тусад нь эрх худалдан авч, таван утасны үзүүртэй байх хэрэгтэй болно.

NAT-н тусламжтай ISP-гаас зөвхөн ганцхан эрх шаардлагатай. Бусад дөрвөн PC-гээ switch буюу шилжүүлэгч уруу холбож, харин switch-ээ таны LAN-нд гарц байдлаар ажиллах FreeBSD системийн NIC руу залгана. NAT нь LAN-д байгаа бүх PC-ны хувьд хувийн IP хаягийг ганцхан гадаад IP хаяг уруу автоматаар хөрвүүлэх болно. NAT нь эргэж ирж байгаа пакетуудын хувьд эсрэг хөрвүүлэлтийг мөн хийнэ.

NAT хийгдсэн хувийн LAN-уудад зориулж бүлэг IP хаягийг тусгайлан гаргасан байдаг. RFC 1918 стандартад зааснаар бол, дараах бүлэг IP-г хувийн сүлжээндээ ашиглах боломжтой, эдгээр IP хэзээ ч гадаад Интернэт уруу гарахгүй болно:

| | | |
|---|---|---|
| Эхлэх IP `10.0.0.0` | - | Төгсөх IP `10.255.255.255` |
| Эхлэх IP `172.16.0.0` | - | Төгсөх IP `172.31.255.255` |
| Эхлэх IP `192.168.0.0` | - | Төгсөх IP `192.168.255.255` |

### 31.5.15. IPNAT

NAT дүрмүүдийг `ipnat` тушаалын тусламжтай ачаална. Ихэвчлэн NAT дүрмүүд /etc/ipnat.rules файл дотор байрлана. Дэлгэрэнгүйг ipnat(8) хэсгээс үзнэ үү.

NAT ажиллаж эхэлсний дараа NAT дүрмүүдэд өөрчлөлт оруулах шаардлагатай бол NAT дүрмүүд байгаа файл дотор өөрчлөлтийг хийсний дараа, одоо хэрэглэгдэж байгаа NAT дүрмүүдийг устгаж, хөрвүүлэгч хүснэгтийг цэвэрлэхийн тулд `ipnat` тушаалыг `-CF` тугийн

хамт ажиллуулах хэрэгтэй.

Харин NAT дүрмүүдийг дахин ачаалахдаа тушаалыг дараах байдалтай өгөх хэрэгтэй:

```
# ipnat -CF -f /etc/ipnat.rules
```

NAT-н талаар зарим статистикийг харъя гэвэл дараах тушаалыг ашиглана:

```
# ipnat -s
```

NAT хүснэгтийн одоо ашиглаж байгаа оноолтын жагсаалтыг харахын тулд дараах тушаалыг ашиглана:

```
# ipnat -l
```

Вербос буюу хэр зэрэг харуулах горимыг нээхийн тулд, дүрэмтэй ажиллах болон идэвхтэй байгаа дүрмүүдийн хүснэгтийг харахын тулд:

```
# ipnat -v
```

### 31.5.16. IPNAT Дүрмүүд

NAT дүрмүүд нь маш уян хатан бөгөөд хэрэглэгчдийн хэрэгцээг хангах олон зүйлүүдийг хийж чадна.

Энд үзүүлсэн дүрмийн синтаксыг ашгийн-бус орчинд ихэвчлэн хэрэглэгддэг дүрмүүдэд зориулан хялбаршуулсан байгаа. Дүрмийн синтаксын бүрэн тайлбарыг ipnat(5) заавар хуудаснаас үзнэ үү.

NAT дүрмийн синтакс дараах байдалтай байна:

```
map IF LAN_IP_RANGE -> PUBLIC_ADDRESS
```

Дүрэм нь `map` гэсэн түлхүүр үгээр эхэлнэ.

*IF*-г гадаад интерфэйсээр сольж тавьна.

*LAN_IP_RANGE* нь танай дотоод хэрэглэгчийн хэрэглэж буй IP хаяглалтыг заана, ихэвчлэн `192.168.1.0/24` гэсэн маягтай байна.

*PUBLIC_ADDRESS* нь гадаад IP байж болно эсвэл *IF*-д оноосон IP хаягийг хэрэглэхийг заасан `0/32` гэсэн тусгай түлхүүр үг байж болно.

### 31.5.17. NAT хэрхэн ажилладаг вэ

Гадаад очих хаягтай пакет галт хана дээр LAN-с хүрэлцэн ирнэ. Эхлээд гадагшаа шүүлтийн дүрмүүдээр гарна, дараа нь NAT-н ээлж ирэх ба өөрийн дүрмүүдийг дээрээс доош шалгаж эхэлнэ. Хамгийн эхэнд тохирсон нь дийлнэ. NAT өөрийн дүрэм бүрийг пакетийн интерфэйсийн нэр болон эхлэл хаягаар тулгаж шалгана. Пакетийн интерфэйсийн нэр NAT дүрэмтэй тохирвол пакетийн эхлэл IP хаяг (өөрөөр хэлбэл хувийн LAN IP хаяг) NAT дүрмийн сумны зүүн талд зааж өгсөн IP хаягийн зурвас дотор байгаа эсэхийг шалгана. Хэрэв энэ тохирвол пакетийн эхлэл хаягийг `0/32` түлхүүр үгийн тусламжтай олж авсан гадаад IP хаягаар сольж бичнэ. NAT өөрийн дотоод NAT хүснэгтэнд бичлэг нэмэх ба энэ нь пакет Интернэтээс буцаж ирэхэд түүнийг буцаан хувийн IP хаяг уруу нь хөрвүүлэн, цааш шүүлтийн дүрмүүдээр оруулах боломжийг олгоно.

### 31.5.18. IPNAT-г идэвхжүүлэх

IPNAT-г идэвхжүүлэхийн тулд эдгээр илэрхийллүүдийг /etc/rc.conf дотор нэмж бичнэ.

Өөрийн машиныг интерфэйсүүдийн хооронд пакетуудыг чиглүүлдэг болгохын тулд:

```
gateway_enable="YES"
```

Систем ачаалахад IPNAT-г автоматаар ачаалдаг болгохын тулд:

```
ipnat_enable="YES"
```

IPNAT-н дүрмүүдийг хаанаас ачаалахыг зааж өгөхдөө:

```
ipnat_rules="/etc/ipnat.rules"
```

### 31.5.19. Маш том LAN-д зориулсан NAT

LAN-даа олон тооны PC-тэй сүлжээний хувьд эсвэл нэгээс олон LAN-тай сүлжээний хувьд, энэ олон хувийн IP хаягуудыг нэг гадаад IP хаяг уруу нийлүүлэх үйл явцад NAT хийгдсэн олон LAN PC дээр ижил портын дугаар олон дахин хэрэглэгдсэнээс мөргөлдөөн үүсэх гэх мэт нөөцтэй холбоотой асуудал гардаг. Нөөцтэй холбоотой энэ асуудлаас гарахын тулд дараах хоёр арга зам байдаг.

#### 31.5.19.1. Хэрэглэх портуудыг оноох

Энгийн NAT дүрэм дараах байдалтай байна:

```
map dc0 192.168.1.0/24 -> 0/32
```

Дээрх дүрмэнд пакет IPNAT-р дайрч өнгөрөхөд пакетийн эхлэл порт өөрчлөгдөхгүй. `portmap` гэсэн түлхүүр үгийг нэмсэнээр IPNAT-ийг заасан зурвас дахь зөвхөн эхлэл портуудыг

ашиглахаар зааж өгнө. Жишээ нь, дараах дүрэм IPNAT-г эхлэл порт хаягийг тухайн зурвас дотор байхаар өөрчлөхийг зааж өгч байна.

```
map dc0 192.168.1.0/24 -> 0/32 portmap tcp/udp 20000:60000
```

Дээр нь бид `auto` түлхүүр үгийн тусламжтай аль портуудыг ашиглах боломжтой байгааг өөрөө тодорхойлохыг зааж өгч болно:

```
map dc0 192.168.1.0/24 -> 0/32 portmap tcp/udp auto
```

#### 31.5.19.2. Гадаад хаягийн цөөрмийг хэрэглэх

Маш том LAN-уудын хувьд дэндүү олон LAN хаягуудыг нэг гадаад хаягт оноох нь боломжгүй болох үе ирдэг. Хэрэв бүлэг гадаад IP сул байгаа бол, эдгээр IP хаягуудыг "цөөрөм" байдлаар ашиглаж болох ба, IPNAT эдгээрээс нэгийг сонгон авч гадагшаа явж байгаа пакетийн хаягт оноох байдлаар хэрэглэх болно.

Жишээ нь, доор үзүүлсэн шиг бүх пакетуудыг ганц гадаад IP-д оноохын оронд:

```
map dc0 192.168.1.0/24 -> 204.134.75.1
```

гадаад IP хаягийн зурвасыг сүлжээний хуваалтын хамт зааж өгч болно:

```
map dc0 192.168.1.0/24 -> 204.134.75.0/255.255.255.0
```

эсвэл CIDR тэмдэглэгээг хэрэглэж болно:

```
map dc0 192.168.1.0/24 -> 204.134.75.0/24
```

### 31.5.20. Портын дахин чиглүүлэлт

LAN дотор вэб сервер, цахим шуудангийн сервер, өгөгдлийн сангийн сервер болон DNS серверийг өөр өөр PC дээр тараан ажиллуулах нь түгээмэл байдаг. Энэ тохиолдолд эдгээр серверээс гарч байгаа урсгал мөн NAT хийгдсэн байх ёстой. Гэхдээ гаднаас ирж буй урсгалыг зөв LAN PC уруу дахин чиглүүлэх арга зам байх хэрэгтэй болно. Энэ асуудлыг шийдэхийн тулд IPNAT нь дахин чиглүүлэлт хийх NAT нэмэлт боломжийг олгодог. Вэб сервер `10.0.10.25` гэсэн LAN хаягтай бөгөөд `20.20.20.5` гэсэн ганц гадаад IP-тай байлаа гэж бодъё. Тэгвэл дүрмийг дараах байдалтай:

```
rdr dc0 20.20.20.5/32 port 80 -> 10.0.10.25 port 80
```

эсвэл:

```
rdr dc0 0.0.0.0/0 port 80 -> 10.0.10.25 port 80
```

эсвэл гаднаас DNS хүсэлтүүд хүлээн авдаг `10.0.10.33` гэсэн хаягтай LAN DNS Серверийн хувьд:

```
rdr dc0 20.20.20.5/32 port 53 -> 10.0.10.33 port 53 udp
```

гэж бичих байсан.

## 31.5.21. FTP ба NAT

FTP-г Интернэт одоогийнх шиг байхаас өмнөх үе, их сургуулиуд түрээсийн шугамаар хоорондоо холбогдож, судлаач эрдэмтэд хоорондоо файл солилцохын тулд FTP-г ашигладаг байх үес үлдсэн үлэг гүрвэл гэж хэлж болох юм. Тэр үед өгөгдлийн аюулгүй байдлын талаар огт анхаардаггүй байлаа. Цаг хугацаа өнгөрөхөд FTP протоколыг шинээр гарч ирж байгаа Интернэтийн гол нуруу сүлжээнд хэрэглэх болсон ба түүний хэрэглэгчийн нэр, нууц үгийг цэвэр текст хэлбэрээр дамжуулдаг байдал нь хэзээ ч өөрчлөгдөөгүй бөгөөд орчин үеийн аюулгүй байдлын шаардлагад нийцэхгүй болсон билээ. FTP нь active буюу идэвхтэй, passive буюу идэвхгүй гэсэн хоёр горимд ажилладаг. Өгөгдлийн сувгийг хэрхэн ашиглаж байгаа дээр гол ялгаа нь гардаг. Өгөгдлийн сувгийг эхэлж ftp сесс хүсэгч нь авдаг тул идэвхгүй горимд ажиллах нь аюулгүй байдлыг илүүтэйгээр хангана. FTP-н талаар илүү сайн тайлбарыг болон түүний горимуудын талаар http://www.slacksite.com/other/ftp.html хаягаар үзнэ үү.

### 31.5.21.1. IPNAT Дүрмүүд

IPNAT нь дотроо NAT оноолт дүрэмд тодорхойлж өгөх боломжтой тусгай FTP прокси тохируулгыг агуулсан байдаг. Энэ нь идэвхтэй болон идэвхгүй FTP сесс эхлүүлэх хүсэлтэд оролцож байгаа бүх гадагшаа чиглэлтэй FTP пакетийг хянаж чадна. Мөн өгөгдлийн сувагт үнэхээр хэрэглэгдэж байгаа порт дугаарыг агуулсан түр зуурын шүүлтийн дүрмүүдийг динамикаар үүсгэж чадна. Ийм байдлаар FTP-с болж үүсдэг дээд хэсгийн портуудыг өргөн зурвасаар нээх эрсдэлээс галт ханыг хамгаалж байгаа юм.

Доорх дүрэм нь дотоод LAN-н бүх урсгалыг зохицуулна:

```
map dc0 10.0.10.0/29 -> 0/32 proxy port 21 ftp/tcp
```

Доорх дүрэм гарцаас ирж буй FTP урсгалыг зохицуулна:

```
map dc0 0.0.0.0/0 -> 0/32 proxy port 21 ftp/tcp
```

Доорх дүрэм дотоод LAN-с ирж буй бүх FTP-н биш урсгалыг зохицуулна:

```
map dc0 10.0.10.0/29 -> 0/32
```

FTP оноолтын дүрэм нь бидний ердийн оноолтын дүрмүүдийн өмнө бичигдэнэ. Бүх

пакетийг хамгийн дээр бичигдсэн дүрмээс эхлэн шалгана. Интерфэйсийн нэр тохирвол дотоод LAN эхлэл IP хаяг, дараа нь FTP пакет эсэхийг шалгана. Хэрэв бүгд тохирвол, тусгай FTP прокси эдгээр FTP сесс пакетуудыг NAT хийхээс гадна гадагш нь болон дотогш нь нэвтрүүлэх түр зуурын шүүлтийн дүрмийг үүсгэнэ. FTP-н биш бусад бүх LAN пакетууд эхний дүрмэнд тохирохгүй тул гуравдугаар дүрэм уруу шилжин дахин шалгагдана. Интерфэйс болон эхлэл IP тохирох тул NAT хийгдэнэ.

#### 31.5.21.2. IPNAT FTP Шүүлтийн Дүрмүүд

NAT FTP прокси ашиглаж байгаа тохиолдолд FTP-н хувьд ганцхан шүүлтийн дүрэм хэрэгтэй.

FTP Прокси байхгүй бол дараах гурван дүрмийг хэрэглэнэ:

```
# Allow out LAN PC client FTP to public Internet
# Active and passive modes
pass out quick on rl0 proto tcp from any to any port = 21 flags S keep state

# Allow out passive mode data channel high order port numbers
pass out quick on rl0 proto tcp from any to any port > 1024 flags S keep state

# Active mode let data channel in from FTP server
pass in quick on rl0 proto tcp from any to any port = 20 flags S keep state
```

## 31.6. IPFW

IPFIREWALL (IPFW) нь FreeBSD-ийн хандиваар FreeBSD-ийн сайн дурын гишүүдийн бүтээсэн, тэдний эрх мэдэлд байдаг галт ханын програм юм. Энэ нь хуучин уламжлалт төлөвт дүрмүүдийг хэрэглэдэг бөгөөд Simple Stateful logic буюу Хялбар Төлөвт логикийг бий болгохын тулд уламжлалт дүрэм бичих техникийг хэрэглэдэг.

Стандарт FreeBSD суулгац дахь IPFW-н хялбар дүрмийн олонлог (/etc/rc.firewall болон /etc/rc.firewall6 файл дотор байрлана) нь нилээд хялбар бөгөөд өөрт тохируулан засварласны дараа хэрэглэхээр бодолцон бичигдсэн байдаг. Жишээн дээр ихэнх суулгацад тохиромжтой төлөвт шүүлтийг хэрэглээгүй байгаа. Тиймээс энэ хэсэгт энэ жишээг хэрэглэхгүй болно.

IPFW-н төлөвт дүрмийн синтакс нь галт хана суулгах анхан шатны мэдлэгээс хол давсан техникийн хувьд ярвигтай сонголтын боломжуудаар хүч нэмсэн байдаг. IPFW нь мэргэжлийн түвшний хэрэглэгчид эсвэл өндөр түвшний пакет сонголт шаардлагатай байгаа техникийн өндөр түвшний компьютер сонирхогчид зориулагдсан юм. IPFW-н дүрмүүдийн хүчийг мэдрэхийн өмнө протоколууд өөрийн тусгай пакетийн толгойн мэдээллийг хэрхэн үүсгэдэг болон хэрэглэдэг талаар нилээд дэлгэрэнгүй мэдлэгийг олж авсан байх хэрэгтэй. Тийм түвшний тайлбарыг энд өгөх нь номын энэ бүлгийн мэдлээс халих тул энд оруулах боломжгүй юм.

IPFW нь долоон хэсгээс бүрдэнэ, гол хэсэг болох цөмийн галт ханын шүүлтийн дүрмийг боловсруулагч болон түүний бусад хэсэг болох пакет данслах боломж, бүртгэх боломж, NAT боломжийг идэвхжүүлэх `divert` дүрэм, болон өндөр түвшний тусгай зориулалттай боломжууд, dummynet трафик хязгаарлагч боломжууд, `fwd дүрэм` дамжуулах боломж, гүүр

боломжууд, болон ipstealth боломжуудаас бүрдэнэ. IPFW нь IPv4 болон IPv6-г дэмждэг.

### 31.6.1. IPFW-г идэвхжүүлэх

IPFW нь FreeBSD үндсэн суулгацтай хамт, ажиллах үеийн ачаалах боломжтой модуль байдлаар ирдэг. rc.conf тохиргооны файл дотор `firewall_enable="YES"` илэрхийлэл байгаа үед систем IPFW цөмийн модулийг динамикаар ачаална. IPFW-г цөмд эмхэтгэх шаардлага байхгүй.

rc.conf файл дотор `firewall_enable="YES"` илэрхийллийг нэмээд системийг дахин асаасны дараа ачаалах үйл явцын нэг хэсэг болж дараах мессеж дэлгэцэн дээр гарах болно:

```
ipfw2 initialized, divert disabled, rule-based forwarding disabled, default to deny,
logging disabled
```

Ачаалах боломжтой модульд бүртгэх боломжийг эмхэтгээгүй байгаа. Бүртгэлийг идэвхжүүлж вербос бүртгэлийн хязгаарыг тогтоохын тулд /etc/sysctl.conf файл дотор тохируулж болох тохиргоо бий. Эдгээр илэрхийллүүдийг нэмсэнээр бүртгэлийн систем дараагийн удаа ачаалахад идэвхжинэ:

```
net.inet.ip.fw.verbose=1
net.inet.ip.fw.verbose_limit=5
```

### 31.6.2. Цөмийн тохируулгууд

FreeBSD цөм уруу дараах боломжуудыг эмхэтгэн IPFW-г идэвхжүүлэх албагүй болно. Суурь мэдлэг болгон энд үзүүллээ.

```
options    IPFIREWALL
```

Энэ тохируулга IPFW-г цөмийн нэг хэсэг болгон идэвхжүүлнэ

```
options    IPFIREWALL_VERBOSE
```

Энэ тохируулга `log` гэсэн түлхүүр үг орсон дүрмийн хувьд IPFW-р дайран өнгөрөх пакетуудыг бүртгэх боломжтой болгоно.

```
options    IPFIREWALL_VERBOSE_LIMIT=5
```

Энэ тохируулга syslogd(8)-р нэгэн зэрэг бүртгэгдэж буй пакетийн тоог хязгаарлана. Галт ханын үйлдлүүдийг бүртгэхийг хүсэж байгаа дайсагнасан орчнуудад энэ тохируулгыг хэрэглэж болно. Энэ тохируулга нь syslog-г живүүлэх замаар явагдах үйлчилгээг зогсоох халдлагыг хааж өгөх болно.

```
options    IPFIREWALL_DEFAULT_TO_ACCEPT
```

Энэ тохируулга нь анхдагчаар галт ханыг дамжин өнгөрч байгаа бүх зүйлийг нэвтрүүлэхийг зөвшөөрөх бөгөөд энэ нь галт ханыг анх удаа тохируулж байгаа үед илүү тохиромжтой.

```
options    IPDIVERT
```

Энэ тохируулга NAT функцыг идэвхжүүлнэ.

Хэрэв та IPFIREWALL_DEFAULT_TO_ACCEPT-г оруулаагүй эсвэл ирж байгаа пакетуудыг нэвтрүүлэх дүрмүүд бичээгүй бол ирж байгаа болон явж байгаа бүх пакетуудыг галт хана хаах болно.

### 31.6.3. /etc/rc.conf Тохируулгууд

Галт ханыг идэвхжүүлэхийн тулд:

```
firewall_enable="YES"
```

FreeBSD-тэй хамт ирдэг анхдагч галт ханын төрлүүдээс нэгийг сонгохын тулд, /etc/rc.firewall файлыг уншсаны дараа нэгийг сонгоод, түүнийгээ дараах илэрхийлэлд бичиж өгнө:

```
firewall_type="open"
```

Боломжит утгууд нь:

- `open` - бүх урсгалыг нэвтрүүлнэ.
- `client` - зөвхөн энэ машиныг хамгаална.
- `simple` - бүхэл бүтэн сүлжээг хамгаална.
- `closed` - loopback интерфэйсээс бусад IP урсгалыг боломжгүй болгоно.
- `UNKNOWN` - галт ханын дүрмүүдийг ачаалах боломжгүй болгоно.
- filename - галт ханын дүрмүүдийг агуулсан файлын бүрэн зам.

ipfw галт хана уруу тусгайлан бэлдсэн дүрмүүдийг хоёр аргаар ачаалж болно. Нэг нь, `firewall_type` хувьсагчийн утганд ipfw(8)-д зориулсан ямар ч тушаал мөрийн тохируулгагүйгээр бичигдсэн *галт ханын дүрмүүд*-г агуулсан файлын бүрэн замыг өгөх. Дараах нь орж байгаа болон гарч байгаа урсгалыг хаах дүрмийн энгийн жишээ юм:

```
add deny in
add deny out
```

Нөгөө нь, систем ачаалах үед ажиллах `ipfw` тушаалуудыг агуулсан ажиллах боломжтой

скриптийн бүрэн замыг `firewall_script` хувьсагчид оноох юм. Дээр үзүүлсэн дүрмүүдийн файлтай дүйх дүрмүүдийн скрипт дараах байдалтай байна:

```
#!/bin/sh

ipfw -q flush

ipfw add deny in
ipfw add deny out
```

Хэрэв `firewall_type` нь `client` эсвэл `simple` уттыг авсан бол, /etc/rc.firewall файл доторх анхдагч дүрмүүдийг тухайн машинд тохируулан өөрчлөх хэрэгтэй. Мөн энэ бүлэгт хэрэглэж байгаа жишээнүүдийн хувьд `firewall_script`-н утга /etc/ipfw.rules гэж үзэж байгаа болно.

Бүртгэлийг идэвхжүүлэхийн тулд:

```
firewall_logging="YES"
```

`firewall_logging` хувьсагчийн хийх ганц зүйл гэвэл `net.inet.ip.fw.verbose` sysctl хувьсагчийн уттыг `1` болгох юм (IPFW-г идэвхжүүлэх хэсгийг үзнэ үү). rc.conf дотор бүртгэлийг хязгаарлах хувьсагч байхгүй, харин үүний тулд sysctl хувьсагчаар дамжуулан хийж болно. /etc/sysctl.conf файл дотор эсвэл гараараа утгыг оноож өгч болно:

```
net.inet.ip.fw.verbose_limit=5
```

Хэрэв таны машин гарц байдлаар ажиллаж байгаа бол, жишээ нь natd(8)-н тусламжтай Сүлжээний хаягийн Хөрвүүлэлт (NAT) хийж байгаа бол, /etc/rc.conf файл доторх шаардлагатай тохируулгуудын мэдээллийг Network Address Translation буюу Сүлжээний Хаягийн Хөрвүүлэлт хэсэг уруу хандана уу.

### 31.6.4. IPFW Тушаал

Галт ханыг ажиллаж байх явцад түүний идэвхтэй байгаа дотоод дүрмүүдэд шинэ дүрэм нэмэх, дүрэм хасах зэрэг өөрчлөлтүүдийг гараар хийх гол механизм бол `ipfw` тушаал юм. Энэ аргыг хэрэглэхэд тулгардаг нэг асуудал бол нэгэнт системийг унтраасан эсвэл зогсоосон бол нэмсэн эсвэл хассан эсвэл өөрчилсөн бүх дүрмүүд алга болно. Бүх дүрмүүдээ нэг файлд бичээд систем ачаалах үед энэ файлыг ашиглан дүрмүүдийг ачаалах, эсвэл одоо ажиллаж байгаа галт ханын дүрмүүдийг файл дотор хийсэн өөрчлөлтүүдээр бүхлээр нь сольж тавих нь энд хэрэглэж байгаа, та бүхэнд зөвлөх арга барил юм.

Удирдлагын дэлгэцэн дээр ажиллаж байгаа галт ханын дүрмүүдийг харуулахад `ipfw` тушаалыг одоо хэр нь хэрэглэсээр байна. IPFW-н данс хөтлөх боломж нь дүрэм бүрийн хувьд тухайн дүрэмд тохирсон пакетийг тоолох тоолуурыг үүсгэдэг. Ямар нэг дүрмийг шалгах үйл

явцад тухайн дүрэм ажиллаж байгаа эсэхийг тогтоох аргуудын нэг бол дүрмийг тоолуурын хамт жагсаан харах байдаг.

Бүх дүрмүүдийг дараагаар нь жагсаан харахын тулд:

```
# ipfw list
```

Бүх дүрмүүдийг тухайн дүрэм хамгийн сүүлд тохирсон цагны хамт жагсаан харахын тулд:

```
# ipfw -t list
```

Дараагийн жишээ нь данслалтын мэдээлэл буюу дүрмүүдийг тохирсон пакетийн тооны хамт харуулж байна. Эхний багана нь дүрмийн дугаар, дараа нь энэ дүрэмд тохирсон гарч байгаа пакетийн тоо, дараа нь энэ дүрэмд тохирсон орж байгаа пакетийн тоо, тэгээд дүрэм өөрөө байна.

```
# ipfw -a list
```

Статик дүрмүүдээс гадна динамик дүрмүүдийг жагсаан харахын тулд:

```
# ipfw -d list
```

Мөн хугацаа нь дууссан динамик дүрмүүдийг харахын тулд:

```
# ipfw -d -e list
```

Тоолууруудыг тэглэхийн тулд:

```
# ipfw zero
```

Зөвхөн *NUM* дугаартай тоолуурыг тэглэхийн тулд:

```
# ipfw zero NUM
```

### 31.6.5. IPFW Дүрмийн Олонлог

Пакет дотор агуулагдаж байгаа утгуудыг үндэслэн тухайн пакетийг нэвтрүүлэх болон хаахыг хэлж байгаа дүрмүүдийг IPFW дүрмийн олонлог гэнэ. Хостуудын хоорондох хоёр чиглэлтэй пакет солилцоо нь сесс харилцааг бүрдүүлнэ. Галт ханын дүрмийн олонлог нь Интернэтээс ирж байгаа пакетууд болон тэдгээрт хариу болж системээс явж байгаа пакетуудыг боловсруулдаг. Бүх TCP/IP үйлчилгээнүүдийн хувьд (жишээ нь: telnet, www, mail, г.м.) протокол болон зөвшөөрөгдсөн (сонсох) портыг урьдчилан тодорхойлсон байдаг.

Тухайн нэг үйлчилгээ рүү чиглэсэн пакетууд нь зөвшөөрөгдөөгүй (өндөр) портууд ашиглан эх хаягаас гарч очих хаягийн тухайн үйлчилгээний порт руу хүрдэг. Дээрх өгөгдлүүд (өөрөөр хэлбэл портууд ба хаягууд) нь үйлчилгээнүүдийг зөвшөөрөх эсвэл хаах дүрмүүдийг үүсгэхэд шалгуур болон ашиглагдаж болно.

Пакетийг галт хана хүлээн аваад дүрмийн олонлогт байгаа хамгийн эхний дүрэмтэй тулгах ба цааш дүрмүүдийн дугаарын өсөх дарааллын дагуу дээрээс доош нэг нэгээр шалгаж эхэлнэ. Пакет аль нэг дүрмийн сонголтын параметртай тохирвол, түүнд харгалзах үйлдлийг хийж, тухайн пакетийн хувьд цааш хайлтыг дуусгана. Энэ аргыг "эхэнд тохирсон нь дийлнэ" хайлтын арга гэнэ. Хэрэв тухайн пакет ямар ч дүрэмд тохирохгүй бол, энэ пакетийг 65535 дугаартай бүх пакетийг хааж, явуулсан хүнд нь ямар ч хариу өгөлгүй орхигдуулна гэсэн IPFW-н анхдагч дүрэмд албаар тохируулна.

`count`, `skipto` ба `tee` дүрмүүдийн дараа хайлт үргэлжилнэ.

Энд байгаа зааварчилгаанууд нь төлөвт `keep state`, `limit`, `in`, `out` болон `via` зэрэг тохируулгуудыг агуулсан дүрмүүд дээр үндэслэгдсэн байгаа. Энэ бол хамааруулсан галт ханын дүрмийн олонлогийг бичих үндсэн арга барил юм.

Галт ханын дүрмүүдтэй ажиллахдаа маш анхааралтай байх хэрэгтэй. Зарим тохиргоо серверээс *бүх холбоог тань тасалж* мэднэ.

#### 31.6.5.1. Дүрмийн Синтакс

Энд үзүүлсэн дүрмийн синтакс нь стандарт хамааруулсан галт хана үүсгэхэд шаардлагатай дүрмийн олонлогийг бичих хэмжээнд тохируулан хялбаршуулсан байгаа. Дүрмийн синтаксын бүрэн тайлбарыг ipfw(8) заавар хуудаснаас үзнэ үү.

Дүрмүүд нь түлхүүр үгүүдийг агуулна. Эдгээр түлхүүр үгүүд нь тухайн мөрөнд зүүнээс баруун тийш тодорхой дэс дараагаар бичигдэнэ. Түлхүүр үгүүд тодоор бичигдэх тул түүгээр нь ялгаж танина. Зарим түлхүүр үгүүд дэд-тохируулгатай байна. Тэр нь өөрөө түлхүүр үг байж болохоос гадна цааш дэд-тохируулгуудыг агуулсан байж болно.

`#` гэсэн тэмдэгт тайлбарын эхлэлийг заах ба дүрэм бичсэн мөрийн төгсгөлд эсвэл өөрөө бүтэн мөр байдлаар байрлана. Хоосон мөрийг тооцохгүй.

*CMD RULE_NUMBER ACTION LOGGING SELECTION STATEFUL*

##### 31.6.5.1.1. CMD

Шинэ дүрэм бүр дотоод хүснэгтэнд бичигдэхийн тулд add гэсэн түлхүүр үгийг өмнөө агуулж байх ёстой.

##### 31.6.5.1.2. RULE_NUMBER

Дүрэм бүр өөрийн дүрмийн дугаартай холбоотой бөгөөд 1..65535-н дотор байдаг.

##### 31.6.5.1.3. ACTION

Тухайн дүрмийн сонголтын үзүүлэлтэд пакет тохироход заасан action буюу үйлдлийг

гүйцэтгэх ба дүрэм нь дараах үйлдлүүдийн аль нэгтэй холбогдсон байна.

allow | accept | pass | permit

Эдгээр нь бүгд нэг зүйлийг, тухайлбал: дүрэмд тохирсон пакетуудыг нэвтрүүлж, галт ханын дүрэмтэй ажиллах явцаас гарахыг хэлж өгч байна. Эдгээр дүрмүүдийн дараа хайлт дуусна.

check-state

нь динамик дүрмийн хүснэгттэй пакетуудыг тулгана. Хэрэв тохирвол, энэ динамик дүрмийг үүсгэсэн дүрэмд харгалзах үйлдлийг гүйцэтгэнэ, үгүй бол дараагийн дүрэмд шилжинэ. check-state дүрэмд сонголтын шалгуур байхгүй. Хэрэв дүрмийн олонлогт check-state дүрэм байхгүй бол эхний keep-state эсвэл limit дүрмийг динамик дүрмийн хүснэгттэй тулгана.

deny | drop

Энэ хоёр үг хоёул дүрэмд тохирсон пакетуудыг хаяхыг заана. Хайлт энд дуусна.

#### 31.6.5.1.4. Бүртгэл хөтлөлт

log эсвэл logamount

Пакет `log` гэсэн түлхүүр үг орсон дүрэмд тохироход, энэ тухай мессеж syslogd(8) уруу SECURITY гэсэн facility нэртэйгээр бүртгэгдэнэ. Зөвхөн тухайн дүрмийн хувьд бүртгэгдсэн пакетийн тоо `logamount` параметрийн утгыг даваагүй тохиолдолд бүртгэл явагдана. Хэрэв `logamount`-н утгыг зааж өгөөгүй бол, sysctl-н `net.inet.ip.fw.verbose_limit` хувьсагчийн утгыг хязгаарын утга болгон авна. Аль ч тохиолдолд тэг гэсэн утга бүртгэлийн хязгаарыг үгүй болгоно. Хязгаарт тулсан тохиолдолд, бүртгэлийг дахин идэвхжүүлэхийн тулд бүртгэлийн тоолуурыг эсвэл тухайн дүрмийн пакет тоолуурыг дахин эхлүүлнэ. `ipfw reset log` тушаалыг үзнэ үү.

Бүртгэл нь бусад бүх пакет тохирох нөхцлүүд амжилттай нотлогдсоны дараа, мөн тухайн пакет дээр эцсийн үйлдлийг(зөвшөөрөх, татгалзах) хийхийн өмнө явагдана. Ямар дүрмүүдийн хувьд бүртгэл явуулахыг та шийдэх болно.

#### 31.6.5.1.5. Сонголт

Энд танилцуулах түлхүүр үгнүүд нь тухайн пакет дүрэмд тохирч байгаа үгүй эсэхийг тодорхойлох үед, шалгагдаж байгаа пакетийн шинжүүдийг тодорхойлно. Дараах байнгын хэрэглээний шинжүүд өгөгдсөн байдаг ба доорх дэс дарааллаар хэрэглэнэ:

udp | tcp | icmp

/etc/protocols файлд байгаа ямар ч протоколын нэрийг бас хэрэглэж болно. Харин утга нь шалгагдах протоколын нэрийг заана. Энэ бол заавал тавигдах шаардлага юм.

from src to dst

`from` ба `to` гэсэн түлхүүр үгүүд IP хаягийг шүүхэд хэрэглэгдэнэ. Дүрмэнд хэрэглэхдээ эхлэл ба очих параметрүүдийг *ХОЁУЛАНГ* зааж өгөх ёстой. `any` гэсэн тусгай түлхүүр үгийн

тусламжтай бүх IP хаягийг зөвшөөрч өгч болно. `me` гэсэн тусгай түлхүүр үг нь таны FreeBSD системийн аль нэг интерфэйс дээр тохируулсан IP хаягийг заах ба галт хана ажиллаж байгаа PC-г (өөрөөр хэлбэл энэ машин) `from me to any` эсвэл `from any to me` эсвэл `from 0.0.0.0/0 to any` эсвэл `from any to 0.0.0.0/0` эсвэл `from 0.0.0.0 to any` эсвэл `from any to 0.0.0.0` or `from me to 0.0.0.0` гэсэн байдлаар төлөөлнө. IP хаягуудыг цэгтэй тоон хэлбэр/багийн-урт байдлаар эсвэл зүгээр цэгтэй тоон хэлбэрээр бичиж болно. Энэ бол заавал тавигдах шаардлага юм. Тооцооллыг хялбар болгохын тулд net-mgmt/ipcalc портыг ашиглаж болох юм. Нэмэлт мэдээллийг хэрэгслийн вэб хуудаснаас үзэж болно: http://jodies.de/ipcalc

port number

Портын дугаарыг дэмждэг протоколуудын хувьд (TCP ба UDP гэх мэт), тааруулахыг хүсэж байгаа портын дугаарыг заавал бичиж өгөх ёстой байдаг. Портын тоон уттын оронд үйлчилгээний нэрийг(/etc/services файлаас) хэрэглэж болно.

in | out

Орж байгаа болон гарч байгаа пакетуудыг харгалзан тааруулна. `in` ба `out` нь түлхүүр үгүүд бөгөөд дүрэмд тааруулах шалгуур болгож энэ хоёр үгийн аль нэгийг заавал бичсэн байх ёстой.

via IF

Нэрээр нь зааж өгсөн интерфэйсээр дайран өнгөрч буй пакетуудыг тааруулна. `via` гэсэн түлхүүр үг нь тухайн интерфэйсийг тааруулах үйл явцын нэг хэсэг байдлаар байнга шалгаж байхыг зааж өгнө.

setup

Энэ түлхүүр үг нь TCP пакетуудын хувьд сесс эхлүүлэх хүсэлтийг зааж өгч байгаа заавал хэрэглэх түлхүүр үг юм.

keep-state

Энэ бол заавал хэрэглэх түлхүүр үг юм. Дүрэм таарахад, галт хана яг тэр протоколыг ашиглан эхлэл болон очих IP/портын хооронд үүсэх хоёр чиглэлтэй урсгалыг тааруулах анхдагч чанартай динамик дүрэм үүсгэнэ.

limit {src-addr | src-port | dst-addr | dst-port}

Дүрэмд заасантай адил параметрүүдтэй холболтын тоог *N*-р хязгаарлана. Нэг ба түүнээс дээш тооны эхлэл болон очих хаягууд, портуудыг зааж өгч болно. `limit` ба `keep-state`-г нэг дүрэмд хамтад нь хэрэглэж болохгүй. `limit` тохируулга нь `keep-state`-тэй адил төлөвт функцуудыг гүйцэтгэхээс гадна өөрийн нэмэлт функцүүлтэй.

#### 31.6.5.2. Төлөвт Дүрмийн Тохируулгууд

Төлөвт шүүлт нь урсгалыг сесс харилцаанаас бүрдэж буй хоёр чиглэлтэй пакет солилцоо гэж үздэг. Мөн энэ нь сесс эхлүүлсэн болон хүлээн авсан хостуудын хоорондох сесс харилцаа нь хоёр чиглэлтэй пакет солилцох процедуртай зохицож байгаа эсэхийг магадлах чадвартай байдаг. Сесс харилцааны загварт зохицоогүй пакетийг хуурамч гэж үзэн хүлээж

авахгүй.

`check-state` нь IPFW дүрмийн олонлогийн хаана нь пакетийг динамик дүрмүүдийн боломжоор шалгахыг тогтооно. Таарсан тохиолдолд, пакет галт ханыг нэвтэрч цааш явах ба энэ хоёр чиглэлт сесс харилцааны туршид солилцох пакетуудын хувьд шинэ динамик дүрэм үүснэ. Тааараагүй тохиолдолд, пакет дүрмийн олонлогийн дараагийн дүрэмд шалгагдахаар шилжинэ.

Динамик дүрмүүдийн боломж нь маш олон тооны динамик дүрмүүдийг нээдэг SYN-живүүлэх халдлагаас үүсэх нөөцийн хомсдолд эмзэг байдаг. Энэ халдлагаас зайлсхийхийн тулд FreeBSD `limit` гэсэн шинэ тохируулгыг нэмж өгсөн байдаг. Энэ тохируулгын тусламжтай нэгэн зэрэг явагдах сесс харилцааны тоог хязгаарлана. `limit` тохируулгад зааж өгсөн эхлэл болон очих талбаруудаар пакетийн IP хаягийг асуулга явуулах замаар шалгасны дараа, энэ дүрэмд энэ IP хаягийн хослол хэдэн удаа таарсан тоог харгалзан хэрэв энэ тоо `limit`-д зааснаас давсан бол тухайн пакетийг гээнэ.

#### 31.6.5.3. Галт ханын мессежийг бүртгэх

Бүртгэл хөтлөлтийн ашиг тус тодорхой юм: Таны бүртгэхээр идэвхжүүлсэн дүрмүүдийн хувьд, ямар пакетууд гээгдсэн, тэдгээр нь ямар хаягаас ирсэн, хаашаа явж байсан зэрэг мэдээллийг эргэн харах боломжийг олгох ба гадны халдлагыг мөрдөхөд танд чухал хувь нэмэр болно.

Бүртгэл хөтлөх боломжийг идэвхжүүлсэн хэдий ч, IPFW нь өөрөө ямар ч дүрмийг үүсгэхгүй. Администратор аль дүрмүүдийн хувьд бүртгэл явуулахыг шийдэн, тэдгээр дүрмүүддээ `log` гэсэн түлхүүр үгийг нэмж бичнэ. Ихэвчлэн зөвхөн татгалзах дүрмүүдийг бүртгэдэг, жишээлбэл ирж буй ICMP ping-г татгалзах гэх мэт. Хамгийн сүүлд байгаа "ipfw default deny everything" дүрмийг хувилан `log` түлхүүр үгтэйгээр үүсгэх нь элбэг байдаг. Ийм байдлаар дүрмийн олонлогийн аль ч дүрмэнд тааараагүй пакетуудыг харах боломжтой болно.

Бүртгэл хөтлөлт нь хоёр талдаа иртэй сэлэмтэй адил юм, хэрэв та хайхрамжгүй хандвал, диск дүүрэн бүртгэлийн мэдээлэл дотроо учраа олохгүй суух болно. Дискийг дүүргэх DoS халдлага нь хамгийн эртний халдлагуудын нэг юм. Эдгээр бүртгэлийн мессеж нь syslogd-д бичигдэхээс гадна, root консол дэлгэцэн дээр гарах учир удахгүй ядаргаатай санагдаж эхэлдэг.

`IPFIREWALL_VERBOSE_LIMIT=5` гэсэн цөмийн тохируулга нь системийн бүртгэл хөтлөгч болох syslogd(8) уруу шидэгдэж байгаа тухайн дүрэмд тохирсон пакетад харгалзах дараалсан мессежийн тоог хязгаарлана. Энэ тохируулгыг идэвхжүүлсэн үед, тодорхой дүрмийн хувьд дараалсан мессежийн тоог зааж өгсөн тоогоор хязгаарлана. Нэг ижил зүйлийг хэлсэн 200 бүртгэлийн бичлэгээс мэдэж авах зүйл хомс юм. Жишээ нь, тодорхой дүрмийн хувьд дараалсан таван бичлэг syslogd-д бичигдэнэ, үлдсэн дараалсан ижил бичлэгүүд тоологдоод syslogd-д дараах байдалтай бичигдэнэ:

```
last message repeated 45 times
```

Бүртгэл хөтлөгдөж байгаа бүх пакетуудын мессежүүд /etc/syslog.conf файлд анхдагч байдлаар зааж өгсөн /var/log/security файлд бичигдэнэ.

#### 31.6.5.4. Дүрмийн скриптийг бүтээх

Туршлагатай IPFW хэрэглэгчид дүрмүүдийг агуулсан файл үүсгээд түүнийгээ скрипт байдлаар ажиллуулах боломжтой байхаар бичдэг. Үүний гол давуу тал нь дүрмүүдийг идэвхжүүлэхийн тулд системийг дахин ачаалах шаардлагагүй болно. Энэ аргыг ашиглан хэдэн ч удаа дараалан галт ханын дүрмүүдийг ачаалж болох тул шинэ дүрмүүдийг шалгах үед хэрэглэхэд тохиромжтой байдаг. Скрипт учраас олон дахин бичигдэж байгаа утгын оронд симбол орлуулалтыг ашиглах боломжтой. Энэ талаар дараах жишээн дээрээс харна уу.

Энд хэрэглэгдсэн скриптийн синтакс нь sh(1), csh(1), tcsh(1) бүрхүүл дээр ажиллах боломжтой. Симбол орлуулалттай талбарууд нь урдаа $ буюу долларын тэмдэгтэй байна. Симбол талбарууд нь $ тэмдэг урдаа байхгүй. Симбол талбарыг орлох утга нь давхар хашилтан (") дотор байрлана.

Дүрмийн файлаа дараах байдалтай үүсгэж эхлэх хэрэгтэй:

```
################ start of example ipfw rules script #############
#
ipfw -q -f flush       # Delete all rules
# Set defaults
oif="tun0"             # out interface
odns="192.0.2.11"      # ISP's DNS server IP address
cmd="ipfw -q add "     # build rule prefix
ks="keep-state"        # just too lazy to key this each time
$cmd 00500 check-state
$cmd 00502 deny all from any to any frag
$cmd 00501 deny tcp from any to any established
$cmd 00600 allow tcp from any to any 80 out via $oif setup $ks
$cmd 00610 allow tcp from any to $odns 53 out via $oif setup $ks
$cmd 00611 allow udp from any to $odns 53 out via $oif $ks
################### End of example ipfw rules script ############
```

Байх ёстой бүх зүйлс бүгд байна. Энэ жишээн дээр дүрмүүд чухал биш, харин симбол орлуулалт хэрхэн ажилладагыг харуулсан байна.

Хэрэв дээрх жишээ /etc/ipfw.rules нэртэй файл дотор байсан бол, эдгээр дүрмүүдийг дараах тушаалыг өгч дахин ачаалах боломжтой:

```
# sh /etc/ipfw.rules
```

/etc/ipfw.rules гэсэн файл ямар ч нэртэй байж болох ба таны хүссэн ямар ч газар байж болно.

Дараах скриптийг хоёр аргын нэгээр хэрэглэж болно:

Дээрхтэй адил зүйлсийг дараах тушаалыг гараар оруулан гүйцэтгэж болно:

```
# ipfw -q -f flush
```

```
# ipfw -q add check-state
# ipfw -q add deny all from any to any frag
# ipfw -q add deny tcp from any to any established
# ipfw -q add allow tcp from any to any 80 out via tun0 setup keep-state
# ipfw -q add allow tcp from any to 192.0.2.11 53 out via tun0 setup keep-state
# ipfw -q add 00611 allow udp from any to 192.0.2.11 53 out via tun0 keep-state
```

#### 31.6.5.5. Төлөвт дүрмийн олонлог

Дараах NAT хийгдээгүй дүрмийн олонлог нь аюулгүй байдлыг маш сайн хангасан, хамааруулсан галт ханын дүрмүүдийг хэрхэн бичих жишээ юм. Хамааруулсан галт хана нь зөвхөн pass дүрмүүдэд тохирсон үйлчилгээг зөвшөөрч, бусдыг бүгдийг хаана. Сүлжээний бүх сегментийг хамгаалахаар хийгдсэн галт хананууд хамгийн багадаа хоёр интерфэйстэй байх ба галт хана маягаар ажиллахын тулд дүрмүүдийг тусгайлан зааж өгсөн байна.

FreeBSD-г оролцуулаад бүх UNIX® төрлийн систем нь үйлдлийн систем дэх дотоод харилцаандаа lo0 интерфэйс болон 127.0.0.1 гэсэн IP хаягийг хэрэглэхээр бүтээгдсэн байдаг. Галт ханын дүрмүүд нь дотооддоо хэрэглэж байгаа эдгээр пакетуудыг чөлөөтэй нэвтрүүлэх дүрмүүдийг агуулсан байх ёстой.

Интернэттэй холбогдож байгаа интерфэйс дээр гадагшаа болон дотогшоо холболтуудыг удирдах болон хянах дүрмүүдийг байрлуулна. Энэ нь таны PPP tun0 интерфэйс эсвэл таны DSL эсвэл кабель модемд холбогдсон NIC байж болно.

Галт ханын цаана байгаа хувийн LAN-уудад нэг болон түүнээс дээш тооны NIC-ууд холбогдсон тохиолдолд, тэдгээр интерфэйсүүдийн хувьд тэдгээр LAN интерфэйсүүдээс ирсэн пакетуудыг чөлөөтэй нэвтрүүлэх дүрмийг агуулсан байх ёстой.

Дүрмүүд эхлээд гурван үндсэн хэсэгт хуваагдан зохион байгуулагдсан байх ёстой: бүх чөлөөт халдашгүй интерфэйсүүд, нийтийн гадагшаа интерфэйс болон нийтийн дотогшоо интерфэйс.

Нийтийн интерфэйс бүр дээр байгаа дүрмүүдийн дараалал нь хамгийн олон тохиолддог дүрмүүд нь хамгийн түрүүнд цөөн тохиолддог дүрмүүдээс өмнө байхаар, тухайн интерфэйс болон чиглэлийн хувьд хаах болон бүртгэх дүрмүүд нь хамгийн сүүлд байхаар бичигдсэн байна.

Дараах жишээн дээрх гадагшаа хэсэг нь Интернэт хандалтыг зөвшөөрч өгөх үйлчилгээг тодорхойлох сонголтын утгуудыг агуулсан зөвхөн `allow` дүрмүүдээс бүрдэж байна. Бүх дүрмүүд `proto`, `port`, `in/out`, `via` ба `keep state` тохируулгуудыг агуулсан байгаа. `proto tcp` дүрмүүд нь төлөвт байдлыг идэвхжүүлж байгаа пакетийг keep state төлөвт хүснэгтэд нэмэх байдлаар, сесс эхлүүлэх хүсэлтийг тодорхойлох зорилгоор `setup` тохируулгыг агуулсан байна.

Дотогшоо хэсэгт эхлээд хүсээгүй пакетуудыг хаах дүрмүүдийг бичсэн байна. Энэ нь хоёр өөр шалтгаантай. Эхнийх нь хортой пакетууд нь зөвшөөрөгдсөн пакетийн нэг хэсэг байж болох талтай. Эдгээр пакетуудыг `allow` дүрэмд таарсан хэсэг дээр тулгуурлан зөвшөөрөлгүйгээр орхих ёстой. Хоёр дахь шалтгаан нь тухайн хэсгийн хамгийн сүүлийн дүрмээр хааж бүртгэхийн оронд мэдэгдэж байгаа, сонирхолгүй пакетуудыг чимээгүйхэн хааж болох юм.

Бүх пакетуудыг бүртгээд хаадаг хэсгийн хамгийн сүүлийн дүрмийг өөрийн систем уруу халдаж байгаа хүмүүсийг шүүхэд шаардагдах халдлагын нотолгоог цуглуулахад хэрэглэж болно.

Өөр нэг тэмдэглэн хэлэх зүйл бол, хүсээгүй пакетуудын хариуд ямар ч хариу явуулахгүй байх ёстойг санах хэрэгтэй. Буруу пакетууд орхигдож алга болох ёстой. Ингэснээр халдлага явуулагч нь түүний явуулсан пакетууд таны системд хүрсэн эсэх талаар юу ч мэдэхгүй үлдэх болно. Таны системийн талаар хэдий хэр бага мэднэ, төдий чинээ аюулгүй байна гэсэн үг юм. Танигдаагүй портын дугаартай пакетуудын хувьд /etc/services/ файлаас эсвэл http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers хаягаар тухайн порт ямар зориулалтаар ашиглагддагийг орж шалгаарай. Троянуудын хэрэглэдэг портын дугааруудыг http://www.sans.org/security-resources/idfaq/oddports.php хаягаар орж шалгаарай.

#### 31.6.5.6. Хамааруулсан дүрмийн олонлогийн жишээ

Дараах NAT хийгдээгүй дүрмийн олонлог нь бүрэн хэмжээний хамааруулсан дүрмийн олонлог байгаа юм. Та энэ дүрмүүдийг өөрийн системдээ ашиглахад буруудах юмгүй. Зөвшөөрөхийг хүсэхгүй байгаа үйлчилгээнүүдийн хувьд харгалзах `pass` дүрмийг далдлаарай. Хэрэв бүртгэлд байгаа зарим мэдэгдлийг дахин харахыг хүсэхгүй, бүртгэхийг хүсэхгүй байгаа бол дотогшоо хэсэгт `deny` дүрэм нэмж бичээрэй. Дүрэм бүрт байгаа dc0 гэсэн интерфэйсийн нэрийн оронд таны системийг Интернэт уруу холбож байгаа NIC картны интерфэйсийн нэрээр сольж тавиарай. Хэрэглэгчийн PPP-н хувьд, энэ нь tun0 байна.

Эдгээр дүрмүүдийг хэрэглэх явцад хэв маяг олж харах болно.

- Интернэт уруу чиглэсэн сесс эхлүүлэх хүсэлтийг төлөөлж байгаа илэрхийллүүд бүгд `keep-state` хэрэглэж байгаа.
- Интернэтээс ирж буй бүх зөвшөөрөгдсөн үйлчилгээнүүд живүүлэх халдлагыг зогсоох үүднээс `limit` гэсэн тохируулгын хамт бичигдсэн байгаа.
- Бүх дүрмүүд чиглэлийг тодотгохын тулд `in` эсвэл `out`-г хэрэглэсэн байгаа.
- Бүх дүрмүүд пакетийн дайран өнгөрөх *interface-name* интерфэйсийг тодорхойлж өгөхдөө `via`-г хэрэглэсэн байгаа.

Дараах дүрмүүд /etc/ipfw.rules дотор байрлана.

```
################ Start of IPFW rules file ###################################
# Flush out the list before we begin.
ipfw -q -f flush

# Set rules command prefix
cmd="ipfw -q add"
pif="dc0"     # public interface name of NIC
              # facing the public Internet

#############################################################################
# No restrictions on Inside LAN Interface for private network
# Not needed unless you have LAN.
# Change xl0 to your LAN NIC interface name
```

```
#####################################################################
#$cmd 00005 allow all from any to any via xl0

#####################################################################
# No restrictions on Loopback Interface
#####################################################################
$cmd 00010 allow all from any to any via lo0

#####################################################################
# Allow the packet through if it has previous been added to the
# the "dynamic" rules table by a allow keep-state statement.
#####################################################################
$cmd 00015 check-state

#####################################################################
# Interface facing Public Internet (Outbound Section)
# Interrogate session start requests originating from behind the
# firewall on the private network or from this gateway server
# destined for the public Internet.
#####################################################################

# Allow out access to my ISP's Domain name server.
# x.x.x.x must be the IP address of your ISP.s DNS
# Dup these lines if your ISP has more than one DNS server
# Get the IP addresses from /etc/resolv.conf file
$cmd 00110 allow tcp from any to x.x.x.x 53 out via $pif setup keep-state
$cmd 00111 allow udp from any to x.x.x.x 53 out via $pif keep-state

# Allow out access to my ISP's DHCP server for cable/DSL configurations.
# This rule is not needed for .user ppp. connection to the public Internet.
# so you can delete this whole group.
# Use the following rule and check log for IP address.
# Then put IP address in commented out rule & delete first rule
$cmd 00120 allow log udp from any to any 67 out via $pif keep-state
#$cmd 00120 allow udp from any to x.x.x.x 67 out via $pif keep-state

# Allow out non-secure standard www function
$cmd 00200 allow tcp from any to any 80 out via $pif setup keep-state

# Allow out secure www function https over TLS SSL
$cmd 00220 allow tcp from any to any 443 out via $pif setup keep-state

# Allow out send & get email function
$cmd 00230 allow tcp from any to any 25 out via $pif setup keep-state
$cmd 00231 allow tcp from any to any 110 out via $pif setup keep-state

# Allow out FBSD (make install & CVSUP) functions
# Basically give user root "GOD" privileges.
$cmd 00240 allow tcp from me to any out via $pif setup keep-state uid root

# Allow out ping
```

```
$cmd 00250 allow icmp from any to any out via $pif keep-state

# Allow out Time
$cmd 00260 allow tcp from any to any 37 out via $pif setup keep-state

# Allow out nntp news (i.e. news groups)
$cmd 00270 allow tcp from any to any 119 out via $pif setup keep-state

# Allow out secure FTP, Telnet, and SCP
# This function is using SSH (secure shell)
$cmd 00280 allow tcp from any to any 22 out via $pif setup keep-state

# Allow out whois
$cmd 00290 allow tcp from any to any 43 out via $pif setup keep-state

# deny and log everything else that.s trying to get out.
# This rule enforces the block all by default logic.
$cmd 00299 deny log all from any to any out via $pif

########################################################################
# Interface facing Public Internet (Inbound Section)
# Check packets originating from the public Internet
# destined for this gateway server or the private network.
########################################################################

# Deny all inbound traffic from non-routable reserved address spaces
$cmd 00300 deny all from 192.168.0.0/16 to any in via $pif  #RFC 1918 private IP
$cmd 00301 deny all from 172.16.0.0/12 to any in via $pif     #RFC 1918 private IP
$cmd 00302 deny all from 10.0.0.0/8 to any in via $pif          #RFC 1918 private IP
$cmd 00303 deny all from 127.0.0.0/8 to any in via $pif        #loopback
$cmd 00304 deny all from 0.0.0.0/8 to any in via $pif            #loopback
$cmd 00305 deny all from 169.254.0.0/16 to any in via $pif   #DHCP auto-config
$cmd 00306 deny all from 192.0.2.0/24 to any in via $pif       #reserved for docs
$cmd 00307 deny all from 204.152.64.0/23 to any in via $pif  #Sun cluster interconnect
$cmd 00308 deny all from 224.0.0.0/3 to any in via $pif         #Class D & E multicast

# Deny public pings
$cmd 00310 deny icmp from any to any in via $pif

# Deny ident
$cmd 00315 deny tcp from any to any 113 in via $pif

# Deny all Netbios service. 137=name, 138=datagram, 139=session
# Netbios is MS/Windows sharing services.
# Block MS/Windows hosts2 name server requests 81
$cmd 00320 deny tcp from any to any 137 in via $pif
$cmd 00321 deny tcp from any to any 138 in via $pif
$cmd 00322 deny tcp from any to any 139 in via $pif
$cmd 00323 deny tcp from any to any 81 in via $pif

# Deny any late arriving packets
```

```
$cmd 00330 deny all from any to any frag in via $pif

# Deny ACK packets that did not match the dynamic rule table
$cmd 00332 deny tcp from any to any established in via $pif

# Allow traffic in from ISP's DHCP server. This rule must contain
# the IP address of your ISP.s DHCP server as it.s the only
# authorized source to send this packet type.
# Only necessary for cable or DSL configurations.
# This rule is not needed for .user ppp. type connection to
# the public Internet. This is the same IP address you captured
# and used in the outbound section.
#$cmd 00360 allow udp from any to x.x.x.x 67 in via $pif keep-state

# Allow in standard www function because I have apache server
$cmd 00400 allow tcp from any to me 80 in via $pif setup limit src-addr 2

# Allow in secure FTP, Telnet, and SCP from public Internet
$cmd 00410 allow tcp from any to me 22 in via $pif setup limit src-addr 2

# Allow in non-secure Telnet session from public Internet
# labeled non-secure because ID & PW are passed over public
# Internet as clear text.
# Delete this sample group if you do not have telnet server enabled.
$cmd 00420 allow tcp from any to me 23 in via $pif setup limit src-addr 2

# Reject & Log all incoming connections from the outside
$cmd 00499 deny log all from any to any in via $pif

# Everything else is denied by default
# deny and log all packets that fell through to see what they are
$cmd 00999 deny log all from any to any
################## End of IPFW rules file ###############################
```

#### 31.6.5.7. NAT болон Төлөвт дүрмийн олонлогийн жишээ

IPFW-н NAT функцыг идэвхжүүлэхийн тулд зарим нэмэлт тохиргооны илэрхийллүүдийг идэвхжүүлэх хэрэгтэй болдог. Цөмийн эх кодын бусад IPFIREWALL илэрхийллүүд дээр `option IPDIVERT` илэрхийллийг нэмж эмхэтгэн тусгайлан бэлдсэн цөмийг гаргаж авах хэрэгтэй.

/etc/rc.conf доторх энгийн IPFW тохируулгууд дээр нэмж дараах тохируулгууд хэрэгтэй болно.

```
natd_enable="YES"                   # Enable NATD function
natd_interface="rl0"                # interface name of public Internet NIC
natd_flags="-dynamic -m"            # -m = preserve port numbers if possible
```

Төлөвт дүрмүүдийг `divert natd` (Сүлжээний хаягийн Хөрвүүлэлт) дүрмийн хамт хэрэглэх нь дүрмийн олонлог бичих логикийг төвөгтэй болгодог. `check-state` ба `divert natd` дүрмүүдийн

дүрмийн олонлог дахь байрлал нь маш их нөлөөтэй. Энэ нь хялбар дайраад-өнгөрөх логик урсгал биш болно. `skipto` гэсэн шинэ үйлдлийн төрлийг хэрэглэх болно. `skipto`-г хэрэглэхдээ `skipto` дүрмийн дугаар хаашаа үсрэхээ мэдэж байхын тулд бүх дүрмүүдийг дугаарлах хэрэгтэй болно.

Дараах тайлбаргүй жишээн дээр пакет дүрмийн олонлогоор дайрч өнгөрөх дарааллыг тайлбарлахаар сонгон авсан дүрэм бичих арга байгаа юм.

Дүрэмтэй ажиллах процесс дүрмийн файлд байгаа хамгийн эхний дүрмээр эхлэн цааш дүрмүүдийг нэг нэгээр уншин, файлын төгсгөл хүртэл эсвэл пакет аль нэг дүрмийн сонголтын шалгуурт тохирч галт ханыг орхих хүртэл үргэлжилнэ. 100, 101, 450, 500, ба 510 дугаартай дүрмүүдийн байрлалыг сайн анзаарах хэрэгтэй. Эдгээр дүрмүүд нь гадагшаа болон дотогшоо чиглэлтэй пакетуудын хөрвүүлэлтийг удирдах бөгөөд ингэснээр keep-state динамик хүснэгтэн дэх тэдгээрт харгалзах мөрөнд хувийн LAN IP хаяг бүртгэгдсэн байх нөхцөлийг хангана. Дараа нь, бүх зөвшөөрөх болон татгалзах дүрмүүдэд пакетийн явж буй чиглэл (өөрөөр хэлбэл гадагшаа эсвэл дотогшоо) ба интерфэйсийг зааж өгсөн байгааг анзаараарай. Мөн гадагшаа сесс эхлүүлэх хүсэлтүүд, сүлжээний хаягийн хөрвүүлэлтийн бүх `skipto rule 500`-с эхэлж байгааг анзаарна уу.

Нэгэн LAN хэрэглэгч вэб хуудас үзэхийн тулд вэб хөтчийг хэрэглэж байна гэж бодъё. Веб хуудсууд 80-р портоор дамждаг. Пакет галт хананд ирнэ, гадагшаа чиглэж байгаа тул 100-р дүрмэнд тохирохгүй. 101-р дүрмийг мөн өнгөрнө, яагаад гэвэл энэ нь хамгийн анхны пакет тул keep-state динамик хүснэгтэнд хараахан бичигдэж амжаагүй байгаа. Пакет эцэст нь 125-р дүрэм дээр ирж, дүрэмд таарна. Энэ пакет Интернэт уруу харсан NIC-р гадагшаа гарч байгаа. пакетийн эхлэл IP хаяг нь хувийн LAN IP хаяг хэвээр байгаа. Энэ дүрмэнд таарах үед хоёр үйлдэл хийгдэнэ. `keep-state` тохируулга энэ дүрмийг keep-state динамик дүрмийн хүснэгтэнд нэмнэ, мөн зааж өгсөн үйлдлийг гүйцэтгэнэ. Үйлдэл нь динамик хүснэгтэд нэмэгдсэн мэдээллийн нэг хэсэг байна. Энэ тохиолдолд `skipto rule 500` байна. 500-р дүрэм нь пакетийн IP хаягийг NAT хийж, пакетийг гадагш явуулна. Үүнийг бүү мартаарай, энэ бол маш чухал шүү. Энэ пакет өөрийн замаар хүрэх газраа хүрэх бөгөөд хариу пакет үүсч буцаж илгээгдэнэ. Энэ шинэ пакет буцаж ирэхдээ мөн л энэ дүрмийн олонлогийг дайрна. Энэ үед харин 100-р дүрэмд тохирч, очих IP хаяг нь буцаж харгалзах LAN IP хаяг уруу хөрвүүлэгдэнэ. Дараа нь `check-state` дүрмээр гарах ба хүснэгтэнд явагдаж байгаа сесс харилцаанд оролцож байгаа гэж тэмдэглэгдсэн тул цааш LAN уруу нэвтрэн орно. Тэгээд өөрийг нь анх явуулсан LAN PC уруу очих ба алсын серверээс өөр хэсэг өгөгдлийг авахыг хүссэн шинэ пакетийг явуулна. Энэ удаа энэ пакет `check-state` дүрмээр шалгагдах ба түүний гадагшаа урсгалд харгалзах мөр олдох тул харгалзах үйлдэл `skipto 500`-г гүйцэтгэнэ. Пакет 500-р мөр уруу үсэрч NAT хийгдэн цааш өөрийн замаар явах болно.

Дотогшоо урсгал дээр, идэвхтэй сесс харилцаанд оролцож байгаа гаднаас ирж байгаа бүх зүйлс автоматаар `check-state` дүрмээр болон зохих `divert natd` дүрмүүдээр шийдэгдэнэ. Энд бидний хийх ёстой зүйл бол хэрэггүй пакетуудыг татгалзаж, зөвшөөрөгдсөн үйлчилгээг нэвтрүүлэх юм. Галт ханын байгаа машин дээр апачи сервер ажиллаж байна, тэгээд Интернэтээс хүмүүс энэ дотоод вэб сайт уруу хандаж байна гэж бодъё. Шинэ дотогшоо сесс эхлүүлэх хүсэлтийн пакет 100-р дүрэмд тохирох бөгөөд түүний IP хаяг галт ханын LAN IP хаяг уруу оноолт хийгдэнэ. Дараа нь энэ пакет цааш бүх дүрмүүдээр шалгагдан эцэст нь 425-р дүрэмд тохирно. Энэ дүрэмд таарах үед хоёр үйлдэл хийгдэнэ. Энэ дүрэм keep-state динамик дүрмийн хүснэгтэнд нэмэгдэнэ, мөн зааж өгсөн үйлдлийг гүйцэтгэнэ. Гэвч энэ тохиолдолд энэ эхлэл IP хаягнаас эхэлсэн шинэ сесс эхлүүлэх хүсэлтийн тоо 2-оор

хязгаарлагдана. Энэ нь тодорхой порт дээр ажиллаж байгаа үйлчилгээний хувьд DoS халдлагаас хамгаална. Харгалзах үйлдэл нь `allow` тул пакет LAN уруу нэвтэрнэ. Хариу болон үүсгэгдсэн пакетыг `check-state` дүрэм идэвхтэй сесс харилцаанд хамаарч байгаа гэж танина. Тэгээд 500-р дүрэм уруу шилжүүлэн, пакет тэнд NAT хийгдээд цааш гадагшаа интерфэйсээр гарна.

Жишээ дүрмийн олонлог #1:

```
#!/bin/sh
cmd="ipfw -q add"
skip="skipto 500"
pif=rl0
ks="keep-state"
good_tcpo="22,25,37,43,53,80,443,110,119"

ipfw -q -f flush

$cmd 002 allow all from any to any via xl0  # exclude LAN traffic
$cmd 003 allow all from any to any via lo0  # exclude loopback traffic

$cmd 100 divert natd ip from any to any in via $pif
$cmd 101 check-state

# Authorized outbound packets
$cmd 120 $skip udp from any to xx.168.240.2 53 out via $pif $ks
$cmd 121 $skip udp from any to xx.168.240.5 53 out via $pif $ks
$cmd 125 $skip tcp from any to any $good_tcpo out via $pif setup $ks
$cmd 130 $skip icmp from any to any out via $pif $ks
$cmd 135 $skip udp from any to any 123 out via $pif $ks

# Deny all inbound traffic from non-routable reserved address spaces
$cmd 300 deny all from 192.168.0.0/16  to any in via $pif  #RFC 1918 private IP
$cmd 301 deny all from 172.16.0.0/12   to any in via $pif  #RFC 1918 private IP
$cmd 302 deny all from 10.0.0.0/8      to any in via $pif  #RFC 1918 private IP
$cmd 303 deny all from 127.0.0.0/8     to any in via $pif  #loopback
$cmd 304 deny all from 0.0.0.0/8       to any in via $pif  #loopback
$cmd 305 deny all from 169.254.0.0/16  to any in via $pif  #DHCP auto-config
$cmd 306 deny all from 192.0.2.0/24    to any in via $pif  #reserved for docs
$cmd 307 deny all from 204.152.64.0/23 to any in via $pif  #Sun cluster
$cmd 308 deny all from 224.0.0.0/3     to any in via $pif  #Class D & E multicast

# Authorized inbound packets
$cmd 400 allow udp from xx.70.207.54 to any 68 in $ks
$cmd 420 allow tcp from any to me 80 in via $pif setup limit src-addr 1

$cmd 450 deny log ip from any to any

# This is skipto location for outbound stateful rules
$cmd 500 divert natd ip from any to any out via $pif
$cmd 510 allow ip from any to any
```

```
######################### end of rules  ###################
```

Дараах жишээ дээрхтэй нилээд төстэй боловч туршлагагүй IPFW дүрэм бичигчид зориулан дүрмүүд юунд зориулагдсаныг тайлбарласан тайлбартай байгаагаараа онцлог юм.

Жишээ дүрмийн олонлог #2:

```
#!/bin/sh
################ Start of IPFW rules file ###############################
# Flush out the list before we begin.
ipfw -q -f flush

# Set rules command prefix
cmd="ipfw -q add"
skip="skipto 800"
pif="rl0"     # public interface name of NIC
              # facing the public Internet

#########################################################################
# No restrictions on Inside LAN Interface for private network
# Change xl0 to your LAN NIC interface name
#########################################################################
$cmd 005 allow all from any to any via xl0

#########################################################################
# No restrictions on Loopback Interface
#########################################################################
$cmd 010 allow all from any to any via lo0

#########################################################################
# check if packet is inbound and nat address if it is
#########################################################################
$cmd 014 divert natd ip from any to any in via $pif

#########################################################################
# Allow the packet through if it has previous been added to the
# the "dynamic" rules table by a allow keep-state statement.
#########################################################################
$cmd 015 check-state

#########################################################################
# Interface facing Public Internet (Outbound Section)
# Check session start requests originating from behind the
# firewall on the private network or from this gateway server
# destined for the public Internet.
#########################################################################

# Allow out access to my ISP's Domain name server.
# x.x.x.x must be the IP address of your ISP's DNS
```

```
# Dup these lines if your ISP has more than one DNS server
# Get the IP addresses from /etc/resolv.conf file
$cmd 020 $skip tcp from any to x.x.x.x 53 out via $pif setup keep-state

# Allow out access to my ISP's DHCP server for cable/DSL configurations.
$cmd 030 $skip udp from any to x.x.x.x 67 out via $pif keep-state

# Allow out non-secure standard www function
$cmd 040 $skip tcp from any to any 80 out via $pif setup keep-state

# Allow out secure www function https over TLS SSL
$cmd 050 $skip tcp from any to any 443 out via $pif setup keep-state

# Allow out send & get email function
$cmd 060 $skip tcp from any to any 25 out via $pif setup keep-state
$cmd 061 $skip tcp from any to any 110 out via $pif setup keep-state

# Allow out FreeBSD (make install & CVSUP) functions
# Basically give user root "GOD" privileges.
$cmd 070 $skip tcp from me to any out via $pif setup keep-state uid root

# Allow out ping
$cmd 080 $skip icmp from any to any out via $pif keep-state

# Allow out Time
$cmd 090 $skip tcp from any to any 37 out via $pif setup keep-state

# Allow out nntp news (i.e. news groups)
$cmd 100 $skip tcp from any to any 119 out via $pif setup keep-state

# Allow out secure FTP, Telnet, and SCP
# This function is using SSH (secure shell)
$cmd 110 $skip tcp from any to any 22 out via $pif setup keep-state

# Allow out whois
$cmd 120 $skip tcp from any to any 43 out via $pif setup keep-state

# Allow ntp time server
$cmd 130 $skip udp from any to any 123 out via $pif keep-state

################################################################
# Interface facing Public Internet (Inbound Section)
# Check packets originating from the public Internet
# destined for this gateway server or the private network.
################################################################

# Deny all inbound traffic from non-routable reserved address spaces
$cmd 300 deny all from 192.168.0.0/16  to any in via $pif  #RFC 1918 private IP
$cmd 301 deny all from 172.16.0.0/12   to any in via $pif  #RFC 1918 private IP
$cmd 302 deny all from 10.0.0.0/8      to any in via $pif  #RFC 1918 private IP
$cmd 303 deny all from 127.0.0.0/8     to any in via $pif  #loopback
```

```
$cmd 304 deny all from 0.0.0.0/8       to any in via $pif  #loopback
$cmd 305 deny all from 169.254.0.0/16  to any in via $pif  #DHCP auto-config
$cmd 306 deny all from 192.0.2.0/24    to any in via $pif  #reserved for docs
$cmd 307 deny all from 204.152.64.0/23 to any in via $pif  #Sun cluster
$cmd 308 deny all from 224.0.0.0/3     to any in via $pif  #Class D & E multicast

# Deny ident
$cmd 315 deny tcp from any to any 113 in via $pif

# Deny all Netbios service. 137=name, 138=datagram, 139=session
# Netbios is MS/Windows sharing services.
# Block MS/Windows hosts2 name server requests 81
$cmd 320 deny tcp from any to any 137 in via $pif
$cmd 321 deny tcp from any to any 138 in via $pif
$cmd 322 deny tcp from any to any 139 in via $pif
$cmd 323 deny tcp from any to any 81  in via $pif

# Deny any late arriving packets
$cmd 330 deny all from any to any frag in via $pif

# Deny ACK packets that did not match the dynamic rule table
$cmd 332 deny tcp from any to any established in via $pif

# Allow traffic in from ISP's DHCP server. This rule must contain
# the IP address of your ISP's DHCP server as it's the only
# authorized source to send this packet type.
# Only necessary for cable or DSL configurations.
# This rule is not needed for 'user ppp' type connection to
# the public Internet. This is the same IP address you captured
# and used in the outbound section.
$cmd 360 allow udp from x.x.x.x to any 68 in via $pif keep-state

# Allow in standard www function because I have Apache server
$cmd 370 allow tcp from any to me 80 in via $pif setup limit src-addr 2

# Allow in secure FTP, Telnet, and SCP from public Internet
$cmd 380 allow tcp from any to me 22 in via $pif setup limit src-addr 2

# Allow in non-secure Telnet session from public Internet
# labeled non-secure because ID & PW are passed over public
# Internet as clear text.
# Delete this sample group if you do not have telnet server enabled.
$cmd 390 allow tcp from any to me 23 in via $pif setup limit src-addr 2

# Reject & Log all unauthorized incoming connections from the public Internet
$cmd 400 deny log all from any to any in via $pif

# Reject & Log all unauthorized out going connections to the public Internet
$cmd 450 deny log all from any to any out via $pif

# This is skipto location for outbound stateful rules
```

```
$cmd 800 divert natd ip from any to any out via $pif
$cmd 801 allow ip from any to any

# Everything else is denied by default
# deny and log all packets that fell through to see what they are
$cmd 999 deny log all from any to any
################## End of IPFW rules file ###############################
```

# Chapter 32. Сүлжээний нэмэлт ойлголтууд

## 32.1. Ерөнхий агуулга

Энэ бүлэг нь хэд хэдэн сүлжээний дэвшилтэт нэмэлт сэдвүүдийг хамрах болно.

Энэ бүлгийг уншсаны дараа, та дараах зүйлсийг мэдэх болно:

- Гарцууд болон чиглүүлэлтүүдийн үндсүүд.
- IEEE® 802.11 болон Bluetooth® төхөөрөмжүүдийг хэрхэн суулгах талаар.
- FreeBSD-г гүүр болгож хэрхэн тохируулах талаар.
- Дискгүй машин дээр сүлжээгээр ачаалахыг хэрхэн тохируулах талаар.
- NFS root файлын систем ашиглан сүлжээгээр PXE ачаалалт хийхийг тохируулах талаар.
- Сүлжээний хаягийн хөрвүүлэлтийг хэрхэн тохируулах талаар.
- FreeBSD машин дээр IPv6-г хэрхэн тохируулах талаар.
- ATM-ийг хэрхэн тохируулах талаар.
- FreeBSD дээр Common Address Redundancy Protocol буюу CARP-ийн боломжуудыг хэрхэн идэвхжүүлж ашиглах талаар.

Энэ бүлгийг уншихаасаа өмнө та дараах зүйлсийг гүйцэтгэх хэрэгтэй:

- /etc/rc скриптүүдийн үндсүүдийг ойлгосон байх.
- Сүлжээний үндсэн ухагдахуудын талаар мэдлэгтэй байх.
- Шинэ FreeBSD цөм хэрхэн тохируулж суулгах талаар мэдэх (FreeBSD цөмийг тохируулах нь).
- Нэмэлт гуравдагч талуудын хийсэн програм хангамжийг хэрхэн суулгах талаар мэдэх (Програм суулгах. Багцууд болон портууд).

## 32.2. Гарцууд болон Чиглүүлэлтүүд

Сүлжээгээр нэг машин нөгөө машиныг олж чаддаг байхын тулд нэгээс нөгөө уруу хэрхэн хүрэхийг тайлбарласан арга замууд байх ёстой. Үүнийг *routing* буюу *чиглүүлэлт хийх* гэдэг. "Чиглүүлэлт" нь "destination буюу зорьсон газар" болон "гарц" хаягийн хослолоор тодорхойлогддог. Хэрэв та энэ "зорьсон газар" уруу очихоор оролдож байгаа бол энэ "гарц"аар холбогдоно гэж энэ хослол нь зааж байгаа юм. Гурван төрлийн зорьсон газар байдаг: эдгээр нь хостууд, дэд сүлжээнүүд болон "анхдагч" юм. "Анхдагч чиглүүлэлт" нь аль ч чиглүүлэлтэд хамаарахгүй бол ашиглагддаг. Бид анхдагч чиглүүлэлтийн талаар дараа нь арай дэлгэрэнгүй ярилцах болно. Бас гурван төрлийн гарц байдаг: эдгээр нь хостууд, интерфэйсүүд (бас "links" буюу холбоосууд гэгддэг) болон Ethernet тоног төхөөрөмжийн хаягууд (MAC хаягууд) юм.

### 32.2.1. Жишээ

Чиглүүлэлтийн өөр ойлголтуудыг үзүүлэхийн тулд бид `netstat`-ийн дараах жишээг ашиглах болно:

```
% netstat -r
Routing tables

Destination      Gateway            Flags     Refs     Use     Netif Expire

default          outside-gw         UGSc        37      418      ppp0
localhost        localhost          UH           0      181       lo0
test0            0:e0:b5:36:cf:4f   UHLW         5    63288       ed0     77
10.20.30.255     link#1             UHLW         1     2421
example.com      link#1             UC           0        0
host1            0:e0:a8:37:8:1e    UHLW         3     4601       lo0
host2            0:e0:a8:37:8:1e    UHLW         0        5       lo0 =>
host2.example.com link#1             UC          0        0
224              link#1             UC           0        0
```

Эхний хоёр мөр нь анхдагч чиглүүлэлт (бид үүнийг дараагийн хэсэгт авч үзэх болно) болон `localhost` чиглүүлэлтийг заана.

Энэ чиглүүлэлтийн хүснэгтийн заасан `localhost`-д зориулж ашиглах интерфэйс (`Netif` багана) нь lo0 бөгөөд энэ нь бас loopback буюу буцах интерфэйс гэгддэг. Энэ нь уг очих газрын хувьд бүх урсгалыг LAN уруу илгээхийн оронд дотооддоо үлдээнэ гэж хэлж байгаа бөгөөд энэ нь хаанаас эхэлсэн тэндээ буцаж очих учраас тэр юм.

Дараагийн байгаа зүйл бол `0:e0:`-с эхэлсэн хаягууд юм. Эдгээр нь Ethernet тоног төхөөрөмжийн хаягууд бөгөөд бас MAC хаягууд гэгддэг. FreeBSD нь локал Ethernet дээр байгаа ямар ч хостуудыг (жишээн дээрх `test0`) автоматаар таньж тэр хостод зориулж шууд түүн уруу ed0 Ethernet интерфэйсээр гарахаар чиглүүлэлт нэмдэг. Энэ төрлийн чиглүүлэлттэй холбоотой хугацаа (`Expire` багана) байдаг бөгөөд энэ нь заасан хугацаанд тэр хостоос бид нар юу ч сонсохгүй үед ашиглагддаг. Ийм явдал болоход энэ хост уруу заагдсан чиглүүлэлт автоматаар устгагдах болно. Эдгээр хостууд нь RIP (Routing Information Protocol буюу чиглүүлэлтийн мэдээллийн протокол) гэгддэг арга замаар танигддаг бөгөөд энэ нь хамгийн богино замыг тодорхойлсны үндсэн дээр локал хостууд уруу очих чиглүүлэлтийг олдог.

FreeBSD нь бас локал дэд сүлжээнд зориулж дэд сүлжээний чиглүүлэлтүүдийг нэмдэг (`10.20.30.255` нь `10.20.30` дэд сүлжээний цацах хаяг бөгөөд `example.com` нь тэр дэд сүлжээтэй холбоотой домэйний нэр юм). `link#1` тэмдэглэгээ нь машин дахь эхний Ethernet картыг заана. Та тэдгээрт зориулж ямар ч нэмэлт интерфэйс заагдаагүйг харах болно.

Эдгээр бүлгүүд (локал сүлжээний хостууд болон локал дэд сүлжээнүүд) нь автоматаар routed гэгддэг дэмоноор тохируулагдсан чиглүүлэлтүүдтэй байна. Хэрэв энэ нь ажиллахгүй байгаа бол зөвхөн статикаар тодорхойлогдсон (өөрөө хэлбэл илэрхий оруулж өгсөн) чиглүүлэлтүүд байх болно.

`host1` мөр нь бидний хостыг зааж байгаа бөгөөд түүнийг Ethernet хаягаар нь мэддэг. Бид илгээж байгаа хост болохоор FreeBSD нь Ethernet интерфэйсээр илгээхийн оронд loopback буюу буцах интерфэйсийг (lo0) ашиглахаа мэддэг.

Хоёр `host2` мөрүүд нь биднийг ifconfig(8) alias буюу өөр нэр ашиглах үед (Ethernet-ийн талаар дурдсан хэсгээс бид үүнийг яагаад хийдгийг үзээрэй) учирч болзошгүй зүйлийн жишээ юм. lo0 интерфэйсийн дараа байгаа ⇒ тэмдэг нь бид зөвхөн loopback буюу буцах интерфэйсийг ашиглаад зогсохгүй (энэ хаяг нь бас локал хостыг хэлж байгаа болохоор) энэ нь ялангуяа alias буюу өөр нэр гэдгийг хэлж байгаа юм. Ийм чиглүүлэлтүүд нь alias-ийг дэмждэг хост дээр зөвхөн харагдана; локал сүлжээн дэх бусад бүх хостууд ийм чиглүүлэлтүүдийнхээ хувьд ердөө л `link#1` мөртэй байна.

Төгсгөлийн мөр (очих дэд сүлжээ `224`) нь multicast-тай ажиллаж байгаа бөгөөд үүнийг өөр хэсэгт авч үзэх болно.

Төгсгөлд нь чиглүүлэлт бүрийн төрөл бүрийн шинж чанаруудыг `Flags` баганаас харж болно. Эдгээр тугуудын зарим болон тэдгээрийн харгалзах утга санаануудыг доорх богино хүснэгтээр харуулав:

| | |
|---|---|
| U | Up: Чиглүүлэлт идэвхтэй байна. |
| H | Host: Чиглүүлэлтийн очих газар нь ганц хост байна. |
| G | Gateway: Энэ очих газарт зориулж бүгдийг энэ алсын систем уруу илгээх бөгөөд алсын систем нь тэндээсээ хаашаа илгээхээ олох болно. |
| S | Static: Энэ чиглүүлэлт нь системээр автоматаар үүсгэгдсэн биш гараар тохируулагдсан. |
| C | Clone: Бидний холбогдож байгаа машины хувьд энэ чиглүүлэлт дээр үндэслэн шинэ чиглүүлэлт үүсгэдэг. Энэ төрлийн чиглүүлэлт нь ихэвчлэн локал сүлжээнүүдэд ашиглагддаг. |
| W | WasCloned: Локал сүлжээний (Clone) чиглүүлэлт дээр үндэслэн автоматаар тохируулагдсан чиглүүлэлтийг заана. |
| L | Link: Чиглүүлэлт нь Ethernet тоног төхөөрөмжтэй холбоотой зүйлийг хамарна. |

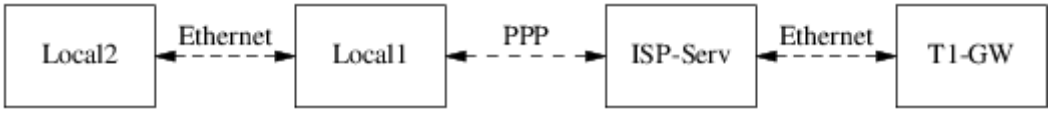
### 32.2.2. Анхдагч чиглүүлэлтүүд

Локал систем нь алсын хост уруу холболт хийх хэрэгтэй үед мэдэгдэж байгаа зам байгаа эсэхийг тодорхойлохын тулд энэ нь чиглүүлэлтийн хүснэгтээс шалгадаг. Хэрэв алсын хост нь бидний хэрхэн хүрэхийг нь мэдэх (клон хийгдсэн чиглүүлэлтүүд) дэд сүлжээнд байгаа бол систем тэр интерфэйсээр холбогдож болох эсэхийг шалгадаг.

Хэрэв бүх мэдэгдэж байгаа зам амжилттүй болвол системд сүүлчийн ганц сонголт "анхдагч" чиглүүлэлт үлдэнэ. Энэ чиглүүлэлт нь гарц чиглүүлэлтийн тусгай төрөл (системд ихэвчлэн ганц байдаг) бөгөөд тугнуудын талбартаа үргэлж `c` гэж тэмдэглэгдсэн байдаг. Локал сүлжээн дэх хостын хувьд энэ гарц нь машины гадаад ертөнц уруу шууд холбогдох зүйлээр (PPP холболт, DSL, кабель модем, T1, эсвэл өөр сүлжээний интерфэйсээр) тохируулагддаг.

Хэрэв та өөрөө гаднах ертөнц уруу гарц маягаар ажиллаж байгаа машины хувьд анхдагч чиглүүлэлтийг тохируулж байгаа бол анхдагч чиглүүлэлт нь таны Интернэтийн Үйлчилгээ Үзүүлэгчийн (ISP) сайт дахь гарц машин болох юм.

Анхдагч чиглүүлэлтүүдийн жишээг харцгаая. Энэ нь нийтлэг тохиргоо юм:




`Local1` болон `Local2` хостууд нь таны сайтад байна. `Local1` нь ISP уруу dial-up PPP холболтоор холбогдсон. Энэ PPP сервер компьютер нь дотоод сүлжээгээр өөр нэг гарц компьютер уруу гадаад интерфэйсээр ISP-ийн Интернэт өгч байгаа уруу холбогдсон байна.

Таны машин бүрийн хувьд анхдагч чиглүүлэлтүүд нь ийм болно:

| **Хост** | **Анхдагч Гарц** | **Интерфэйс** |
|---|---|---|
| Local2 | Local1 | Ethernet |
| Local1 | T1-GW | PPP |

Түгээмэл асуулт нь "Бид яагаад `Local1`-ийн хувьд түүний холбогдсон ISP-ийн серверийн оронд `T1-GW`-г анхдагч гарц гэж тохируулсан бэ?" гэсэн асуулт юм.

PPP интерфэйс нь өөрийн талын холболтондоо ISP-ийн локал сүлжээн дэх хаягийг ашиглаж байгаа болохоор ISP-ийн локал сүлжээн дэх бусад дурын машинуудад зориулсан чиглүүлэлт автоматаар үүсгэгдэх болно гэдгийг санаарай. Ийм учраас та `T1-GW` машин уруу хэрхэн хүрэхээ аль хэдийн мэдэж байгаа болохоор ISP-ийн сервер уруу урсгал илгээх дундын алхам шаардлагагүй юм.

Өөрийн локал сүлжээний хувьд `X.X.X.1` хаягийг гарц хаяг болгож ашиглах нь түгээмэл байдаг. Тэгэхээр (адил жишээг ашиглаад) хэрэв таны локал С ангиллын хаягийн талбар `10.20.30` байсан бөгөөд таны ISP `10.9.9`-г ашиглаж байгаа бол анхдагч чиглүүлэлтүүд нь ийм байна:

| **Хост** | **Анхдагч Чиглүүлэлт** |
|---|---|
| Local2 (10.20.30.2) | Local1 (10.20.30.1) |
| Local1 (10.20.30.1, 10.9.9.30) | T1-GW (10.9.9.1) |

Та /etc/rc.conf файлын тусламжтай анхдагч чиглүүлэлтийг хялбараар тодорхойлж болно. Бидний жишээн дээр `Local2` машин дээр бид дараах мөрийг /etc/rc.conf файлд нэмсэн:

```
defaultrouter="10.20.30.1"
```

Үүнийг шууд тушаалын мөрөөс route(8) тушаалаар хийж бас болно:

```
# route add default 10.20.30.1
```

Сүлжээний чиглүүлэлтийн хүснэгтүүдийг гараар удирдах талаар дэлгэрэнгүй мэдээллийг route(8) гарын авлагын хуудаснаас лавлана уу.

### 32.2.3. Хос гэртэй хостууд

Өөр бас нэг тохиргооны төрлийг бид хэлэлцэх ёстой бөгөөд энэ нь хоёр өөр сүлжээн дээр байгаа хостын тухай юм. Техникийн хувьд гарц болж байгаа дурын машин (дээрх жишээн дээр PPP холболтыг ашиглах нь) хос гэртэй хост гэж тооцогддог. Гэхдээ үнэндээ энэ ухагдахуун нь хоёр дотоод сүлжээнд байгаа машиныг хэлэхэд зөвхөн ашиглагддаг.

Нэг тохиолдолд машин нь хоёр Ethernet карттай бөгөөд карт бүр тусдаа дэд сүлжээний хаягтай байна. Өөрөөр, машин нь зөвхөн нэг Ethernet карттай бөгөөд ifconfig(8) alias хийлт буюу өөр нэрийг ашигладаг байж болно. Эхнийх нь физикийн хувьд тусдаа хоёр Ethernet сүлжээ ашиглагдаж байгаа үед хэрэглэгдэх бөгөөд сүүлийнх нь ганц физик сүлжээний сегмент байгаа боловч логикийн хувьд хоёр тусдаа дэд сүлжээнүүд байгаа үед хэрэглэгддэг.

Аль ч тохиолдолд энэ машин нь нөгөө дэд сүлжээ уруу хүрэх тодорхойлогдсон гарц (гаднаас ирэх чиглүүлэлт) гэдгийг дэд сүлжээ бүр мэддэг байхаар чиглүүлэлтийн хүснэгтүүд тохируулагддаг. Хоёр дэд сүлжээний хооронд чиглүүлэгч маягаар ажиллах машинтай энэ тохиргоо нь бид аль нэг тал уруу эсвэл хоёр талын хоёулангийн хувьд пакет шүүгч буюу галт хана дээр тулгуурласан аюулгүй байдлын шийдлийг хийх шаардлагатай үед ихэвчлэн хэрэглэгддэг.

Хоёр интерфэйсийн хооронд пакетуудыг дамжуулдаг байхаар энэ машиныг байлгахыг хэрэв та хүсвэл энэ боломжийг идэвхжүүлэхийг FreeBSD-д хэлэх хэрэгтэй. Үүнийг хэрхэн хийх талаар дэлгэрэнгүйг дараагийн хэсгээс үзнэ үү.

### 32.2.4. Чиглүүлэгч бүтээх нь

Сүлжээний чиглүүлэгч нь ердөө л нэг интерфэйсээс нөгөө интерфэйс уруу пакетуудыг дамжуулдаг систем юм. Интернэтийн стандартууд болон сайн инженерчлэлийн туршлага нь үүнийг FreeBSD дээр анхдагчаар идэвхжүүлсэн байхаас FreeBSD төслийг сэргийлдэг. rc.conf(5) дахь дараах хувьсагчийг `YES` болгон өөрчилснөөр та энэ боломжийг идэвхжүүлж болно:

```
gateway_enable="YES"          # Set to YES if this host will be a gateway
```

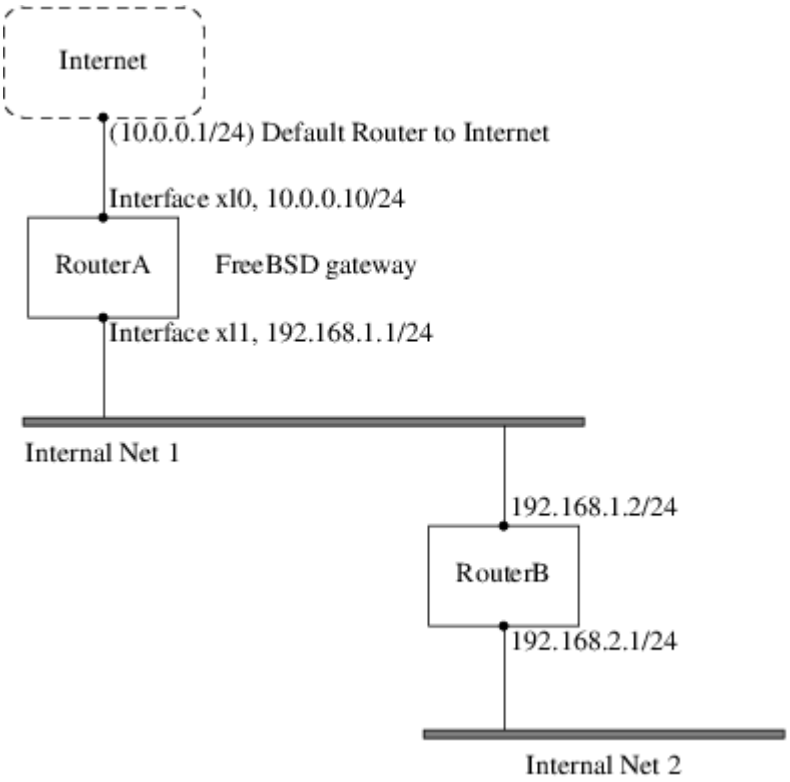
Энэ тохируулга нь sysctl(8)-ийн хувьсагч `net.inet.ip.forwarding`-г `1` болгоно. Хэрэв та чиглүүлэлтийг түр зуур зогсоох хэрэгтэй бол үүнийг түр зуур `0` болгож тохируулж болно.

Урсгалыг хааш нь илгээхээ мэдэхийн тулд таны шинэ чиглүүлэгчид чиглүүлэлтүүд хэрэгтэй болно. Хэрэв таны сүлжээ хангалттай хялбар байх юм бол статик чиглүүлэлтүүд ашиглаж болно. FreeBSD нь стандарт BSD чиглүүлэлтийн дэмон routed(8) програмтай бас ирдэг. Энэ нь RIP (1 ба 2-р хувилбар) болон IRDP протоколуудыг дэмждэг. BGP v4, OSPF v2 болон бусад төвөгтэй чиглүүлэлтийн протоколуудын дэмжлэгийг net/zebra багц хангадаг. GateD® зэрэг арилжааны бүтээгдэхүүнүүд бас илүү төвөгтэй сүлжээний чиглүүлэлтийн шийдлүүдэд зориулагдсан байдаг.

## 32.2.5. Статик чиглүүлэлтүүдийг тохируулах нь

### 32.2.5.1. Гар тохиргоо

Бид дараах сүлжээтэй байна гэж төсөөлье:




Энэ тохиолдолд `RouterA` нь Интернэт уруу гарах чиглүүлэгч маягаар ажиллаж байгаа бидний FreeBSD машин байна. Энэ нь анхдагч чиглүүлэлтээ `10.0.0.1` гэж тохируулсан бөгөөд ингэснээр гаднах ертөнц уруу холбогдох боломжийг олгоно. Бид `RouterB` нь зөв тохируулагдсан бөгөөд хаа явах хэрэгцээтэй газраа хэрхэн хүрэхээ мэддэг гэж үзэх болно. (Энэ нь зураг дээр хялбар юм. Ердөө л `RouterB` дээр `192.168.1.1`-г гарц болгон анхдагч чиглүүлэлт нэмнэ.)

Хэрэв бид `RouterA`-ийн чиглүүлэлтийн хүснэгтийг харвал доор дурдсантай төстэйг харах болно:

```
% netstat -nr
Routing tables

Internet:
Destination        Gateway            Flags    Refs      Use  Netif  Expire
```

```
default            10.0.0.1           UGS         0    49378    xl0
127.0.0.1          127.0.0.1          UH          0        6    lo0
10.0.0.0/24          link#1             UC          0        0    xl0
192.168.1.0/24       link#2             UC          0        0    xl1
```

Энэ үеийн чиглүүлэлтийн хүснэгттэй байхад RouterA нь бидний Internal Net 2 буюу дотоод сүлжээ 2 уруу хүрч чадахгүй байх болно. Энэ нь 192.168.2.0/24-ийн хувьд чиглүүлэлтгүй байна. Үүнийг арилгах нэг арга нь гараар чиглүүлэлт нэмэх явдал юм. Дараах тушаал нь RouterA-ийн чиглүүлэлтийн хүснэгтэд 192.168.1.2-г дараагийн зөрлөг (hop) болгон ашиглан Internal Net 2 сүлжээг нэмэх болно:

```
# route add -net 192.168.2.0/24 192.168.1.2
```

Одоо RouterA нь 192.168.2.0/24 сүлжээн дэх дурын хост уруу хүрч чадна.

#### 32.2.5.2. Байнгын тохиргоо

Дээрх жишээ нь ажиллаж байгаа систем дээр статик чиглүүлэлтийг тохируулахад төгс төгөлдөр юм. Гэхдээ нэг асуудал нь таны FreeBSD машин дахин ачаалахад чиглүүлэлтийн мэдээлэл үлдэхгүй байх явдал юм. Статик чиглүүлэлтийг байнга байлгахын тулд түүнийг /etc/rc.conf файлд нэмнэ:

```
# Add Internal Net 2 as a static route
static_routes="internalnet2"
route_internalnet2="-net 192.168.2.0/24 192.168.1.2"
```

static_routes тохиргооны хувьсагч нь зайгаар тусгаарлагдсан үгнүүдийн жагсаалт юм. Үг бүр чиглүүлэлтийн нэрийг заана. Бидний жишээн дээр static_routes мөрөнд бид зөвхөн нэг үгтэй байна. Энэ нь *internalnet2* юм. Бид дараа нь route(8) тушаалд өгөх бүх тохиргооны нэмэлт өгөгдлүүдийг route_internalnet2 гэгдсэн тохиргооны хувьсагчийн хамтаар нэмнэ. Бидний жишээний хувьд бид ийм тушаал ашиглаж болно:

```
# route add -net 192.168.2.0/24 192.168.1.2
```

тэгэхээр бидэнд "-net 192.168.2.0/24 192.168.1.2" хэрэгтэй.

Дээр дурдсанаар бид static_routes мөрөнд нэгээс илүү үгс оруулж болно. Энэ нь бидэнд олон статик чиглүүлэлтүүд үүсгэх боломжийг олгоно. Дараах мөрүүд нь 192.168.0.0/24 болон 192.168.1.0/24 сүлжээний хувьд статик чиглүүлэлтүүдийг санаандаа төсөөлсөн чиглүүлэгч дээрээ нэмж байгаа жишээг үзүүлж байна:

```
static_routes="net1 net2"
route_net1="-net 192.168.0.0/24 192.168.0.1"
route_net2="-net 192.168.1.0/24 192.168.1.1"
```

### 32.2.6. Чиглүүлэлтийн тархалт

Бид гаднах ертөнц уруу чиглэсэн өөрсдийн чиглүүлэлтүүдийг хэрхэн тодорхойлох талаар ярилцсан боловч гаднах ертөнц хэрхэн биднийг олох талаар хэлэлцээгүй билээ.

Чиглүүлэлтийн хүснэгтүүдийг тухайн нэг хаягийн талбарын (бидний жишээн дээр C ангиллын дэд сүлжээ) бүх урсгал тэр сүлжээний тухайн нэг хост уруу илгээгдэж тэр нь пакетуудыг дотогшоо дамжуулдгаар тохируулж болно гэдгийг бид мэдэж байгаа билээ.

Та өөрийн сайтдаа заагдсан хаягийн талбарыг авах үед таны үйлчилгээ үзүүлэгч өөрсдийн чиглүүлэлтийн хүснэгтүүдийг таны дэд сүлжээнд зориулагдсан бүх урсгал таны PPP холбоосоор таны сайт уруу илгээгддэгээр тохируулах болно. Гэхдээ улс даяар байх бусад сайтууд таны ISP уруу илгээхээ яаж мэдэх вэ?

Бүх заагдсан хаягийн талбаруудыг хянаж Интернэтийн үндсэн сүлжээ уруу холбогдох холболтын цэгийг тодорхойлдог систем (тараагдсан DNS-ийн мэдээлэлтэй бараг адил) байдаг. "Backbone" буюу "гол нуруу (үндсэн сүлжээ)" нь улс даяар болон дэлхий даяар Интернэтийн урсгалыг зөөвөрлөж байдаг гол шугамнууд юм. Гол нуруу машин бүр мастер хүснэгтүүдийн хуулбартай байдаг бөгөөд эдгээр хүснэгтүүд нь тухайн нэг сүлжээнд зориулсан урсгалыг онцгой гол нуруу зөөвөрлөгч уруу чиглүүлж тэндээсээ доош бусад үйлчилгээ үзүүлэгчдээр дамжин таны сүлжээнд хүрдэг.

Үндсэн (гол нуруу) сайтууд уруу зарлаж тэдгээр нь таны сайтын хувьд холболтын цэг (буюу орох зам) гэж тодорхойлох нь таны үйлчилгээ үзүүлэгчийн үүрэг юм. Үүнийг чиглүүлэлтийн тархалт гэж нэрлэдэг.

### 32.2.7. Алдааг олж засварлах

Заримдаа чиглүүлэлтийн тархалт асуудалтай бөгөөд зарим нэгэн сайтууд өөр уруу чинь холбогдож чадахгүй байж болно. Чиглүүлэлт хаана ажиллахгүй болж байгааг олоход тус болох магадгүй хамгийн ашигтай тушаал нь traceroute(8) тушаал юм. Хэрэв та алсын машин уруу холбогдож чадахгүй юм шиг байвал (өөрөөр хэлбэл ping(8) амжилтгүй болвол) энэ нь бас адил ашигтай байдаг.

traceroute(8) тушаалыг таны холбогдохыг оролдож байгаа алсын хостын нэртэй ажиллуулдаг. Энэ нь гарц хостуудыг оролдлого хийж байгаа замын хамтаар харуулдаг бөгөөд эцсийн хост уруу хүрэх юм уу эсвэл холболтын асуудлаас болоод эцсийн хост уруугаа хүрэлгүйгээр ажиллаж дуусдаг.

Дэлгэрэнгүй мэдээллийг traceroute(8)-ийн гарын авлагын хуудаснаас үзнэ үү.

### 32.2.8. Multicast чиглүүлэлт хийх

FreeBSD нь multicast програмууд болон multicast чиглүүлэлтийг угаасаа дэмждэг. Multicast програмууд нь FreeBSD-ийн ямар нэг тусгай тохиргоо шаарддаггүй; програмууд нь шууд л ажилладаг. Multicast чиглүүлэлт нь дэмжлэгийг цөмд эмхэтгэхийг шаарддаг:

```
options MROUTING
```

Мөн multicast чиглүүлэлтийн дэмон mrouted(8) нь туннелиуд болон DVMRP-г үүсгэхээр /etc/mrouted.conf файлын тусламжтай тохируулагдсан байх ёстой. Multicast тохиргооны талаар дэлгэрэнгүй мэдээллийг mrouted(8)-ийн гарын авлагын хуудаснаас олж болно.

mrouted(8) multicast чиглүүлэлтийн демон нь DVMRP multicast чиглүүлэлтийн протоколыг хэрэгжүүлдэг бөгөөд үүнийг ихэнх multicast суулгацуудад pim(4)-р ихэвчлэн сольсон байдаг. Холбогдох map-mbone(8) болон mrinfo(8) хэрэгслүүд нь FreeBSD-ийн портын цуглуулгад net/mrouted портод байгаа болно.

# 32.3. Утасгүй сүлжээ

## 32.3.1. Утасгүй сүлжээний үндсүүд

Ихэнх утасгүй сүлжээнүүд нь IEEE® 802.11-ийн стандартууд дээр үндэслэсэн байдаг. Үндсэн утасгүй сүлжээ нь 2.4GHz юм уу эсвэл 5GHz-ийн зурваст цацдаг радио долгионоор холбогддог олон станцуудаас тогтдог (энэ нь орон нутгаас хамаарч өөр өөр байдаг бөгөөд 2.3GHz-с 4.9GHz-ийн хүрээнд холбоог идэвхжүүлэхээр өөрчлөгдөж байгаа).

802.11 сүлжээ нь хоёр аргаар зохион байгуулагддаг: *дэд бүтцийн горим*д нэг станц мастер маягаар ажиллаж бусад станцууд нь түүнтэй холбогддог; энэ сүлжээг BSS гэдэг бөгөөд мастер станц нь хандалтын цэг (access point)(AP) гэгддэг. BSS-т бүх холбоо AP-р дамждаг; нэг станц өөр нэг утасгүй станцтай холбоо тогтоохыг хүссэн байсан ч гэсэн мэдэгдлүүд нь AP-р дамжих ёстой байдаг. Хоёр дахь хэлбэрийн сүлжээнд мастер байдаггүй бөгөөд станцууд нь шууд холбогддог. Энэ сүлжээний хэлбэрийг IBSS гэдэг бөгөөд бас *ad-hoc сүлжээ* гэгддэг.

802.11 сүлжээнүүд нь 2.4GHz-ийн зурваст IEEE® 802.11 болон 802.11b стандартуудаар тодорхойлогдсон протоколуудыг ашиглан эхлэн хийгдэж байсан. Эдгээр тодорхойлолтууд нь ажиллах давтамжууд, хүрээ хийх болон дамжуулах хурд (холболт төрөл бүрийн хурдаар хийгдэж болно) зэрэг MAC давхаргын үзүүлэлтүүдийг агуулдаг. Сүүлд 802.11a стандарт нь өөр дохионы арга замууд болон илүү өндөр дамжуулах хурдууд зэрэг 5GHz-ийн зурвасын ажиллагааг тодорхойлсон билээ. Мөн дараа нь 802.11b сүлжээнүүдтэй буцаж нийцтэй байхаар 802.11a дохионы хэрэглээ болон 2.4GHz-ийн зурваст дамжуулах арга замуудыг идэвхжүүлж 802.11g стандарт тодорхойлогдсон байна.

Суурь болсон дамжуулах техникүүдээс гадна 802.11 сүлжээнүүд нь төрөл бүрийн аюулгүй байдлын арга замуудтай байдаг. Анхдагч 802.11 тодорхойлолтууд нь WEP гэгддэг аюулгүй байдлын энгийн протоколыг тодорхойлсон билээ. Энэ протокол нь тогтмол, урьдчилан хуваалцсан түлхүүр болон RC4 криптограф шифр ашиглан сүлжээнд дамжуулагдах өгөгдлийг кодчилдог. Станцууд нь өөр хоорондоо холбогдохын тулд бүгд нэг тогтмол түлхүүр дээр зөвшилцөх ёстой. Энэ схемийг амархан эвдэх боломжтойг харуулсан бөгөөд дамжин өнгөрөх хэрэглэгчдийг сүлжээнд нэгдэхийг зөвлөдөггүйгээс бусад тохиолдолд одоо ховор ашиглагдах болсон байна. Криптографийн шинэ шифрүүд болон станцуудыг хандалтын цэгт танин нэвтрүүлж өгөгдлийн холболтыг хийхийн тулд түлхүүрүүдийг солилцох нэмэлт протоколыг тодорхойлдог IEEE® 802.11i тодорхойлолтоор одоогийн аюулгүй байдлын практикийг тодорхойлдог. Мөн криптограф түлхүүрүүд нь үе үе шинэчлэгддэг бөгөөд халдлагын оролдлогуудыг илрүүлэх (ба халдлагын оролдлогуудыг сөрөх) арга замууд бас байдаг. Утасгүй сүлжээнүүдэд түгээмэл ашиглагддаг өөр нэг аюулгүй байдлын протоколын

тодорхойлолт нь WPA юм. Энэ нь үйлдвэрлэлийн бүлгийн тодорхойлсон 802.11i-ийн өмнөх протокол бөгөөд 802.11i-г батлахыг хүлээж байх хугацаанд түр зуурын арга хэмжээ болох юм. WPA нь 802.11i-д байгаа шаардлагуудын дэд олонлогийг тодорхойлж хуучин тоног төхөөрөмжүүд дээрх шийдэлд зориулагдан хийгджээ. Ялангуяа WPA нь анхдагч WEP шифрээс гарсан TKIP шифрийг зөвхөн шаарддаг. 802.11i нь TKIP-ийн хэрэглээг зөвшөөрдөг боловч өгөгдлийг шифрлэхэд зориулж илүү хүчирхэг шифр болох AES-CCM-ийн дэмжлэгийг бас шаарддаг. (AES шифр нь WPA-д шаардагддаггүй, учир нь үүнийг хуучин тоног төхөөрөмж дээр хийхэд тооцооллын хувьд өртөгтэй гэж үздэг.)

Дээрх протоколын стандартуудаас гадна мэдэж байх өөр нэг чухал стандарт бол 802.11e юм. Энэ нь 802.11 сүлжээнд видео дамжуулах, IP дээгүүрх дуу (VoIP) зэрэг мультимедиа хэрэглээнүүдэд шаардлагатай протоколуудыг тодорхойлдог. 802.11i-тай адил 802.11e нь үйлдвэрлэлийн бүлгээс 802.11e-ийн дэд олонлог гэж тодорхойлсон, 802.11e-г батлахыг хүлээж байх хугацаанд мультимедиа хэрэглээнүүдийг идэвхжүүлэхэд ашиглаж болох WME (сүүлд WMM) гэгдсэн урьдчилсан тодорхойлолт байдаг. 802.11e болон WME/WMM-ийн талаар мэдэх ёстой хамгийн чухал зүйл нь утасгүй сүлжээний зэрэглэл тогтоогдсон урсгалын хэрэглээг Quality of Service (QoS) буюу үйлчилгээний чанарын протоколууд болон өргөтгөсөн зөөвөрлөгчийн хандалтын протоколуудын тусламжтайгаар идэвхжүүлдэг явдал юм. Эдгээр протоколуудын зөв шийдэл нь өгөгдлийн өндөр хурдтай тэсрэлт болон зэрэглэл тогтоогдсон урсгалыг идэвхжүүлдэг.

FreeBSD нь 802.11a, 802.11b, болон 802.11g-г ашиглан ажилладаг сүлжээнүүдийг дэмждэг. WPA болон 802.11i аюулгүй байдлын протоколууд нь адилхан (11a, 11b, болон 11g-тэй цуг) дэмжигдсэн байдаг бөгөөд WME/WMM протоколуудын шаарддаг QoS болон урсгалын зэрэглэлт нь хязгааргүй тооны утасгүй төхөөрөмжүүдэд дэмжигдсэн байдаг.

## 32.3.2. Үндсэн тохируулга

### 32.3.2.1. Цөмийн тохиргоо

Утасгүй сүлжээг ашиглахын тулд танд утасгүй сүлжээний карт болон цөмийг тохирох утасгүй сүлжээний дэмжлэгтэйгээр тохируулах хэрэгтэй болно. Сүүлийнх нь олон модулиудад хуваагддаг. Тэгэхээр танд зөвхөн өөрийнхөө ашиглах програм хангамжийг тохируулах хэрэгтэй болох юм.

Танд эхлээд хэрэг болох зүйл нь утасгүй сүлжээний төхөөрөмж юм. Хамгийн түгээмэл хэрэглэгддэг төхөөрөмжүүд нь Atheros-ийн хийдэг хэсгүүдийг ашигладаг төхөөрөмжүүд байдаг. Эдгээр төхөөрөмжүүд нь ath(4) драйвераар дэмжигддэг бөгөөд /boot/loader.conf файлд дараах мөрийг нэмэхийг шаарддаг:

```
if_ath_load="YES"
```

Atheros драйвер нь гурван тусдаа хэсэгт хуваагддаг: эдгээр нь зөв драйвер (ath(4)), бичил схемийн функцуудыг зохицуулдаг, тоног төхөөрөмжийн дэмжлэгийн давхарга (ath_hal(4)), болон хүрээнүүдийг дамжуулахын тулд хэд хэдэн боломжит хурдуудаас (энд ath_rate_sample) сонгох алгоритм юм. Эдгээр дэмжлэгийг модуль хэлбэрээр дуудах үед түүнтэй хамааралтай зүйлс автоматаар зохицуулагдах болно. Хэрэв танд Atheros төхөөрөмжийн оронд өөр төхөөрөмж байгаа бол та тэр төхөөрөмжид зориулсан модулийг

сонгох юм, өөрөөр хэлбэл:

```
if_wi_load="YES"
```

гэж Intersil Prism хэсгүүд дээр суурилсан төхөөрөмжүүдийн хувьд хийж өгнө (wi(4) driver).

Энэ баримтын үлдсэн хэсэгт бид ath(4) төхөөрөмжийг ашиглах бөгөөд жишээнүүд дээрх төхөөрөмжийн нэр таны тохиргооны дагуу өөрчлөгдөх ёстой. Байгаа утасгүй драйверууд болон дэмжигдсэн адаптеруудын жагсаалтыг FreeBSD Hardware Notes хуудаснаас олж болно. Төрөл бүрийн хувилбарууд болон архитектуруудад зориулсан эдгээр тэмдэглэгээнүүдийн хуулбар FreeBSD-ийн вэб сайтын Хувилбарын Мэдээлэл хуудсанд бий. Таны утасгүй төхөөрөмжид зориулагдсан FreeBSD-ийн драйвер байхгүй бол магадгүй Windows® драйверийг NDIS драйверийн гүйцэтгэл хялбаршуулагчийн тусламжтай шууд ашиглаж болох юм.

Үүнтэй цуг таны ашиглах аюулгүй байдлын протоколуудад зориулагдсан криптографын дэмжлэгийг хийдэг модулиуд хэрэгтэй болно. Эдгээр нь wlan(4) модулиар автоматаар шаардлагын дагуу дуудагддаг байхаар зориулагдсан боловч одоохондоо эдгээрийг гараар тохируулах шаардлагатай. Дараах модулиуд байдаг: wlan_wep(4), wlan_ccmp(4) болон wlan_tkip(4). wlan_ccmp(4) болон wlan_tkip(4) драйверууд нь WPA ба/эсвэл 802.11i аюулгүй байдлын протоколуудыг ашиглахаар болсон тохиолдолд танд зөвхөн хэрэгтэй байдаг. Хэрэв таны сүлжээ шифрлэлт ашигладаггүй бол танд wlan_wep(4) дэмжлэг хэрэггүй. Эдгээр модулиудыг ачаалах үед дуудахын тулд дараах мөрүүдийг /boot/loader.conf файлд нэмнэ:

```
wlan_wep_load="YES"
wlan_ccmp_load="YES"
wlan_tkip_load="YES"
```

Системийн эхлүүлэх тохиргооны файлд (өөрөөр хэлбэл /boot/loader.conf) ийм мэдээлэлтэйгээр та өөрийн FreeBSD хайрцгийг дахин ачаалах хэрэгтэй. Хэрэв та өөрийн машиныг дахин ачаалахыг хүсэхгүй байгаа бол та модулиудыг kldload(8)-ийн тусламжтайгаар гараар дуудаж болно.

Хэрэв та модулиудыг ашиглахыг хүсэхгүй байгаа бол дараах мөрүүдийг өөрийн цөмийн тохиргооны файлд нэмж эдгээр драйверуудыг цөмд эмхэтгэж болно:

```
device wlan                # 802.11 support
device wlan_wep           # 802.11 WEP support
device wlan_ccmp          # 802.11 CCMP support
device wlan_tkip          # 802.11 TKIP support
device wlan_amrr          # AMRR transmit rate control algorithm
device ath                # Atheros pci/cardbus NIC's
device ath_hal            # pci/cardbus chip support
options AH_SUPPORT_AR5416 # enable AR5416 tx/rx descriptors
```

```
device ath_rate_sample   # SampleRate tx rate control for ath
```

Цөмийн тохиргооны файлдаа ийм мэдээлэлтэйгээр цөмөө дахин эмхэтгээд өөрийн FreeBSD машиныг дахин ачаалах хэрэгтэй.

Систем ассаны дараа ачаалах үеийн мэдэгдлүүдэд үүнтэй адил утасгүй төхөөрөмжийн талаар зарим мэдээллийг олж болно:

```
ath0: <Atheros 5212> mem 0x88000000-0x8800ffff irq 11 at device 0.0 on cardbus1
ath0: [ITHREAD]
ath0: AR2413 mac 7.9 RF2413 phy 4.5
```

## 32.3.3. Дэд бүтцийн горим

Дэд бүтцийн горим буюу BSS горим нь ихэвчлэн ашиглагддаг горим юм. Энэ горимд утасгүй хандалтын цэгүүд нь утастай сүлжээнд холбогдсон байдаг. Утасгүй сүлжээ бүр өөрийн нэртэй байдаг бөгөөд энэ нэр нь сүлжээний SSID гэгддэг. Утасгүй клиентүүд нь утасгүй хандалтын цэгүүдэд холбогддог.

### 32.3.3.1. FreeBSD клиентүүд

#### 32.3.3.1.1. Хандалтын цэгүүдийг хэрхэн олох вэ

Сүлжээнүүдийг хайхын тулд `ifconfig` тушаалыг ашиглана. Энэ хүсэлт нь хэсэг хором болж болох бөгөөд боломжтой утасгүй давтамж бүр уруу шилжиж, байгаа хандалтын цэгүүдийг шалгахыг системээс шаарддаг. Зөвхөн супер хэрэглэгч ийм хайлт эхлүүлж чадна:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 up scan
SSID/MESH ID        BSSID              CHAN RATE    S:N   INT CAPS
dlinkap         00:13:46:49:41:76   11   54M -90:96   100 EPS  WPA WME
freebsdap       00:11:95:c3:0d:ac    1   54M -83:96   100 EPS  WPA
```

Та хайлт хийхээсээ өмнө интерфэйсээ `up` болгож тэмдэглэх ёстой. Дараа дараагийн хайлтын хүсэлтүүд интерфэйсийг up болгож тэмдэглэхийг шаарддаггүй.

Хайлтын хүсэлтийн гаралт олсон BSS/IBSS сүлжээ бүрийг харуулдаг. Сүлжээний нэр `SSID`-с гадна хандалтын цэгийн MAC хаяг болох `BSSID`-г бид харах болно. `CAPS` талбар нь сүлжээ бүрийн төрөл болон тэнд ажиллаж байгаа станцуудын боломжуудыг заана:

*Хүснэгт 13. Station Capability Codes*

| Capability Code | Meaning |
| --- | --- |
| E | Extended Service Set (ESS) буюу өргөтгөсөн үйлчилгээний олонлог. Станц нь дэд бүтцийн сүлжээний хэсэг гэдгийг харуулна (IBSS/ad-hoc сүлжээтэй харьцуулах юм бол ). |
| I | IBSS/ad-hoc сүлжээ. Станц нь ad-hoc сүлжээний хэсэг гэдгийг харуулна (ESS сүлжээтэй харьцуулах юм бол). |
| P | Хувийн нууц. BSS-ийн дотор солилцож байгаа өгөгдлийн бүх хүрээнүүдэд өгөгдлийн нууц байдал шаардлагатай байдаг. Энэ BSS нь бусадтай солилцох өгөгдлийн хүрээнүүдийг шифрлэх/буцаах WEP, TKIP эсвэл AES-CCMP зэрэг криптограф ашиглахыг станцаас шаарддаг гэсэн үг юм. |
| S | Богино оршил (preamble). Сүлжээ богино оршлуудыг (802.11b өндөр хурд/DSSS PHY-д тодорхойлогдсоноор богино оршил нь урт оршилд ашиглагддаг 128 бит талбартай харьцуулах юм бол 56 бит sync талбарыг хэрэглэдэг) ашиглаж байгааг харуулдаг. |
| s | Богино слот (ангархай) хугацаа. 802.11g сүлжээ хуучин (802.11b) станцууд байхгүй байгаа учраас богино слот хугацааг хэрэглэж байгааг харуулна. |

Мэдэгдэж байгаа сүлжээнүүдийн одоогийн жагсаалтыг бас доорх тушаалаар харуулж болно:

```
# ifconfig wlan0 list scan
```

Энэ мэдээллийг автоматаар хувиргагчаар (adapter) эсвэл гараар `scan` хүсэлтийн тусламжтай шинэчилж болно. Хуучин өгөгдөл кэшээс автоматаар арилгагдах болохоор хэсэг хугацаа өнгөрсний дараа нэмэлт хайлтууд хийхгүй л бол энэ жагсаалт багасаж болох юм.

#### 32.3.3.1.2. Үндсэн тохиргоонууд

Энэ хэсэгт FreeBSD дээр утасгүй сүлжээний хувиргагчийг шифрлэлтгүйгээр хэрхэн ажиллуулахыг харуулсан энгийн жишээг үзүүлнэ. Эдгээр ойлголтуудыг мэддэг болсныхоо дараа өөрийнхөө утасгүй сүлжээг тохируулахдаа WPA-г ашиглахыг бид тууштай зөвлөж байна.

Утасгүй сүлжээг тохируулах гурван үндсэн алхам байдаг: эдгээрт хандалтыг цэгийг сонгох, өөрийн станцыг таниулж нэвтрүүлэх, болон IP хаягийг тохируулах багтана. Дараах хэсэг алхам бүрийг хэлэлцэнэ.

#### 32.3.3.1.2.1. Хандалтын цэгийг сонгох нь

Ихэнх тохиолдолд систем өөртөө бүтээгдсэн гүнзгийрүүлэн шалгах аргаа (heuristic) ашиглан хандалтын цэгийг сонгохыг нь зөвшөөрөх хангалттай байдаг. Таныг интерфэйсийг up гэж тэмдэглэх үед энэ нь анхдагч байх бөгөөд хэрэв үгүй бол интерфэйсийг /etc/rc.conf файлд жагсаан тохируулна, өөрөөр хэлбэл:

```
ifconfig_ath0="wlan0"
ifconfig_wlan0="DHCP"
```

Хэрэв олон хандалтын цэгүүд байгаа бөгөөд та тухайн нэгийг сонгохыг хүсвэл түүнийг SSID-аар нь сонгож болно:

```
wlans_ath0="wlan0"
ifconfig_wlan0="ssid your_ssid_here DHCP"
```

Адил SSID-тай олон хандалтын цэгүүд байгаа орчинд (роуминг-ийг (тэнүүчлэх) хялбар болгохын тулд ихэвчлэн хийдэг) аль нэг тухайн төхөөрөмж уруу холбогдох шаардлагатай болж болох юм. Энэ тохиолдолд та хандалтын цэгийн BSSID-ийг зааж өгч бас болох юм (та бас SSID-ийг үлдээж болно):

```
wlans_ath0="wlan0"
ifconfig_wlan0="ssid your_ssid_here bssid xx:xx:xx:xx:xx:xx DHCP"
```

Системийн хайлт хийх давтамжуудын олонлогийг хязгаарлах зэрэг хандалтын цэгийн сонголтыг шаардах өөр аргууд байдаг.Та олон зурвасын утасгүй сүлжээний карттай байх тохиолдолд энэ нь ашигтай байдаг. Ийм тохиолдолд бүх боломжит сувгуудыг хайх нь цаг хугацаа шаардсан ажил байдаг. Ажиллагааг тухайн нэг зурвас дээр хязгаарлахын тулд `mode` параметрийг та ашиглаж болно, өөрөөр хэлбэл:

```
wlans_ath0="wlan0"
ifconfig_wlan0="mode 11g ssid your_ssid_here DHCP"
```

нь картыг 2.4GHz давтамжуудын хувьд тодорхойлогдсон 802.11g горимд картыг ажиллуулах бөгөөд бусад 5GHz-ийн сувгуудыг авч үзэхгүй байх болно. Үүнийг хийх өөр нэг арга нь ажиллагааг тухайн нэг давтамжид түгжих `channel` параметр болон хайлт хийхдээ сувгуудын жагсаалтыг заах `chanlist` параметрийг ашиглах явдал юм. Эдгээр параметрүүдийн талаар дэлгэрэнгүй мэдээллийг ifconfig(8) гарын авлагын хуудаснаас лавлана уу.

#### 32.3.3.1.2.2. Нэвтрэлт танилт

Хандалтын цэгийг сонгосны дараа таны станц өгөгдөл дамжуулахаа өмнө өөрийгөө таниулан нэвтрэх хэрэгтэй. Нэвтрэлт танилт хэд хэдэн аргаар хийгдэж болно. Ашиглагддаг хамгийн түгээмэл схем бол нээлттэй нэвтрэлт танилт гэгддэг бөгөөд энэ нь дурын станц сүлжээнд нэгдэж холбогдохыг зөвшөөрдөг. Энэ нь эхний удаа утасгүй сүлжээг тохируулж

байхдаа тестийн зорилгоор таны ашиглах ёстой нэвтрэлт танилт юм. Бусад схемүүд нь өгөгдлийн урсгал эхлэхээс өмнө криптограф мэдээлэл солилцоо хийгдэхийг шаарддаг; урьдчилсан хуваалцсан түлхүүрүүд эсвэл нууц үгс ашиглах эсвэл RADIUS зэрэг арын үйлчилгээнүүдийг ашигладаг илүү төвөгтэй схемүүд зэргийг дурдаж болно. Ихэнх хэрэглэгчид нээлттэй нэвтрэлт танилт ашиглах бөгөөд энэ нь анхдагч тохиргоо байдаг. Дараагийн хамгийн түгээмэл тохируулга бол WPA-PSK бөгөөд энэ нь бас WPA Personal гэгддэг. Энэ тохируулга доор тайлбарлагдсан байгаа.

Хэрэв танд хандалтын цэгт зориулсан Apple® AirPort® Extreme дээр суурилсан үндсэн станц байгаа бол WEP түлхүүртэй хамт хуваалцсан түлхүүрийг тохируулах хэрэгтэй болж болох юм. Үүнийг /etc/rc.conf файл эсвэл wpa_supplicant(8) програмыг ашиглан хийж болно. Хэрэв та ганц AirPort® үндсэн станцтай бол хандалтыг иймэрхүүгээр хийж өгч болно:

```
wlans_ath0="wlan0"
ifconfig_wlan0="authmode shared wepmode on weptxkey 1 wepkey 01234567
DHCP"
```

Ерөнхийдээ хуваалцсан түлхүүр бүхий нэвтрэлт танилтыг ашиглахаас зайлсхийх хэрэгтэй, яагаад гэвэл WEP түлхүүрийн материалыг их хүчилсэн хэлбэрээр ашигладаг нь түлхүүрийг эвдэх боломжийг илүү амархан болгодог. Хэрэв WEP ашиглагдах ёстой бол (өөрөөр хэлбэл хуучин төхөөрөмжтэй нийцтэй байх үүднээс) WEP-ийг `open` буюу нээлттэй нэвтрэлт танилттай цуг ашиглах нь зүйтэй юм. WEP-ийн талаар дэлгэрэнгүй мэдээллийг WEP-с олж болно.

#### 32.3.3.1.2.3. DHCP-ээр IP хаяг авах

Хандалтын цэгийг сонгож нэвтрэлт танилтын параметрүүдийг тохируулсны дараа холбогдохын тулд та IP хаяг авах хэрэгтэй болно. Та өөрийн утасгүй IP хаягийг ихэвчлэн DHCP-ээр авах болно. Ингэхийн тулд ердөө л /etc/rc.conf файлыг засварлаж дээрх жишээн дээр үзүүлсэн шиг өөрийн төхөөрөмжийн тохиргоонд зориулж `DHCP` гэж нэмж өгнө:

```
wlans_ath0="wlan0"
ifconfig_wlan0="DHCP"
```

Энэ хүрэхэд та утасгүй интерфэйсээ ажиллуулахад бэлэн байх болно:

```
# service netif start
```

Интерфэйс ажиллаж эхэлсний дараа ath0 интерфэйсийн төлөвийг үзэхийн тулд `ifconfig` тушаалыг ашиглана:

```
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
```

```
        ether 00:11:95:d5:43:62
    inet 192.168.1.100 netmask 0xffffff00 broadcast 192.168.1.255
        media: IEEE 802.11 Wireless Ethernet OFDM/54Mbps mode 11g
        status: associated
    ssid dlinkap channel 11 (2462 Mhz 11g) bssid 00:13:46:49:41:76
    country US ecm authmode OPEN privacy OFF txpower 21.5 bmiss 7
    scanvalid 60 bgscan bgscanintvl 300 bgscanidle 250 roam:rssi 7
    roam:rate 5 protmode CTS wme burst
```

`status: associated` мөр нь таныг утасгүй сүлжээнд (бидний тохиолдолд `dlinkap` сүлжээ уруу) холбогдсон гэдгийг харуулж байна. `bssid 00:13:46:49:41:76` хэсэг нь таны хандалтын цэгийн MAC хаяг юм; `authmode OPEN` хэсэг нь холболт шифрлэгдээгүй гэдгийг танд харуулж байна (`OPEN`).

#### 32.3.3.1.2.4. Статик IP хаяг

DHCP серверээс IP хаяг авч чадахгүй тохиолдолд тогтмол IP хаяг та тавьж болно. Дээр үзүүлсэн `DHCP` гэсэн түлхүүр үгийг хаягийн мэдээллээр солих хэрэгтэй. Хандалтын цэгийг сонгохдоо хэрэглэсэн бусад параметрүүдийг үлдээхээ мартуузай:

```
wlans_ath0="wlan0"
ifconfig_wlan0="inet 192.168.1.100 netmask 255.255.255.0 ssid your_ssid_here"
```

#### 32.3.3.1.3. WPA

WPA (Wi-Fi Protected Access буюу Wi-Fi Хамгаалагдсан Хандалт) нь WEP-ийн сул тал болон зөв нэвтрэлт танилтын дутмаг байдлыг арилгах зорилгоор 802.11 сүлжээнүүдэд цуг ашиглагддаг аюулгүй байдлын протокол юм. WPA нь 802.1X нэвтрэлт танилтын протоколыг хөшүүрэгддэг бөгөөд өгөгдлийн бүрэн бүтэн байдалд зориулж WEP-ийн оронд хэд хэдэн шифрүүдийн нэгийг ашигладаг. WPA-ийн шаарддаг цорын ганц шифр бол TKIP (Temporary Key Integrity Protocol) юм. TKIP нь бүрэн бүтэн байдал шалгалт, хуурамч үйлдлийг илрүүлэлт болон илрүүлсэн халдлагуудад хариулахад зориулсан арга хэмжээнүүдийг WEP-ийн ашигладаг үндсэн RC4 шифрт нэмэн өргөтгөсөн шифр юм. TKIP нь хуучин тоног төхөөрөмж дээр зөвхөн програм хангамжийн өөрчлөлттэйгөөр ажиллахаар хийгдсэн; энэ нь аюулгүй байдлыг сайжруулдаг боловч халдлагаас бүрэн гүйцэд хамгаалж чаддаггүй. WPA нь TKIP-г орлуулж AES-CCMP шифрийг бас заадаг бөгөөд боломжтой тохиолдолд үүнийг хэрэглэхийг урьтал болгодог; энэ тодорхойлолтод WPA2 (эсвэл RSN) гэсэн ухагдахуун нийтлэг ашиглагддаг.

WPA нь нэвтрэлт танилт болон шифрлэлтийн протоколуудыг тодорхойлдог. Нэвтрэлт танилт нь хоёр техникийн аль нэгээр ихэвчлэн хийгддэг: 802.1X болон RADIUS зэрэг арын нэвтрэлт танилтын үйлчилгээгээр эсвэл урьдчилан хуваалцсан нууц үг ашиглан станц болон хандалтын хооронд хамгийн багаар мэдээлэл солилцох (handshake) замаар хийгддэг. Эхнийх нь ихэвчлэн WPA Enterprise, сүүлийнх нь WPA Personal гэгддэг. Ихэнх хүмүүс утасгүй сүлжээнд зориулж арын RADIUS сервер өөрсөддөө зориулж тохируулдаггүй учир WPA-PSK нь WPA-д зориулсан хамгийн түгээмэл тохиолддог тохиргоо юм.

Утасгүй холболтын хяналт болон нэвтрэлт танилт (түлхүүрийн тохиролцоо эсвэл сервертэй

хийх нэвтрэлт танилт) нь wpa_supplicant(8) хэрэгслээр хийгддэг. Энэ програм нь ажиллахын тулд /etc/wpa_supplicant.conf тохиргооны файл шаарддаг. Энэ файлын талаар дэлгэрэнгүй мэдээллийг wpa_supplicant.conf(5) гарын авлагын хуудаснаас олж болно.

#### 32.3.3.1.3.1. WPA-PSK

WPA-PSK нь бас WPA Personal гэгддэг бөгөөд өгөгдсөн нууц үгээс үүсгэгдсэн pre-shared key буюу (PSK) урьдчилан хуваалцсан түлхүүр дээр суурилдаг бөгөөд утасгүй сүлжээнд мастер түлхүүр болон ашиглагддаг. Энэ нь утасгүй хэрэглэгч бүр адил түлхүүрийг хуваалцана гэсэн үг юм. WPA-PSK нь нэвтрэлт танилтын сервер хэрэглэх боломжгүй эсвэл шаардлагагүй жижиг сүлжээнүүдэд зориулагдсан юм.

Хангалттай урт, төрөл бүрийн тэмдэгтүүдээс бүтсэн, таах буюу/эсвэл халдах боломжгүй хэцүү нууц үгсийг үргэлж ашиглах хэрэгтэй.

Эхний алхам нь /etc/wpa_supplicant.conf файлыг өөрийн сүлжээний SSID болон урьдчилан хуваалцсан түлхүүрээр тохируулах явдал юм:

```
network={
  ssid="freebsdap"
  psk="freebsdmall"
}
```

Дараа нь бид /etc/rc.conf файлд утасгүй төхөөрөмжийн тохиргоо WPA-аар хийгдэж IP хаяг DHCP-ээс авагдана гэдгийг зааж өгнө:

```
wlans_ath0="wlan0"
ifconfig_wlan0="WPA DHCP"
```

Дараа нь бид интерфэйсийг босгож ажиллуулна:

```
# service netif start
Starting wpa_supplicant.
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 5
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 6
DHCPOFFER from 192.168.0.1
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.1
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet OFDM/36Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
```

```
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

Эсвэл та үүнийг гараар дээрх /etc/wpa_supplicant.conf ашиглан хийж доор дурдсан тушаалыг ажиллуулж болно:

```
# wpa_supplicant -i wlan0 -c /etc/wpa_supplicant.conf
Trying to associate with 00:11:95:c3:0d:ac (SSID='freebsdap' freq=2412 MHz)
Associated with 00:11:95:c3:0d:ac
WPA: Key negotiation completed with 00:11:95:c3:0d:ac [PTK=CCMP GTK=CCMP]
     CTRL-EVENT-CONNECTED - Connection to 00:11:95:c3:0d:ac completed (auth) [id=0
id_str=]
```

Дараагийн үйлдэл нь DHCP серверээс IP хаяг авахын тулд dhclient тушаалыг ажиллуулах явдал юм:

```
# dhclient wlan0
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.1
bound to 192.168.0.254 -- renewal in 300 seconds.
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet OFDM/36Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

Хэрэв /etc/rc.conf файл нь DHCP гэсэн утгатай ifconfig_wlan0 мөртэй тохируулагдсан бол (ifconfig_wlan0="DHCP") wpa_supplicant нь холболтын цэгтэй холбогдсоны дараа dhclient тушаал автоматаар ажиллах болно.

DHCP-ийн хэрэглээ боломжгүй тохиолдолд wpa_supplicant станцыг таниулж нэвтрүүлсний дараа та статик IP хаяг тохируулж болно:

```
# service netif start
Starting wpa_supplicant.
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.20
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
```

```
        inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
        media: IEEE 802.11 Wireless Ethernet DS/11Mbps mode 11g
        status: associated
        ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
        country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
        AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
        bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
        wme burst roaming MANUAL
```

Урьд нь үзүүлсний адил интерфэйсийг гараар `wpa_supplicant` болон `ifconfig` тушаалуудаар босгох боломжтой байдаг.

#### 32.3.3.1.3.2. EAP-TTLS-тэй WPA

EAP-TLS-тэй байхад нэвтрэлт танилтын сервер болон клиентэд сертификат хэрэгтэй бол EAP-TTLS-тэй (EAP-Tunneled Transport Layer Security) байхад клиентийн сертификат нэмэлт сонголт байдаг. Энэ арга нь зорчигчид клиент талдаа сертификатгүй байсан ч гэсэн нууцлаг SSL туннель үүсгэж чаддаг зарим нэг аюулгүй вэб сайтуудын хийдэгтэй ойролцоо байдаг. EAP-TTLS нь нэвтрэлт танилтын өгөгдлийг аюулгүй тээвэрлэхэд зориулж шифрлэгдсэн TLS туннелийг ашиглах болно.

Тохиргоог /etc/wpa_supplicant.conf файлаар хийнэ:

```
network={
  ssid="freebsdap"
  proto=RSN
  key_mgmt=WPA-EAP
  eap=TTLS ①
  identity="test" ②
  password="test" ③
  ca_cert="/etc/certs/cacert.pem" ④
  phase2="auth=MD5"
}
```

① Энэ талбарт бид өөрийн холболтондоо зориулж EAP аргыг дурдана.

② `identity` талбар нь шифрлэгдсэн TLS туннель доторх EAP нэвтрэлт танилтад зориулсан таниулах мөрийг агуулна.

③ `password` талбар нь EAP нэвтрэлт танилтад зориулсан нэвтрэх үгийг агуулна.

④ `ca_cert` талбар нь CA сертификатын файлын замын нэрийг заана. Энэ файл нь серверийн сертификатыг шалгахад хэрэгтэй байдаг.

Энэ талбарт бид шифрлэгдсэн TLS туннельд ашиглагдсан нэвтрэлт танилтын аргыг заана. Бидний тохиолдолд MD5-Challenge-тай цуг EAP ашиглагдаж байна. "inner authentication" дотоод нэвтрэлт танилт үе нь ихэвчлэн "phase2" буюу хоёрдугаар үе гэгддэг.

Дараах мөрүүдийг та /etc/rc.conf файлд бас нэмэх хэрэгтэй:

```
wlans_ath0="wlan0"
ifconfig_wlan0="WPA DHCP"
```

Дараагийн алхам бол интерфэйсийг босгож ажиллуулах явдал юм:

```
# service netif start
Starting wpa_supplicant.
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.20
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet DS/11Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

#### 32.3.3.1.3.3. EAP-PEAP-тэй WPA

PEAPv0/EAP-MSCHAPv2 нь хамгийн түгээмэл PEAP арга юм. Баримтын үлдсэн хэсэгт энэ аргыг PEAP гэж нэрлэх болно.

PEAP (Protected EAP) нь EAP-TTLS-ийн өөр нэг хувилбар хэлбэрээр байхаар хийгдсэн бөгөөд хамгийн EAP-TLS-с гадна хамгийн түгээмэл EAP стандарт юм. Өөрөөр хэлбэл хэрэв та төрөл бүрийн OS-үүд холилдсон сүлжээтэй бол PEAP нь EAP-TLS-ийн дараа орох хамгийн ихээр дэмжигдсэн стандарт байх юм.

PEAP нь EAP-TLS-тэй төстэй байдаг: энэ нь клиент болон нэвтрэлт танилтын серверийн хооронд нэвтрэлт танилтын мэдээллийн солилцоог хамгаалах шифрлэгдсэн туннель үүсгэн клиентүүдийг танин нэвтрүүлэхийн тулд сервер талын сертификатыг ашигладаг. Аюулгүй байдлын ухагдахуун дахь EAP-TTLS болон PEAP-ийн хоорондох ялгаа нь PEAP нэвтрэлт танилт хэрэглэгчийн нэрийг цэвэр текст хэлбэрээр, зөвхөн нууц үгийг шифрлэгдсэн TLS туннелээр цацдаг явдал юм. EAP-TTLS нь хэрэглэгчийн нэр болон нууц үгт зориулж TLS туннелийг ашиглах болно.

Бид /etc/wpa_supplicant.conf файлыг засварлаж EAP-PEAP-тэй холбоотой тохируулгуудыг нэмэх хэрэгтэй болно:

```
network={
  ssid="freebsdap"
  proto=RSN
```

```
  key_mgmt=WPA-EAP
  eap=PEAP ①
  identity="test" ②
  password="test" ③
  ca_cert="/etc/certs/cacert.pem" ④
  phase1="peaplabel=0" ⑤
  phase2="auth=MSCHAPV2" ⑥
}
```

① Энэ талбарт бид өөрийн холболтондоо зориулж EAP аргыг ашиглахаа дурдана.

② `identity` талбар нь шифрлэгдсэн TLS туннель доторх EAP нэвтрэлт танилтад зориулсан таниулах мөрийг агуулна.

③ `password` талбар нь EAP нэвтрэлт танилтад зориулсан нэвтрэх үгийг агуулна.

④ `ca_cert` талбар нь CA сертификатын файлын замын нэрийг заана. Энэ файл нь серверийн сертификатыг шалгахад хэрэгтэй байдаг.

⑤ Энэ талбар нь нэвтрэлт танилтын (TLS туннель) эхний үед зориулсан параметрүүдийг агуулна. Ашигласан нэвтрэлт танилтын серверээс хамааранд нэвтрэлт танилтад зориулж тусгай хаяг/шошгыг зааж өгөх хэрэгтэй болно. Ихэнх тохиолдолд хаяг/шошго нь "клиентийн EAP шифрлэлт" байх бөгөөд үүнийг `peaplabel=0` гэж тохируулна. Илүү мэдээллийг wpa_supplicant.conf(5) гарын авлагын хуудаснаас олж болно.

⑥ Энэ талбарт бид шифрлэгдсэн TLS туннельд ашиглагдсан нэвтрэлт танилтын аргыг заана. PEAP-ийн хувьд энэ нь `auth=MSCHAPV2` байна.

Доор дурдсаныг /etc/rc.conf файлд нэмэх ёстой:

```
wlans_ath0="wlan0"
ifconfig_wlan0="WPA DHCP"
```

Дараа нь бид интерфэйсийг босгож ажиллуулж болно:

```
# /etc/rc.d/netif start
Starting wpa_supplicant.
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.20
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet DS/11Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
```

```
      wme burst roaming MANUAL
```

#### 32.3.3.1.4. WEP

WEP (Wired Equivalent Privacy) нь анхдагч 802.11 стандартын хэсэг юм. Үүнд ямар ч нэвтрэлт танилтын арга байхгүй, энэ нь хандалт хяналтын зөвхөн сул хэлбэр бөгөөд хялбар эвдэх боломжтой байдаг.

WEP-ийг `ifconfig` тушаалаар тохируулж болно:

```
# ifconfig wlan0 create wlandev ath0
        # ifconfig wlan0 inet 192.168.1.100 netmask 255.255.255.0 \
          ssid my_net wepmode on weptxkey 3 wepkey 3:0x3456789012
```

- `weptxkey` нь дамжуулалтад ямар WEP түлхүүр ашиглахыг хэлж байна. Энд бид гурав дахь түлхүүрийг ашиглаж байна. Энэ нь хандалтын цэг дэх тохиргоотой таарах ёстой. Хэрэв та хандалтын цэг дээр ямар түлхүүр ашиглагдаж байгааг мэдэхгүй байгаа бол энэ утгад `1` -ийг (өөрөөр хэлбэл эхний түлхүүр) ашиглах хэрэгтэй.
- `wepkey` нь аль нэг WEP түлхүүрийг сонгоно. Энэ нь *index:key* хэлбэрийн байх ёстой. Анхдагчаар `1` гэж түлхүүр тохируулагдана. Хэрэв бид эхний түлхүүрээс өөр түлхүүрүүдийг ашиглах бол индексийг тохируулах хэрэгтэй.

Та `0x3456789012`-г хандалтын цэг дээр ашиглахаар тохируулсан түлхүүрээр солих ёстой.

Цаашхи мэдээллийг ifconfig(8) гарын авлагын хуудаснаас унших нь зүйтэй юм.

`wpa_supplicant` хэрэгслийг өөрийн утасгүй интерфэйсийг WEP-тэй тохируулахын тулд бас ашиглаж болно. Дараах мөрийг /etc/wpa_supplicant.conf файлд нэмж дээрх жишээг тохируулж болно:

```
network={
  ssid="my_net"
  key_mgmt=NONE
  wep_key3=3456789012
  wep_tx_keyidx=3
}
```

Дараа нь:

```
# wpa_supplicant -i wlan0 -c /etc/wpa_supplicant.conf
Trying to associate with 00:13:46:49:41:76 (SSID='dlinkap' freq=2437 MHz)
Associated with 00:13:46:49:41:76
```

### 32.3.4. Ad-hoc горим

IBSS горим буюу бас ad-hoc гэгддэг горим нь цэгээс цэгт холбогдох холболтуудад зориулагдан хийгдсэн. Жишээ нь A машин болон B машины хооронд ad-hoc сүлжээ үүсгэхийн тулд бид ердөө л хоёр IP хаяг болон SSID сонгох хэрэгтэй болно.

A машин дээр:

```
# ifconfig wlan0 create wlandev ath0 wlanmode adhoc
# ifconfig wlan0 inet 192.168.0.1 netmask 255.255.255.0 ssid freebsdap
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:c3:0d:ac
          inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <adhoc>
      status: running
      ssid freebsdap channel 2 (2417 Mhz 11g) bssid 02:11:95:c3:0d:ac
          country US ecm authmode OPEN privacy OFF txpower 21.5 scanvalid 60
      protmode CTS wme burst
```

adhoc параметр нь интерфэйс IBSS горимд ажиллаж байгааг харуулж байна.

B машин дээр бид A машиныг илрүүлж чадах ёстой:

```
# ifconfig wlan0 create wlandev ath0 wlanmode adhoc
# ifconfig wlan0 up scan
  SSID/MESH ID       BSSID              CHAN RATE        S:N   INT CAPS
  freebsdap       02:11:95:c3:0d:ac    2   54M -64:-96   100 IS  WME
```

Гаралт дээрх I нь A машин ad-hoc горимд байгааг батална. Бид одоо B-г өөр IP хаягтайгаар тохируулах хэрэгтэй:

```
# ifconfig wlan0 inet 192.168.0.2 netmask 255.255.255.0 ssid freebsdap
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:d5:43:62
          inet 192.168.0.2 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <adhoc>
      status: running
      ssid freebsdap channel 2 (2417 Mhz 11g) bssid 02:11:95:c3:0d:ac
          country US ecm authmode OPEN privacy OFF txpower 21.5 scanvalid 60
      protmode CTS wme burst
```

A болон B нь одоо мэдээлэл солилцоход бэлэн боллоо.

### 32.3.5. FreeBSD хандах цэг болж ажиллах

FreeBSD нь Access Point (AP) буюу хандах цэг болж ажилласнаар нэмэлт хандах цэг болсон төхөөрөмж худалдаж авах эсвэл цэгээс цэгийн хооронд холбогдож ажилладаг сүлжээ байгуулах зэрэг нэмэлт үйлдлүүдийг хэмнэж болдог. Хэрэв таны FreeBSD машин сүлжээнээс сүлжээний хооронд дамжуулдаг гүүр болж (жишээ нь интернэтэд холбож) ажиллах шаардлагатай үед илүү тохирдог.

#### 32.3.5.1. Үндсэн тохиргоо

FreeBSD машинаа хандах цэг болж ажилладгаар тохируулахаасаа өмнө цөмөө сүлжээнийхээ картанд тохируулж тохируулсан байх ёстой. Та мөн хэрэглэх нууцлалын дамжуулах протокол буюу бүртгэлээ нэмэх хэрэгтэй. Нэмэлт мэдээллийг Үндсэн тохируулга хэсгээс харна уу.

NDIS хэмээгч драйвер орлуулагч болон Виндоусын драйверуудыг ашигласан тохиолдолд хандах цэг болгож ашиглах боломж одоогоор байхгүй. Зөвхөн FreeBSD-н үндсэн утасгүй холболтын драйвер л хандах цэг болох горимыг дэмжиж байгаа.

Утасгүй холболтын сүлжээ дэмжигдэх ажилладаг болж ирэнгүүт нь утасгүй сүлжээний төхөөрөмж тань тухайн машин дээрээ хандах цэг болж ажиллах чадвартайг шалгаж болдог (заримдаа hostap горим гэж нэрлэдэг):

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 list caps
drivercaps=6f85edc1<STA,FF,TURBOP,IBSS,HOSTAP,AHDEMO,TXPMGT,SHSLOT,SHPREAMBLE,MONITOR,
MBSS,WPA1,WPA2,BURST,WME,WDS,BGSCAN,TXFRAG>
cryptocaps=1f<WEP,TKIP,AES,AES_CCM,TKIPMIC>
```

Дээрх тушаалын гаралтаас HOSTAP гэдэг үг нь тухай утасгүй сүлжээний карт нь хандах цэг болж ажиллах чадвартай гэдгийг баталж өгч байна. Мөн төрөл бүрийн нууцлалын боломжууд нь WEP, TKIP, AES, гэх мэтээр дурдагддаг. Эдгээр нь хандах цэгийн дамжуулалтаа нууцлахад хэрэглэгдэх чухал хэрэгтэй мэдээлэл болж өгдөг.

Сүлжээний псевдо төхөөрөмжийг үүсгэх үед утасгүй сүлжээний төхөөрөмжийг зөвхөн hostap горимд оруулах боломжтой байдаг, тэгэхээр өмнө нь үүсгэсэн төхөөрөмжийг эхлээд устгах ёстой:

```
# ifconfig wlan0 destroy
```

Дараа нь харин бусад параметрүүдийг тохируулахаасаа өмнө зөв тохиргоотойгоор дахин үүсдэг:

```
# ifconfig wlan0 create wlandev ath0 wlanmode hostap
# ifconfig wlan0 inet 192.168.0.1 netmask 255.255.255.0 ssid freebsdap mode 11g
```

```
channel 1
```

Дахин `ifconfig` тушаалаар wlan0 төхөөрөмжийн төлөв байдлыг шалгая:

```
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:c3:0d:ac
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <hostap>
      status: running
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy OFF txpower 21.5 scanvalid 60
      protmode CTS wme burst dtimperiod 1 -dfs
```

`hostap` үг нь уг төхөөрөмжийг тухайн машин дээр хандах цэг болж ажиллаж байгааг нь илэрхийлж байна.

Тухайн төхөөрөмжийн ийм тохируулгыг машин асах үед нь хийгдэхээр тохируулъя гэж бодсон тохиолдолд /etc/rc.conf файлд дараах мөрүүдийг оруулж өгөх хэрэгтэй:

```
wlans_ath0="wlan0"
create_args_wlan0="wlanmode hostap"
ifconfig_wlan0="inet 192.168.0.1 netmask 255.255.255.0 ssid freebsdap mode 11g channel
1"
```

#### 32.3.5.2. Тухайн машин дээр хандах цэг нь нэвтрэх үйлдэл болон нууцлах үйлчилгээ ашиглахгүй ажиллах

Хэдийгээр хандах цэгийг нэвтрэх үйлдэл болон холболтын нууцлал ашиглахгүйгээр ажиллуулах нь тийм сайшаалтай биш боловч хандах цэгийн ажиллагааг шалгахын тулд ингэж тохируулж ажиллуулах шаардлага гардаг. Энэ тохируулга нь мөн хэрэглэгчтэй холбогдолтой хүндрэлүүдийг шийдэхэд бас хэрэглэгдэж болно.

Өмнө харуулснаар тохируулагдсан хандах цэгийг өөр утасгүй сүлжээтэй ажиллаж буй машинаас хайлт хийж олж болдог:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 up scan
SSID/MESH ID    BSSID              CHAN RATE   S:N     INT CAPS
freebsdap       00:11:95:c3:0d:ac    1   54M -66:-96  100 ES   WME
```

Хэрэглэгч машинаас хандах цэгийг ингэж олоод түүнтэй дараах маягаар холбогдож болно:

```
# ifconfig wlan0 inet 192.168.0.2 netmask 255.255.255.0 ssid freebsdap
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
```

```
        ether 00:11:95:d5:43:62
        inet 192.168.0.2 netmask 0xffffff00 broadcast 192.168.0.255
        media: IEEE 802.11 Wireless Ethernet OFDM/54Mbps mode 11g
        status: associated
        ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
        country US ecm authmode OPEN privacy OFF txpower 21.5 bmiss 7
        scanvalid 60 bgscan bgscanintvl 300 bgscanidle 250 roam:rssi 7
        roam:rate 5 protmode CTS wme burst
```

#### 32.3.5.3. WPA нууцлалтай хандах цэг болсон машин

Энэ хэсэгт хандах цэг болж ажиллаж буй FreeBSD машиныг WPA нууцлалын бүртгэл хэрэглэх тухай өгүүлэх болно. WPA нууцлалтай холбогдолтой болон WPA дээр суурилсан хэрэглэгчийн тохиргооны талаар дэлгэрэнгүй мэдээллийг WPA хэсгээс хараарай.

hostapd нэртэй далд чөтгөр нь WPA-г ашиглаж буй хандах цэг дээр хэрэглэгчийг нэвтрүүлэх болон түлхүүр үг солилцох үйлдлүүдэд зориулагдсан байдаг.

Дараах жишээн дээр бүх тохиргооны үйлдлүүд нь хандах цэг болж ажиллаж буй FreeBSD машин дээр гүйцэтгэгдэх болно. Хандах цэг зөв ажиллаж эхэлсэн үед hostapd далд чөтгөрийг автоматаар машиныг ачаалагдах үед зөвшөөрөхийн тулд /etc/rc.conf файлд дараах мөрийг нэмж өгөх хэрэгтэй:

```
hostapd_enable="YES"
```

hostapd далд чөтгөрийг тохируулахаасаа өмнө Үндсэн тохиргоо хэсэгт заасан ерөнхий тохиргоог хийсэн эсэхээ магадлаарай.

##### 32.3.5.3.1. WPA-PSK

WPA-PSK нь хэрэглэгчид нэвтрэх үйлдэл хийдэг сервер шаардлагагүй юм уу эсвэл боломжгүй болсон бага зэргийг сүлжээнд зориулагдсан билээ.

Тохиргоо нь /etc/hostapd.conf файлд хийгддэг:

```
interface=wlan0 ①
debug=1 ②
ctrl_interface=/var/run/hostapd ③
ctrl_interface_group=wheel ④
ssid=freebsdap ⑤
wpa=1 ⑥
wpa_passphrase=freebsdmall ⑦
wpa_key_mgmt=WPA-PSK ⑧
wpa_pairwise=CCMP TKIP ⑨
```

① Энэ талбар нь хандах цэг болж ажиллаж буй утасгүй сүлжээний төхөөрөмжийг заадаг.

② hostapd-н гүйцэтгэж байгаа явцыг харуулалтын түвшинг заадаг талбар. 1 гэсэн утга нь хамгийн бага түвшинг зааж байгаа билээ. `ctrl_interface` гэсэн талбар нь hostapd-д

хэрэглэгддэг hostapd_cli(8) төрлийн гадны програмуудтай холбогдоход хэрэглэгддэг домэйн сокет файлуудыг хадгалах сангийн байрлалыг зааж өгдөг. Энд анхдагч утга нь хэрэглэгдсэн байна.

③ `ctrl_interface_group` гэсэн мөрөнд хяналт хийдэг төхөөрөмж рүү хандаж болох бүлгийн нэр (энд `wheel` бүлгийг оруулсан) байна.

④ Энэ талбарт сүлжээний нэрийг оруулдаг.

⑤ `wpa` талбар нь WPA нууцлалыг зөвшөөрдөг бөгөөд ямар бүртгэл шаардлагатайг нь тодорхойлж өгнө. `1` гэсэн утга нь хандах цэгийг WPA-PSK нууцлалаар тохируулдаг.

⑥ `wpa_passphrase` талбарт ASCII тэмдэгтээр бичигдсэн WPA нэвтрэх үйлдлийн нууц үг агуулагддаг.

⑦ `wpa_key_mgmt` мөр нь хэрэглэж буй түлхүүр үгний зориулалтанд хамаатай. Манай тохиолдолд энэ нь WPA-PSK байна.

⑧ `wpa_pairwise` талбар нь хандах цэгт тохиромжтой нууц хувиргалтуудыг харуулдаг. Энд TKIP (WPA) ба CCMP (WPA2) хувиргагчид хоёулаа хүлээн зөвшөөрөгдөнө. CCMP хувиргагч нь TKIP-н бас нэг хувилбар бөгөөд боломжтой бол үүнийг илүүд үзэх хэрэгтэй. TKIP хувиргагчийг CCMP хувиргагч хэрэглэх боломжгүй үед л сонгох нь зүйтэй.

Дараагийн алхам нь hostapd-г эхлүүлэх:

```
# service hostapd forcestart
```

```
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 2290
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      inet6 fe80::211:95ff:fec3:dac%ath0 prefixlen 64 scopeid 0x4
      ether 00:11:95:c3:0d:ac
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <hostap>
      status: associated
      ssid freebsdap channel 1 bssid 00:11:95:c3:0d:ac
      authmode WPA2/802.11i privacy MIXED deftxkey 2 TKIP 2:128-bit txpowmax 36
protmode CTS dtimperiod 1 bintval 100
```

Хандах цэг одоо ажиллагаанд орсон бөгөөд хэрэглэгчийн зүгээс холбогдох тухай дэлгэрэнгүй мэдээллийг WPA хэсгээс харна уу. Хандах цэгт холбогдсон төхөөрөмжүүдийг `ifconfig wlan0 list sta` гэсэн тушаалаар харж болно.

#### 32.3.5.4. WEP нууцлал хэрэглэж хандах цэг болсон машин

Ямар ч нэвтрэх үйлдэл хэрэглэдэггүй учраас амархан дайралтад өртдөгийн улмаас WEP нууцлалын хувиргуурыг хэрэглэх нь тийм сайшаалтай биш. Зарим худалдаанд хэрэглэгдэж буй утасгүй сүлжээний картууд нь зөвхөн WEP нууцлалын бүртгэл хэрэглэдэг ба ийм картуудыг хандах цэг болгож хэрэглэхээр бол хэрэглэгчийн таних үйлдэл хийхээр тохируулах боломж байхгүй байх болно.

Утасгүй сүлжээний төхөөрөмж тэгэхээр одоо hostap горимд ажиллаж болох бөгөөд

шаардлагатай зөв SSID дугаар болон IP хаягаа авах шаардлагатай:

```
# ifconfig wlan0 create wlandev ath0 wlanmode hostap
# ifconfig wlan0 inet 192.168.0.1 netmask 255.255.255.0 \
    ssid freebsdap wepmode on weptxkey 3 wepkey 3:0x3456789012 mode 11g
```

- `weptxkey` гэдэг нь дамжуулалтад WEP нууцлал хэрэглэхийг зааж өгч байна. Энд бид гуравдугаар түлхүүрийг (түлхүүрийн дугаар нь 1 гэсэн тоогоор эхэлдэг) хэрэглэсэн байна. Энэ сонголт нь дамжуулагдах өгөгдлийг хувиргахад хэрэглэгддэг.
- `wepkey` гэдэг нь WEP түлхүүрийг сонгосныг зааж байна. Энэ нь *index:key* гэсэн хэлбэртэй байдаг ба хэрэв эхний дугаарлалт нь заагдаагүй бол түлхүүр нь 1 гэж суугддаг. Тэгэхээр хэрэв эхний түлхүүрээс өөр түлхүүр хэрэглэх шаардлагатай бол эхний дугаарлалтыг нь зааж өгөх шаардлагатай гэсэн үг.

Ингээд дахин `ifconfig` тушаалыг ашиглаж wlan0 төхөөрөмжийн төлөв байдлыг шалгаж болно:

```
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:c3:0d:ac
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <hostap>
      status: running
      ssid freebsdap channel 4 (2427 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy ON deftxkey 3 wepkey 3:40-bit
      txpower 21.5 scanvalid 60 protmode CTS wme burst dtimperiod 1 -dfs
```

Өөр утасгүй холбогддог машинаас хандах цэгийг хайж шалгаж болдог:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 up scan
SSID            BSSID              CHAN RATE  S:N   INT CAPS
freebsdap       00:11:95:c3:0d:ac    1   54M 22:1   100 EPS
```

Хэрэглэгчийн машинаас хандах цэгийг олоод зөв тохиргоог нь зааж өгөн (дугаар түлхүүр гэх мэт) холбогдох тухай дэлгэрэнгүй зааврыг WEP хэсгээс харна уу.

### 32.3.6. Утастай болон утасгүй холболтыг ашиглах нь

Утастай сүлжээний холболт нь илүү сайн ажиллагаа болон найдвартай байдлыг хангадаг бол утасгүй сүлжээний холболт нь уян хатан байдал, зөөх боломжийг бүрдүүлдэг бөгөөд зөөврийн компьютер ашигладаг хэрэглэгчид ихэвчлэн эдгээрийг хамтад нь ашиглаж хооронд нь чөлөөтэй шилжиж байхыг хүсдэг.

FreeBSD дээр хоёр болон түүнээс олон сүлжээний интерфэйсийг нийлүүлж "failover" горимоор ашиглах боломжтой бөгөөд энэ нь сүлжээний интерфэйсүүдээс хамгийн

боломжтойг ашиглаж холболтын төлөв өөрчлөгдөхөд үйлдлийн системийн тусламжтайгаар автоматаар өөр холболт руу шилжих боломж юм.

Бид холболт нийлүүлэх болон failover горимыг Холбоос нэгтгэлт ба ослыг тойрон гарах (Link Aggregation and Failover) хэсэгт авч үзэх бөгөөд утастай болон утасгүй холболтыг ашиглах талаарх жишээ Утастай болон утасгүй сүлжээний интерфэйсүүдийн хоорондох Failover горим хэсэгт бас байгаа болно.

### 32.3.7. Алдааг олж засварлах

Хэрэв та утасгүй сүлжээндээ асуудалтай байгаа бол асуудлыг олж засварлахад туслах хэд хэдэн алхмууд байдаг.

- Хайлт хийж байхдаа та хандалтын цэгийг олж харахгүй байгаа бол та өөрийн утасгүй төхөөрөмжөө тодорхой хэдэн сувгууд дээр хязгаарлаж тохируулаагүй эсэхээ шалгаарай.
- Хэрэв та хандалтын цэгт холбогдож чадахгүй байгаа бол таны станцын тохиргоо хандалтын цэгийн аль нэгтэй тохирч байгаа эсэхийг шалгаарай. Үүнд нэвтрэлт танилтын схем болон аюулгүй байдлын протоколууд хамаарна. Өөрийн тохиргоогоо аль болох хялбаршуулах хэрэгтэй. Хэрэв та WPA эсвэл WEP зэрэг аюулгүй байдлын протоколыг ашиглаж байгаа бол хандалтын цэгийг нээлттэй нэвтрэлт танилтад зориулж ямар нэгэн аюулгүй байдлын хамгаалалтгүй тохируулж урсгал дамжиж байгаа эсэхийг үзэх хэрэгтэй.
- Хандалтын цэгт холбогдсоныхоо дараа ping(8) зэрэг хялбар хэрэгслүүдийг ашиглаад ямар ч аюулгүй байдлын тохиргоог оношилж болно.

  `wpa_supplicant` нь илүү дибаг хийх дэмжлэгтэй байдаг; үүнийг `-dd` сонголттой гараар ажиллуулж системийн бүртгэлүүдийг шалгах хэрэгтэй.

- Мөн олон доод түвшний дибаг хийх хэрэгслүүд бас байдаг. Та 802.11 протоколын дэмжлэг давхаргад дибаг мэдэгдлүүдийг /usr/src/tools/tools/net80211 дахь `wlandebug` програмыг ашиглан идэвхжүүлж болно. Жишээ нь:

  ```
  # wlandebug -i ath0 +scan+auth+debug+assoc
    net.wlan.0.debug: 0 => 0xc80000<assoc,auth,scan>
  ```

  тушаал нь хандалтын цэгүүдийг хайх болон холбоог зохион байгуулах 802.11 протоколын мэдээлэл солилцоонуудыг хийхтэй холбоотой консолын мэдэгдлүүдийг идэвхжүүлэхэд ашиглагдаж болох юм.

  802.11 давхаргын арчилж байдаг олон ашигтай статистикууд бас байдаг; энэ мэдээллийг `wlanstats` хэрэгсэл харуулах болно. Эдгээр статистикууд нь 802.11 давхаргаар танигдсан бүх алдаануудыг таних ёстой. Гэхдээ 802.11 давхаргаас доош орших төхөөрөмжийн драйверууд дээр танигдсан зарим алдаанууд нь харуулагдахгүй байж болохыг санаарай. Төхөөрөмжтэй холбоотой асуудлуудыг оношлохын тулд та драйверийн баримтаас лавлах хэрэгтэй юм.

Хэрэв дээрх мэдээлэл асуудлыг тодруулахад тань туслахгүй байгаа бол дээрх хэрэгслүүдээс гарсан гаралтыг оруулж асуудлынхаа тайланг илгээгээрэй.

# 32.4. Bluetooth

## 32.4.1. Танилцуулга

Bluetooth нь 10 метрийн дотор 2.4 GHz давтамжийн лицензжүүлээгүй зурваст ажиллах хувийн сүлжээнүүд үүсгэхэд зориулагдсан утасгүй технологи юм. Сүлжээнүүд нь үүрэн утас, гарын цахим жижиг хэрэгслүүд, болон зөөврийн компьютерууд зэрэг зөөврийн төхөөрөмжүүдээс ad-hoc ихэвчлэн бүрдүүлдэг. Бусад түгээмэл утасгүй технологиудаас ялгаатай тал нь Wi-Fi, Bluetooth нь илүү өндөр түвшний үйлчилгээний хувийн тохиргоонуудыг санал болгодог, өөрөөр хэлбэл FTP-тэй адил файлын серверүүд, файл түлхэх, дуу дамжуулалт, цуваа шугамын эмуляц зэрэг олныг дурдаж болно.

FreeBSD дэх Bluetooth стек нь Netgraph тогтолцоог ашиглан хийгдсэн байдаг (netgraph(4)-г үзнэ үү). Олон төрлийн Bluetooth USB хамгаалах төхөөрөмжүүд (dongle) ng_ubt(4) драйвераар дэмжигдсэн байдаг. Broadcom BCM2033 бичил схем дээр суурилсан Bluetooth төхөөрөмжүүд нь ubtbcmfw(4) болон ng_ubt(4) драйверуудаар дэмжигдсэн байдаг. 3Com Bluetooth PC Карт 3CRWB60-A нь ng_bt3c(4) драйвераар дэмжигдсэн байдаг. Цуваа болон UART дээр суурилсан Bluetooth төхөөрөмжүүд нь sio(4), ng_h4(4) болон hcseriald(8) драйверуудаар дэмжигдсэн. Энэ хэсэг нь USB Bluetooth dongle-ийн хэрэглээг тайлбарлах болно.

## 32.4.2. Төхөөрөмжид залгах нь

Анхдагчаар Bluetooth төхөөрөмжийн драйверууд нь цөмийн модуль хэлбэрээр байдаг. Төхөөрөмжийг залгахаасаа өмнө та драйверийг цөмд дуудаж ачаалах хэрэгтэй болно:

```
# kldload ng_ubt
```

Хэрэв Bluetooth төхөөрөмж системийг эхлүүлэх явцад системд байх юм бол /boot/loader.conf файлаас модулийг дуудна:

```
ng_ubt_load="YES"
```

Өөрийн USB dongle-ийг залга. Консол (эсвэл syslog) дээр доор дурдсантай төстэй гаралт гарч ирэх болно:

```
ubt0: vendor 0x0a12 product 0x0001, rev 1.10/5.25, addr 2
ubt0: Interface 0 endpoints: interrupt=0x81, bulk-in=0x82, bulk-out=0x2
ubt0: Interface 1 (alt.config 5) endpoints: isoc-in=0x83, isoc-out=0x3,
      wMaxPacketSize=49, nframes=6, buffer size=294
```

service(8) нь Bluetooth стекийг эхлүүлэх болон зогсооход хэрэглэгддэг. Төхөөрөмжийг салгахаасаа өмнө стекийг зогсоох нь зөв байдаг, гэхдээ энэ нь (ихэвчлэн) сүйрлийн биш байдаг. Стекийг эхлүүлж байхад доор дурдсантай төстэй гаралтыг та хүлээн авах болно:

```
# service bluetooth start ubt0
```

```
BD_ADDR: 00:02:72:00:d4:1a
Features: 0xff 0xff 0xf 00 00 00 00 00
<3-Slot> <5-Slot> <Encryption> <Slot offset>
<Timing accuracy> <Switch> <Hold mode> <Sniff mode>
<Park mode> <RSSI> <Channel quality> <SCO link>
<HV2 packets> <HV3 packets> <u-law log> <A-law log> <CVSD>
<Paging scheme> <Power control> <Transparent SCO data>
Max. ACL packet size: 192 bytes
Number of ACL packets: 8
Max. SCO packet size: 64 bytes
Number of SCO packets: 8
```

### 32.4.3. Host Controller Interface (HCI) буюу Хостын Хянагчийн Интерфэйс

Host Controller Interface (HCI) буюу Хостын Хянагчийн Интерфэйс нь үндсэн зурвасын хянагч болон холболтын менежерт тушаалын интерфэйсийг, тоног төхөөрөмжийн төлөв болон хяналтын регистрүүдэд хандалтыг өгдөг байна. Энэ интерфэйс нь Bluetooth-ийн үндсэн зурвасын боломжуудад хандах нэгэн хэвийн аргыг олгодог. Хост дээрх HCI давхарга нь өгөгдөл болон тушаалуудыг Bluetooth тоног төхөөрөмж дээрх HCI firmware-тэй солилцдог. Хостын Хянагчийн Тээврийн Давхаргын (өөрөөр хэлбэл физик шугам) драйвер нь HCI давхаргуудад нэг нь нөгөөдөө мэдээлэл солилцох боломжоор хангаж өгдөг.

*hci* төрлийн ганц Netgraph цэг ганц Bluetooth төхөөрөмжийн хувьд үүсдэг. HCI цэг нь Bluetooth төхөөрөмжийн драйверийн цэгт (доош) болон L2CAP цэгт (дээш) ихэвчлэн холбогддог. Бүх HCI үйлдлүүд нь төхөөрөмжийн драйверийн цэг дээр биш HCI цэг дээр хийгдэх ёстой. HCI цэгийн анхдагч нэр нь "devicehci" юм. Илүү дэлгэрэнгүй мэдээллийг ng_hci(4) гарын авлагын хуудаснаас лавлана уу.

Хамгийн нийтлэг ажлуудын нэг нь RF-ийн ойр Bluetooth төхөөрөмжүүдийг олох явдал юм. Энэ үйлдлийг *inquiry* буюу лавлагаа гэдэг. Лавлагаа болон бусад HCI-тэй холбоотой үйлдлүүд нь hccontrol(8) хэрэгслээр хийгддэг. Доорх жишээ нь ойр орчим ямар Bluetooth төхөөрөмжүүд байгааг хэрхэн олохыг харуулж байна. Та төхөөрөмжүүдийн жагсаалтыг хэдхэн секундэд авах ёстой. Алсын төхөөрөмж нь *илрүүлэгдэх* горимд байгаа тохиолдолд лавлагаанд зөвхөн хариулах болно гэдгийг санаарай.

```
% hccontrol -n ubt0hci inquiry
Inquiry result, num_responses=1
Inquiry result #0
       BD_ADDR: 00:80:37:29:19:a4
       Page Scan Rep. Mode: 0x1
       Page Scan Period Mode: 00
       Page Scan Mode: 00
       Class: 52:02:04
       Clock offset: 0x78ef
Inquiry complete. Status: No error [00]
```

BD_ADDR нь Bluetooth төхөөрөмжийн сүлжээний картанд байдаг MAC хаягууд шиг

давтагдашгүй хаяг юм. Энэ хаяг нь төхөөрөмжтэй холбогдоход цаашид хэрэг болдог. BD_ADDR-т хүн уншиж болохоор нэр өгөх боломжтой байдаг. /etc/bluetooth/hosts файл нь мэдэгдэж байгаа Bluetooth хостуудын тухай мэдээллийг агуулдаг. Дараах жишээ нь алсын төхөөрөмжид өгсөн хүн уншиж болохоор нэрийг хэрхэн авч болохыг үзүүлж байна:

```
% hccontrol -n ubt0hci remote_name_request 00:80:37:29:19:a4
BD_ADDR: 00:80:37:29:19:a4
Name: Pav's T39
```

Хэрэв та лавлагааг алсын Bluetooth төхөөрөмж дээр хийх юм бол энэ нь таны компьютерийг "your.host.name (ubt0)" хэлбэрээр олох болно. Локал төхөөрөмжид өгсөн нэрийг ямар ч үед өөрчилж болно.

Bluetooth систем нь цэгээс-цэгт-хүрэх (point-to-point) эсвэл цэгээс-олон-цэгт-хүрэх (point-to-multipoint) холболтын боломжийг олгодог. Цэгээс-олон-цэгт-хүрэх (point-to-multipoint) холболтод холболт нь хэд хэдэн Bluetooth төхөөрөмжүүдийн хооронд хуваалцан хэрэглэгддэг. Дараах жишээ нь локал төхөөрөмжийн хувьд идэвхтэй үндсэн зурвасын холболтуудын жагсаалтыг хэрхэн авахыг үзүүлж байна:

```
% hccontrol -n ubt0hci read_connection_list
Remote BD_ADDR    Handle Type Mode Role Encrypt Pending Queue State
00:80:37:29:19:a4     41  ACL    0 MAST    NONE       0     0 OPEN
```

*connection handle* буюу *холболтын гар* нь үндсэн зурвасын холболтыг дуусгах шаардлагатай үед ашигтай байдаг. Үүнийг гараар хийхийг ерөнхийдөө шаарддаггүйг санаарай. Стек нь идэвхгүй байгаа үндсэн зурвасын холболтуудыг автоматаар дуусгах болно.

```
# hccontrol -n ubt0hci disconnect 41
Connection handle: 41
Reason: Connection terminated by local host [0x16]
```

HCI тушаалуудын бүрэн жагсаалыг `hccontrol help` гэж лавлана уу. HCI тушаалуудын ихэнх нь супер хэрэглэгчийн зөвшөөрлүүдийг шаарддаггүй.

### 32.4.4. Logical Link Control and Adaptation Protocol (L2CAP) буюу Логик Холболтын Хяналт ба Тааруулах Протокол

Логик Холболтын Хяналт ба Тааруулах Протокол (L2CAP) нь холболт дээр тулгуурласан болон холболтгүй (connection-oriented and connectionless) өгөгдлийн үйлчилгээнүүдийг протокол олон хуваагдах (multiplex) чадвар болон сегмент болгож дахин цуглуулах үйлдэлтэй цуг дээд түвшний протоколуудад хангаж өгдөг. L2CAP нь уртаараа 64 килобайт хүртэл хэмжээний L2CAP пакетуудыг дамжуулж хүлээн авахыг өндөр түвшний протоколууд болон програмуудад зөвшөөрдөг.

L2CAP нь *сувгууд* гэсэн ойлголт дээр тулгуурладаг. Суваг нь үндсэн зурвасын холболт дээрх логик холболт юм. Суваг бүр ганц протоколд олноос нэг уруу чиглэсэн загвараар уягдсан

байдаг. Олон сувгууд нэг протоколд уягдаж болдог боловч нэг сувгийг олон протоколд уяж болдоггүй. Суваг дээр хүлээн авсан L2CAP пакет бүр зохих дээд түвшний протокол руу чиглүүлэгддэг. Олон сувгууд нь нэг үндсэн зурвасын холболтыг хуваалцаж болно.

*l2cap* төрлийн ганц Netgraph цэг ганц Bluetooth төхөөрөмжийн хувьд үүсгэгддэг. L2CAP цэг нь Bluetooth HCI цэг (доош) болон Bluetooth сокетуудад (дээш) ихэвчлэн холбогддог. Илүү дэлгэрэнгүй мэдээллийг ng_l2cap(4) гарын авлагын хуудаснаас лавлана уу.

Ашигтай тушаал бол бусад хэрэгслүүд руу ping хийхэд хэрэглэгддэг l2ping(8) тушаал юм. Зарим нэг Bluetooth шийдлүүд нь тэдэн руу илгээсэн бүх өгөгдлийг буцаахгүй байж болох учраас дараах жишээн дээрх `0 bytes` гэдэг нь хэвийн юм.

```
# l2ping -a 00:80:37:29:19:a4
0 bytes from 0:80:37:29:19:a4 seq_no=0 time=48.633 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=1 time=37.551 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=2 time=28.324 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=3 time=46.150 ms result=0
```

l2control(8) хэрэгсэл нь L2CAP цэгүүд дээр төрөл бүрийн үйлдлүүдийг хийдэг. Энэ жишээ нь логик холболтуудын (сувгууд) жагсаалт болон локал төхөөрөмжийн хувьд үндсэн зурвасын жагсаалтыг хэрхэн авахыг үзүүлж байна:

```
% l2control -a 00:02:72:00:d4:1a read_channel_list
L2CAP channels:
Remote BD_ADDR     SCID/ DCID   PSM  IMTU/ OMTU State
00:07:e0:00:0b:ca    66/   64     3   132/  672 OPEN
% l2control -a 00:02:72:00:d4:1a read_connection_list
L2CAP connections:
Remote BD_ADDR    Handle Flags Pending State
00:07:e0:00:0b:ca     41 O           0 OPEN
```

Өөр нэг оношлогооны хэрэгсэл бол btsockstat(1) юм. Энэ нь netstat(1)-ийн хийдэгтэй төстэйг хийдэг, гэхдээ зөвхөн Bluetooth сүлжээтэй холбоотой өгөгдлийн бүтцүүдийн хувьд хийдэг. Доорх жишээ нь дээрх l2control(8)-ийн нэгэн адил логик холболтыг харуулж байна.

```
% btsockstat
Active L2CAP sockets
PCB      Recv-Q Send-Q Local address/PSM       Foreign address   CID   State
c2afe900      0      0 00:02:72:00:d4:1a/3     00:07:e0:00:0b:ca 66    OPEN
Active RFCOMM sessions
L2PCB    PCB      Flag MTU   Out-Q DLCs State
c2afe900 c2b53380 1    127   0     Yes  OPEN
Active RFCOMM sockets
PCB      Recv-Q Send-Q Local address     Foreign address   Chan DLCI State
c2e8bc80      0    250 00:02:72:00:d4:1a 00:07:e0:00:0b:ca 3    6    OPEN
```

### 32.4.5. RFCOMM Протокол

RFCOMM протокол нь L2CAP протоколын дээгүүр цуваа портуудыг эмуляц хийх боломжийг хангадаг. Энэ протокол нь ETSI стандарт TS 07.10 дээр суурилсан юм. RFCOMM нь RS-232 (EIATIA-232-E) цуваа портуудын 9 замыг эмуляц хийх нэмэлт бэлтгэл бүхий ердийн тээвэрлэх протокол юм. RFCOMM протокол нь хоёр Bluetooth төхөөрөмжийн хооронд 60 хүртэлх зэрэг холболтуудыг (RFCOMM сувгууд) дэмждэг.

RFCOMM-ийн хувьд өөр өөр төхөөрөмжүүд (холбооны төгсгөлийн цэгүүд) дээр ажиллаж байгаа хоёр програм болон тэдгээрийн хоорондын холбооны сегмент холбооны бүрэн замд ордог. RFCOMM нь төхөөрөмжүүд дээр байгаа цуваа портуудыг ашигладаг програмуудад зориулагдсан юм. Холбооны сегмент нь нэг төхөөрөмжөөс нөгөө уруу холбогдсон (шууд холболт) Bluetooth холбоос юм.

RFCOMM нь шууд холболтын үед төхөөрөмжүүдийн хоорондох холболт эсвэл сүлжээний хувьд төхөөрөмж болон модемийн хоорондох холболтод зөвхөн санаа тавьдаг. RFCOMM нь нэг талдаа Bluetooth утасгүй технологийг ашиглаж холбогддог бөгөөд нөгөө талдаа утастай интерфэйсээр хангадаг модулиуд зэрэг бусад тохиргоонуудыг дэмждэг.

FreeBSD дээр RFCOMM протокол нь Bluetooth сокетуудын давхаргад хийгддэг.

### 32.4.6. Төхөөрөмжүүдийг хослох

Анхдагчаар Bluetooth холбоонд нэвтрэлт танилт хийгддэггүй бөгөөд ямар ч төхөөрөмж ямар ч төхөөрөмжтэй ярилцаж чаддаг. Bluetooth төхөөрөмж (жишээ нь үүрэн утас) тухайн нэг үйлчилгээг (жишээ нь Dial-Up үйлчилгээ) хангахын тулд нэвтрэлт танилтыг шаарддаг байхаар байж болно. Bluetooth нэвтрэлт танилт нь хэвийн үед *PIN кодуудаар* хийгддэг. Хоёр төхөөрөмжийн хувьд хэрэглэгч адил PIN кодыг оруулах шаардлагатай. Хэрэглэгч PIN код оруулсны дараа хоёр төхөөрөмж *холболтын түлхүүр* үүсгэнэ. Түүний дараа холболтын түлхүүр нь уг төхөөрөмжүүд дээрээ эсвэл байнгын хадгалалтад хадгалагдаж болно. Дараагийн удаа хоёр төхөөрөмж нь урьд нь үүсгэсэн холболтын түлхүүрээ ашиглах болно. Энэ тайлбарласан процедурыг *pairing* буюу *хослох* гэж нэрлэдэг. Ямар нэг төхөөрөмж холболтын түлхүүрийг гээх юм бол хослолтыг дахин хийх ёстой.

hcsecd(8) дэмон нь бүх Bluetooth нэвтрэлт танилтын хүсэлтүүдтэй ажиллах үүрэгтэй. Анхдагч тохиргооны файл нь /etc/bluetooth/hcsecd.conf юм. Дурын "1234" гэж тохируулагдсан PIN кодтой үүрэн утасны жишээ хэсгийг доор үзүүлэв:

```
device {
        bdaddr  00:80:37:29:19:a4;
        name    "Pav's T39";
        key     nokey;
        pin     "1234";
      }
```

PIN код дээр хязгаарлалт байдаггүй (уртаас гадна). Зарим төхөөрөмжүүдэд (жишээ нь Bluetooth чихэвчнүүд) тогтмол PIN код цуг бүтээгдсэн байж болох юм. `-d` тохируулга нь hcsecd(8) дэмонг нүүрэнд үлдэж ажиллахыг заадаг бөгөөд ингэснээр юу болж байгааг харах

боломжтой юм. Хослолыг хүлээн авч алсын төхөөрөмж уруу Bluetooth холболт эхлүүлэхээр алсын төхөөрөмжийг тохируулна. Алсын төхөөрөмж нь хослол хийлтийг хүлээн авсан гэдгээ хэлж PIN код хүсэх ёстой. hcsecd.conf файлд байгаатай адил PIN код оруулаарай. Одоо таны PC болон алсын төхөөрөмж хосолсон байна. Өөрөөр та хослол хийлтийг алсын төхөөрөмж дээр эхлүүлж болно.

hcsecd-г систем эхлэхэд автоматаар эхлүүлэхийн тулд дараах мөрийг /etc/rc.conf файлд нэмэн хийж болно:

```
hcsecd_enable="YES"
```

hcsecd дэмоны гаралтын жишээг доор үзүүлэв:

```
hcsecd[16484]: Got Link_Key_Request event from 'ubt0hci', remote bdaddr
0:80:37:29:19:a4
hcsecd[16484]: Found matching entry, remote bdaddr 0:80:37:29:19:a4, name 'Pav's T39',
link key doesn't exist
hcsecd[16484]: Sending Link_Key_Negative_Reply to 'ubt0hci' for remote bdaddr
0:80:37:29:19:a4
hcsecd[16484]: Got PIN_Code_Request event from 'ubt0hci', remote bdaddr
0:80:37:29:19:a4
hcsecd[16484]: Found matching entry, remote bdaddr 0:80:37:29:19:a4, name 'Pav's T39',
PIN code exists
hcsecd[16484]: Sending PIN_Code_Reply to 'ubt0hci' for remote bdaddr 0:80:37:29:19:a4
```

### 32.4.7. Service Discovery Protocol (SDP) буюу Үйлчилгээ Илрүүлэх Протокол

Service Discovery Protocol (SDP) буюу Үйлчилгээ Илрүүлэх Протокол нь сервер програмуудын үзүүлдэг үйлчилгээнүүдийн байгаа эсэх болон тэдгээр үйлчилгээнүүдийн шинж чанаруудыг илрүүлэх боломжийг клиент програмуудад олгодог. Үйлчилгээний шинж чанарууд нь санал болгосон үйлчилгээний төрөл эсвэл ангилал болон үйлчилгээг хэрэглэхэд шаардагдах арга зам юм уу эсвэл протоколын мэдээллийг агуулдаг.

SDP-д SDP сервер болон SDP клиентийн хоорондох холбоо ордог. Сервер нь сервертэй холбоотой үйлчилгээнүүдийн шинж чанаруудыг тайлбарладаг үйлчилгээний бичлэгүүдийн жагсаалтыг арчилж байдаг. Үйлчилгээний бичлэг бүр ганц үйлчилгээний талаар мэдээллийг агуулдаг. SDP серверийн арчилж байдаг үйлчилгээний бичлэгээс клиент SDP хүсэлт илгээн мэдээллийг авч болно. Хэрэв клиент эсвэл клиенттэй холбоотой програм нь үйлчилгээг ашиглахаар шийдвэл үйлчилгээг хэрэглэхийн тулд үйлчилгээ үзүүлэгч уруу тусдаа холболт нээх ёстой. SDP нь үйлчилгээнүүд болон тэдгээрийн шинж чанаруудыг илрүүлэх арга замаар хангадаг боловч тэдгээр үйлчилгээнүүдийг хэрэглэх арга замуудаар хангадаггүй юм.

Хэвийн үед SDP клиент нь үйлчилгээнүүдийн зарим хүссэн онцгой шинжүүд дээр тулгуурлан үйлчилгээнүүдийг хайдаг. Гэхдээ үйлчилгээнүүдийн талаар урьд нь ямар ч мэдээлэл байхгүй байхад SDP серверийн үйлчилгээний бичлэгүүдээр тайлбарлагдсан

үйлчилгээнүүдийн ямар төрлүүд байгааг олохыг хүсэх үеүүд байдаг. Санал болгосон дурын үйлчилгээнүүдийг хайх процесс нь *browsing* буюу *үзэх* гэгддэг.

Bluetooth SDP сервер sdpd(8) болон тушаалын мөрийн клиент sdpcontrol(8) нь стандарт FreeBSD суулгацад орсон байдаг. Дараах жишээ нь SDP үзэх хүсэлтийг хэрхэн хийж байгаа харуулж байна.

```
% sdpcontrol -a 00:01:03:fc:6e:ec browse
Record Handle: 00000000
Service Class ID List:
        Service Discovery Server (0x1000)
Protocol Descriptor List:
        L2CAP (0x0100)
                Protocol specific parameter #1: u/int/uuid16 1
                Protocol specific parameter #2: u/int/uuid16 1

Record Handle: 0x00000001
Service Class ID List:
        Browse Group Descriptor (0x1001)

Record Handle: 0x00000002
Service Class ID List:
        LAN Access Using PPP (0x1102)
Protocol Descriptor List:
        L2CAP (0x0100)
        RFCOMM (0x0003)
                Protocol specific parameter #1: u/int8/bool 1
Bluetooth Profile Descriptor List:
        LAN Access Using PPP (0x1102) ver. 1.0
```

i. гэх мэт байна. Үйлчилгээ болгон шинж чанаруудтай байгааг анхааралаарай (жишээ нь RFCOMM суваг). Үйлчилгээнээс хамаараад та зарим нэг шинж чанаруудын талаар тэмдэглэгээ хийж авах хэрэгтэй болж болох юм. Зарим Bluetooth шийдлүүд нь үйлчилгээ үзэх боломжийг дэмждэггүй бөгөөд хоосон жагсаалт буцааж болох юм. Энэ тохиолдолд тодорхой үйлчилгээг хайх боломжтой байдаг. Доорх жишээ нь OBEX-ийн Обьект Түлхэх (OPUSH) үйлчилгээг хэрхэн хайхыг үзүүлж байна:

```
% sdpcontrol -a 00:01:03:fc:6e:ec search OPUSH
```

FreeBSD дээр Bluetooth клиентүүдэд үйлчилгээнүүдийг санал болохдоо sdpd(8) серверийн тусламжтайгаар хийдэг. /etc/rc.conf файлд дараах мөрийг нэмж болно:

```
sdpd_enable="YES"
```

Дараа нь sdpd дэмонг ингэж эхлүүлж болно:

```
# service sdpd start
```

Алсын клиентүүдэд Bluetooth үйлчилгээг үзүүлэхийг хүссэн локал серверийн програм нь үйлчилгээг SDP дэмонд бүртгэх болно. Ийм програмуудын нэг нь rfcomm_pppd(8) юм. Эхэлснийхээ дараа энэ нь Bluetooth LAN үйлчилгээг локал SDP дэмонд бүртгэх болно.

Локал SDP серверт бүртгэсэн үйлчилгээнүүдийн жагсаалтыг локал хяналтын сувгаар SDP-ийн үзэх хүсэлтийг илгээн авч болно:

```
# sdpcontrol -l browse
```

### 32.4.8. Dial-Up сүлжээ (DUN) ба PPP ашиглах Сүлжээний Хандалт (LAN) хувийн тохиргоонууд

Dial-Up сүлжээ (DUN) хувийн тохиргоо нь модемууд болон үүрэн утаснуудтай ихэвчлэн ашиглагддаг. Энэ хувийн тохиргоонд хамаарах тохиолдлуудыг доор дурдав:

- үүрэн утас эсвэл модемийг компьютер дээрээ Интернэтийн хандалтын сервер уруу залгаж холбогдох юм уу эсвэл бусад dial-up үйлчилгээнүүдэд хэрэглэхээр утасгүй модем маягаар ашиглах;
- үүрэн утас эсвэл модемийг компьютер дээрээ өгөгдлийн дуудлагуудыг хүлээн авахад ашиглах.

PPP ашиглах Сүлжээний Хандалт (LAN) хувийн тохиргоо дараах тохиолдлуудад ашиглагдаж болно:

- Ганц Bluetooth төхөөрөмжид зориулсан LAN хандалт;
- Олон Bluetooth төхөөрөмжид зориулсан LAN хандалт;
- PC-ээс PC уруу (цуваа кабелийн эмуляцаар PPP сүлжээ ашиглан).

FreeBSD дээр энэ хоёр хувийн тохиргоо нь ppp(8) болон rfcomm_pppd(8) програмуудаар хийгддэг. rfcomm_pppd(8) нь RFCOMM Bluetooth холболтыг PPP-ийн ажиллаж чадах ямар нэгэн зүйл болгож хувиргадаг гүйцэтгэл хялбаршуулагч юм. Аль ч хувийн тохиргоог ашиглахаасаа өмнө /etc/ppp/ppp.conf файлд шинэ PPP хаяг үүсгэгдсэн байх ёстой. Жишээнүүдийн талаар rfcomm_pppd(8) гарын авлагаас лавлана уу.

Дараах жишээн дээр rfcomm_pppd(8) нь DUN RFCOMM суваг дээр BD_ADDR 00:80:37:29:19:a4 хаягтай алсын төхөөрөмж уруу RFCOMM холболт хийхэд ашиглагдах болно. RFCOMM сувгийн дугаарыг алсын төхөөрөмжөөс SDP-ээр авах болно. RFCOMM сувгийг гараар зааж өгөх боломжтой бөгөөд энэ тохиолдолд rfcomm_pppd(8) нь SDP хүсэлт хийхгүй байх болно. Алсын төхөөрөмж дээр RFCOMM сувгийг олохын тулд sdpcontrol(8)-г ашиглаарай.

```
# rfcomm_pppd -a 00:80:37:29:19:a4 -c -C dun -l rfcomm-dialup
```

PPP ашиглах Сүлжээний Хандалтын (LAN) үйлчилгээг хангахын тулд sdpd(8) сервер

ажиллаж байх ёстой. LAN клиентүүдэд зориулсан шинэ оруулгууд /etc/ppp/ppp.conf файлд үүсгэгдсэн байх ёстой. Жишээнүүдийн талаар rfcomm_pppd(8) гарын авлагын хуудаснаас лавлана уу. Төгсгөлд нь RFCOMM PPP серверийг зөв RFCOMM сувгийн дугаар дээр эхлүүлнэ. RFCOMM PPP сервер нь Bluetooth LAN үйлчилгээг локал SDP дэмонд автоматаар бүртгэх болно. Доорх жишээ нь RFCOMM PPP серверийг хэрхэн эхлүүлэхийг үзүүлж байна.

```
# rfcomm_pppd -s -C 7 -l rfcomm-server
```

### 32.4.9. OBEX Object Push (OPUSH) буюу OBEX Обьект Түлхэх хувийн тохиргоо

OBEX нь хөдөлгөөнт төхөөрөмжүүдийн хооронд энгийн файл дамжуулалт хийхэд зориулагдсан өргөн ашиглагддаг протокол юм. Үүний гол хэрэглээ нь хэт ягаан туяаны холбоо бөгөөд зөөврийн компьютерууд эсвэл PDA-уудын хооронд ердийн файл дамжуулахад, нэрийн хуудас эсвэл цагалбарыг үүрэн утас болон PIM програмуудтай бусад төхөөрөмжүүдийн хооронд илгээхэд хэрэглэгддэг.

OBEX сервер болон клиент нь гуравдагч талын багц obexapp хэлбэрээр хийгдсэн байдаг бөгөөд энэ нь comms/obexapp порт хэлбэрээр байдаг.

OBEX клиент нь OBEX серверт обьектуудыг түлхэж оруулах буюу/эсвэл татахад хэрэглэгддэг. Обьект нь жишээ нь нэрийн хуудас юм уу эсвэл уулзалт байж болно. OBEX клиент нь алсын төхөөрөмжөөс SDP-ээр RFCOMM сувгийн дугаарыг авч болно. RFCOMM сувгийн дугаарын оронд үйлчилгээний нэрийг зааж үүнийг хийж болно. Дэмжигдсэн үйлчилгээний нэрсэд: IrMC, FTRN, болон OPUSH ордог. RFCOMM сувгийг дугаар болгон зааж өгөх боломжтой байдаг. Төхөөрөмжийн мэдээллийн обьектийг үүрэн утаснаас татаж авч байгаа болон шинэ обьектийг (нэрийн хуудас) утасны сан уруу хийж байгаа OBEX сессийн жишээг доор үзүүлэв.

```
% obexapp -a 00:80:37:29:19:a4 -C IrMC
obex> get telecom/devinfo.txt devinfo-t39.txt
Success, response: OK, Success (0x20)
obex> put new.vcf
Success, response: OK, Success (0x20)
obex> di
Success, response: OK, Success (0x20)
```

OBEX-ийн Обьект Түлхэх үйлчилгээг хангахын тулд sdpd(8) сервер ажиллаж байх ёстой. Бүх ирж байгаа обьектууд хадгалагдах root хавтас үүсгэгдэх ёстой. root сангийн анхдагч зам нь /var/spool/obex байна. Төгсгөлд нь OBEX серверийг зөв RFCOMM сувгийн дугаар дээр эхлүүлнэ. OBEX сервер нь OBEX-ийн Обьект Түлхэх үйлчилгээг локал SDP дэмонд автоматаар бүртгүүлэх болно. Доорх жишээ нь OBEX серверийг хэрхэн эхлүүлэхийг харуулж байна.

```
# obexapp -s -C 10
```

### 32.4.10. Serial Port Profile (SPP) буюу Цуваа портын хувийн тохиргоо

Serial Port Profile (SPP) буюу Цуваа портын хувийн тохиргоо нь Bluetooth төхөөрөмжүүдэд RS232 (эсвэл түүнтэй төстэй) цуваа кабелийн эмуляц хийхийг зөвшөөрдөг. Энэ хувийн тохиргоонд хамаатай тохиолдол нь кабелийн оронд виртуал цуваа портын хийсвэрлэлтийн тусламжтай Bluetooth-ийг ашигладаг хуучин програмуудтай харьцдаг.

rfcomm_sppd(1) хэрэгсэл нь цуваа портын хувийн тохиргоог хийдэг. Псевдо tty нь виртуал цуваа портын хийсвэрлэлт болон ашиглагддаг. Доорх жишээ нь алсын төхөөрөмжийн цуваа портын үйлчилгээ уруу хэрхэн холбогдохыг харуулж байна. Та RFCOMM сувгийг заах шаардлагагүйг санаарай - rfcomm_sppd(1) нь алсын төхөөрөмжөөс SDP-ээр авч чаддаг. Хэрэв та үүнийг дарж өөрчлөхийг хүсвэл тушаалын мөрөнд RFCOMM сувгийг зааж өгөх хэрэгтэй.

```
# rfcomm_sppd -a 00:07:E0:00:0B:CA -t /dev/ttyp6
rfcomm_sppd[94692]: Starting on /dev/ttyp6...
```

Холбогдсоны дараа псевдо tty нь цуваа порт шиг ашиглагдаж болно:

```
# cu -l ttyp6
```

### 32.4.11. Алдааг олж засварлах

#### 32.4.11.1. Алсын төхөөрөмж холбогдож чадахгүй байх

Зарим нэг хуучин Bluetooth төхөөрөмжүүд нь үүрэг шилжүүлэлтийг дэмждэггүй. Анхдагчаар FreeBSD нь шинэ холболтыг хүлээн авахдаа үүргийг өөрчилж мастер болохыг оролддог. Үүнийг дэмждэггүй төхөөрөмжүүд нь холбогдож чаддаггүй. Шинэ холболт хийгдэхэд үүрэг шилжүүлэлт хийгддэгийг санаарай. Тийм учраас алсын төхөөрөмжөөс үүрэг шилжүүлэлтийг дэмждэг эсэхийг нь асуух боломжгүй юм. Локал тал дээрээ үүрэг шилжүүлэлтийг хаах HCI тохируулга байдаг:

```
# hccontrol -n ubt0hci write_node_role_switch 0
```

#### 32.4.11.2. Ямар нэгэн юм буруу болоод байна, би яг юу болоод байгааг харж болох уу?

Тиймээ, та харж болно. comms/hcidump порт хэлбэрээр байдаг гуравдагч талын багц hcidump-г ашиглана. hcidump хэрэгсэл нь tcpdump(1)-тай төстэй. Энэ нь Bluetooth пакетуудын агуулгыг терминал дээр харуулж Bluetooth пакетуудыг файл уруу гаргахад хэрэглэгдэж болно.

## 32.5. Гүүр

### 32.5.1. Танилцуулга

Заримдаа нэг физик сүлжээг (Ethernet сегмент зэрэг) IP дэд сүлжээнүүд үүсгэж

сегментүүдийг хооронд нь чиглүүлэгчээр цугт нь холбож ашиглалгүйгээр хоёр тусдаа сүлжээний сегмент болгох нь ашигтай байдаг. Ийм маягаар хоёр сүлжээг хооронд нь холбодог төхөөрөмжийг "bridge" буюу "гүүр" гэдэг. Хоёр сүлжээний интерфэйс карттай FreeBSD систем гүүр маягаар ажиллаж чаддаг.

Гүүр нь өөрийн сүлжээний интерфэйс бүрийн төхөөрөмжийн MAC давхаргын хаягуудыг (Ethernet хаягууд) сурч ажилладаг. Түүний эх болон төгсгөл нь зөвхөн өөр өөр сүлжээнд байгаа тохиолдолд хоёр сүлжээний хооронд урсгалыг дамжуулдаг.

Олон талаараа гүүр нь маш цөөн порттой Ethernet шилжүүлэгчтэй адил юм.

## 32.5.2. Гүүр хийхэд тохирох тохиолдлууд

Өнөөдөр гүүр ашиглагддаг олон нийтлэг тохиолдол байдаг.

#### 32.5.2.1. Сүлжээнүүдийг холбох нь

Гүүрний үндсэн үйлдэл нь хоёр буюу түүнээс олон сүлжээний сегментүүдийг хооронд нь холбох явдал юм. Кабелийн хязгаарлалт, галт хана хийх эсвэл виртуал машины интерфэйс зэрэг псевдо сүлжээнүүдийг холбох зэрэг сүлжээний энгийн төхөөрөмжийн оронд хост дээр тулгуурласан гүүрийг ашиглах шалтгаан олон байдаг. Гүүр нь бас hostap горимд ажиллаж байгаа утасгүй сүлжээний интерфэйсийг утастай сүлжээ рүү холбож хандалтын цэг маягаар ажиллах чадвартай.

#### 32.5.2.2. Шүүх/урсгал хэлбэржүүлэх галт хана

Нийтлэг тохиолдол бол чиглүүлэлт юм уу эсвэл сүлжээний хаягийн хөрвүүлэлттүй (NAT) галт ханын ажиллагаа шаардлагатай тохиолдол юм.

Үүний жишээ нь DSL юм уу эсвэл ISDN-ээр ISP уруугаа холбогдсон жижиг компани юм. Тэд 13 ширхэг гаднаас хандах боломжтой IP хаягийг өөрсдийн ISP-ээс авдаг бөгөөд сүлжээндээ 10 PC-тэй. Энэ тохиолдолд чиглүүлэгч дээр тулгуурласан галт хана нь дэд сүлжээний асуудлуудаас болоод төвөгтэй байна.

Гүүр дээр тулгуурласан галт ханыг тохируулж тэдний DSL/ISDN чиглүүлэгчийн замд ямар нэгэн IP хаяглалтын асуудалгүйгээр тавьж болно.

#### 32.5.2.3. Сүлжээ сонсогч (Network tap)

Гүүр нь сүлжээний хоёр сегментийг холбож тэдгээрийн хооронд дамжиж байгаа бүх Ethernet урсгалыг шалгахад хэрэглэгдэж болно. Энэ нь гүүр интерфэйс дээр bpf(4)/tcpdump(1) ашиглах юм уу эсвэл бүх урсгалын хуулбарыг нэмэлт интерфэйс (span порт) уруу илгээх замаар байж болно.

#### 32.5.2.4. Түвшин 2 VPN

Хоёр Ethernet сүлжээ нь IP холбоосын дагуу EtherIP туннель юм уу эсвэл OpenVPN зэрэг tap(4) дээр тулгуурласан шийдлүүдийн тусламжтайгаар холбогдож болно.

#### 32.5.2.5. Түвшин 2 Давхцал

Сүлжээ нь хоорондоо олон холбоосоор холбогдож давхацсан замуудыг хаахын тулд Spanning Tree протоколыг ашиглаж болно. Ethernet сүлжээ нь зөв ажилладаг байхын тулд хоёр төхөөрөмжийн хооронд зөвхөн нэг идэвхтэй зам байх ёстой байдаг бөгөөд Spanning Tree нь давталтыг илрүүлж давхацсан холбоосуудыг хаалттай төлөвт оруулдаг. Аль нэг идэвхтэй холбоос амжилтгүй болсон тохиолдолд уг протокол өөр модыг тооцоолж сүлжээн дэх бүх цэгүүдэд хүрэх холболтыг сэргээхийн тулд аль нэг хаалттай замыг дахин идэвхжүүлдэг.

### 32.5.3. Цөмийн тохиргоо

Энэ хэсэг нь if_bridge(4) гүүрний шийдлийн тухай өгүүлэх болно, netgraph гүүрний драйвер бас байдаг бөгөөд илүү дэлгэрэнгүй мэдээллийг ng_bridge(4) гарын авлагын хуудаснаас үзнэ үү.

Гүүрний драйвер нь цөмийн модуль бөгөөд ifconfig(8)-оор гүүр интерфэйсийг үүсгэх үед автоматаар дуудагддаг. Өөрийн цөмийн тохиргооны файлд `device if_bridge` гэж нэмэн гүүрийг цөмд эмхэтгэж болно.

pfil(9) тогтолцоогоор холбогддог галт ханын дурын багцтай пакет шүүлтийг ашиглаж болдог. Галт хана нь модуль хэлбэрээр дуудагдах юм уу эсвэл цөмд эмхэтгэгдэж болно.

Гүүр нь altq(4) эсвэл dummynet(4)-тэй цуг урсгал хэлбэржүүлэгч болон ашиглагдаж болно.

### 32.5.4. Гүүрийг идэвхжүүлэх нь

Интерфэйс хуулбарлалтыг ашиглан гүүрийг үүсгэдэг. Гүүрийг үүсгэхийн тулд ifconfig(8)-ийг ашиглана. Хэрэв гүүрний драйвер цөмд байхгүй бол автоматаар дуудагддаг.

```
# ifconfig bridge create
bridge0
# ifconfig bridge0
bridge0: flags=8802<BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
        ether 96:3d:4b:f1:79:7a
        id 00:00:00:00:00:00 priority 32768 hellotime 2 fwddelay 15
        maxage 20 holdcnt 6 proto rstp maxaddr 100 timeout 1200
        root id 00:00:00:00:00:00 priority 0 ifcost 0 port 0
```

Гүүр интерфэйс үүсч түүнд санамсаргүйгээр үүсгэгдсэн Ethernet хаяг автоматаар өгөгддөг. `maxaddr` болон `timeout` нэмэлт өгөгдлүүд нь өөрийн дамжуулах хүснэгтэд хичнээн MAC хаягийг хадгалах болон сүүлд харагдсанаасаа хойш оруулга бүр хичнээн секундын дараа устгагдах вэ гэдгийг хянадаг. Бусад нэмэлт өгөгдлүүд нь Spanning Tree хэрхэн ажиллахыг хянадаг.

Гишүүн сүлжээний интерфэйсийг гүүрэнд нэмэх хэрэгтэй. Гүүрний хувьд пакетуудыг дамжуулахын тулд бүх гишүүн интерфэйсүүд болон гүүр өөрөө идэвхжсэн байх шаардлагатай:

```
# ifconfig bridge0 addm fxp0 addm fxp1 up
# ifconfig fxp0 up
# ifconfig fxp1 up
```

Гүүр нь одоо Ethernet хүрээнүүдийг fxp0 болон fxp1-ийн хооронд дамжуулж байна. Гүүр нь ачаалах үед үүсгэгдэх /etc/rc.conf-ийн тохиргоо ийм байна:

```
cloned_interfaces="bridge0"
ifconfig_bridge0="addm fxp0 addm fxp1 up"
ifconfig_fxp0="up"
ifconfig_fxp1="up"
```

Хэрэв гүүр хостод IP хаяг хэрэгтэй бол үүнийг тохируулах зөв газар бол гишүүн интерфэйсийн аль нэг биш харин гүүр интерфэйс өөрөө юм. Үүнийг статикаар эсвэл DHCP-ээр тохируулж болно:

```
# ifconfig bridge0 inet 192.168.0.1/24
```

Гүүр интерфэйсд IPv6 хаягийг өгч бас болно.

### 32.5.5. Галт хана

Пакет шүүлт идэвхжсэн тохиолдолд гүүр хийгдсэн пакетууд нь гарч байгаа интерфэйс ба гүүр интерфэйс дээрээ гарах шүүлтээр, тохирох интерфэйс дээрээ орох шүүлтээр дамждаг. Аль ч шатыг хааж болдог. Пакетийн урсгалын чиглэл нь чухал бол гүүрэн дээр галт хана хийснээс гишүүн интерфэйсүүд дээр хийсэн нь дээр байдаг.

Гүүр нь IP бус ба ARP пакетуудыг дамжуулах болон IPFW-ийн тусламжтай хийгдэх хоёрдугаар түвшний галт хананд зориулсан хэд хэдэн тохируулж болох тохируулгуудтай байдаг. Дэлгэрэнгүй мэдээллийг if_bridge(4)-ээс үзнэ үү.

### 32.5.6. Spanning Tree

Гүүрний драйвер нь хуучин Spanning Tree Протоколтой (STP) нийцтэй Rapid Spanning Tree Protocol (RSTP эсвэл 802.1w) буюу Түргэн Spanning Tree Протоколыг хийж гүйцэтгэсэн байдаг. Spanning Tree нь сүлжээн дэх давталтуудыг илрүүлж арилгахад хэрэглэгддэг. RSTP нь хуучин STP-г бодох юм бол илүү түргэн нийлэх боломжийг олгодог. Энэ протокол нь давхцал үүсгэлгүйгээр дамжуулалтад хурдан шилжихийн тулд хөрш шилжүүлэгчидтэйгээ мэдээлэл солилцдог. FreeBSD нь RSTP болон STP-г ажиллах горим хэлбэрээр дэмждэг бөгөөд RSTP нь анхдагч горим юм.

Spanning Tree-г `stp` тушаал ашиглан гишүүн интерфэйсүүд дээр идэвхжүүлж болно. Одоо fxp0 болон fxp1 интерфэйсүүдтэй гүүрний хувьд STP-г идэвхжүүлэхийн тулд доор дурдсаныг хийнэ:

```
# ifconfig bridge0 stp fxp0 stp fxp1
bridge0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        ether d6:cf:d5:a0:94:6d
        id 00:01:02:4b:d4:50 priority 32768 hellotime 2 fwddelay 15
        maxage 20 holdcnt 6 proto rstp maxaddr 100 timeout 1200
        root id 00:01:02:4b:d4:50 priority 32768 ifcost 0 port 0
        member: fxp0 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 3 priority 128 path cost 200000 proto rstp
                role designated state forwarding
        member: fxp1 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 4 priority 128 path cost 200000 proto rstp
                role designated state forwarding
```

Энэ гүүр нь `00:01:02:4b:d4:50` гэсэн ID болон `32768` гэсэн ээлж бүхий spanning tree-тэй байна. `root id` адил байгаа нь модны хувьд root гүүр гэдгийг илтгэж байна.

Сүлжээн дэх өөр нэг гүүр бас идэвхжсэн spanning tree-тэй байна:

```
bridge0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        ether 96:3d:4b:f1:79:7a
        id 00:13:d4:9a:06:7a priority 32768 hellotime 2 fwddelay 15
        maxage 20 holdcnt 6 proto rstp maxaddr 100 timeout 1200
        root id 00:01:02:4b:d4:50 priority 32768 ifcost 400000 port 4
        member: fxp0 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 4 priority 128 path cost 200000 proto rstp
                role root state forwarding
        member: fxp1 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 5 priority 128 path cost 200000 proto rstp
                role designated state forwarding
```

`root id 00:01:02:4b:d4:50 priority 32768 ifcost 400000 port 4` мөр нь root гүүр нь дээр дурдсантай адил `00:01:02:4b:d4:50` бөгөөд энэ гүүрнээс `400000` гэсэн замын өртөгтэй, root гүүр рүү хүрэх зам нь fxp0 болох `port 4`-өөр дамжина гэдгийг үзүүлж байна.

## 32.5.7. Гүүрний нэмэлт тохиргоо

### 32.5.7.1. Урсгалыг бүтээх/шинэчлэх

Гүүр нь монитор буюу хянах горимыг дэмждэг бөгөөд энэ горимд пакетууд нь bpf(4) процесс хийгдсэний дараа хаягдаж цааш процесс хийгдэхгүй эсвэл дамжуулагдахгүй болдог. Хоёр буюу түүнээс дээш тооны интерфэйсүүдийн оролтыг нэгтгэж нэг bpf(4) урсгал руу гаргахад үүнийг ашиглаж болно. RX/TX дохионуудыг гадагш хоёр тусдаа интерфэйсээр дамжуулах сүлжээний сонсогчид зориулсан урсгал бүтээхэд энэ нь ашигтай байдаг.

Сүлжээний дөрвөн интерфэйсээс оролтыг нэг урсгал уруу уншихын тулд:

```
# ifconfig bridge0 addm fxp0 addm fxp1 addm fxp2 addm fxp3 monitor up
```

```
# tcpdump -i bridge0
```

#### 32.5.7.2. Span портууд

Гүүрний хүлээн авсан Ethernet хүрээ бүрийн хуулбар томилогдсон span порт уруу дамждаг. Гүүрэн дээр тохируулагдсан span портуудын тоо хязгааргүй байдаг, хэрэв интерфэйс нь span порт гэж томилогдсон бол энэ нь ердийн гүүрний порт маягаар ашиглагдах боломжгүй байж болох юм. Энэ нь гүүрний аль нэг span порт руу холбогдсон өөр нэг хост дээрээс гүүр хийгдсэн сүлжээг идэвхгүйгээр хулгайгаар үзэхэд их ашигтай байдаг.

Бүх хүрээнүүдийн хуулбарыг fxp4 гэж нэрлэгдсэн интерфэйс рүү илгээхийн тулд:

```
# ifconfig bridge0 span fxp4
```

#### 32.5.7.3. Хувийн интерфэйсүүд

Хувийн интерфэйс нь бусад хувийн интерфэйсүүд рүү ямар ч урсгалыг дамжуулдаггүй. Урсгал нь ямар нэг нөхцөлгүйгээр хаагддаг бөгөөд ARP зэрэг ямар ч Ethernet хүрээнүүд дамжуулагдахгүй. Хэрэв урсгал нь сонголтын дагуу хаагдах хэрэгтэй бол үүний оронд галт хана ашиглах ёстой байдаг.

#### 32.5.7.4. Наалттай интерфэйсүүд

Хэрэв гүүрний гишүүн интерфэйс нь наалттай гэж тэмдэглэгдсэн бол динамикаар тогтоосон хаягийн оруулгуудыг дамжуулалтын кэшд ороход статик гэж үздэг. Хаяг өөр интерфэйс дээр үзэгдсэн байлаа ч гэсэн наалттай оруулгууд нь хэзээ ч кэшээс гарч хуучирдаггүй эсвэл солигддоггүй. Энэ нь дамжуулалтын хүснэгтийг урьдчилан нутагшуулах шаардлагагүй болгож статик хаягийн оруулгуудын ашгийг өгдөг бөгөөд гүүрний тодорхой нэг сегмент дээр тогтоогдсон хэрэглэгчид өөр сегмент рүү тэнүүчилж чаддаггүй.

Наалттай хаягуудыг ашиглах өөр нэг жишээ нь IP хаягийн талбарыг дэмий үрэлгүйгээр хэрэглэгчийн сүлжээнүүд нь тусгаарлагдсан чиглүүлэгчийг үүсгэхийн тулд гүүрийг VLAN-уудтай нэгтгэх явдал юм. `CustomerA` нь `vlan100` дээр `CustomerB` нь `vlan101` дээр гэж үзье. Гүүр нь `192.168.0.1` гэсэн хаягтай бөгөөд бас интернэт чиглүүлэгч юм.

```
# ifconfig bridge0 addm vlan100 sticky vlan100 addm vlan101 sticky vlan101
# ifconfig bridge0 inet 192.168.0.1/24
```

Хоёр клиент хоёулаа `192.168.0.1`-г өөрсдийн анхдагч гарц гэж харах бөгөөд гүүрний кэш нь наалттай болохоор тэд өөрсдийн урсгалыг дундаас нь оруулах гэж нөгөө хэрэглэгчийнхээ MAC хаягийг ашиглан хуурч чадахгүй юм.

VLAN-уудын хоорондох ямар ч холбоог хувийн интерфэйсүүд (эсвэл галт хана) ашиглан хааж болно:

```
# ifconfig bridge0 private vlan100 private vlan101
```

Хэрэглэгчид нь нэг нэгнээсээ бүр мөсөн тусгаарлагдсан бөгөөд бүх /24 хаягийн бүсийг дэд сүлжээ болгон хуваалгүйгээр хуваарилж болно.

#### 32.5.7.5. Хаягийн хязгаарууд

Интерфэйсийн цаана байгаа давтагдашгүй эх MAC хаягуудын тоо хязгаартай байж болно. Тэр хязгаарт хүрэх үед мэдэгдэхгүй эх хаягтай пакетууд нь тухайн үед байгаа хостын кэш оруулгын хугацаа дуусах юм уу эсвэл устгагдах хүртэл хаягдсаар байх болно.

Доор дурдсан жишээ нь vlan100 дээр байгаа CustomerA-д зориулж Ethernet төхөөрөмжүүдийн хамгийн их тоог 10 болгон тохируулж байна.

```
# ifconfig bridge0 ifmaxaddr vlan100 10
```

#### 32.5.7.6. SNMP монитор хийх

Гүүр интерфэйс болон STP параметрүүдийг FreeBSD-ийн үндсэн системд орсон байдаг SNMP демоны тусламжтайгаар монитор хийж болно. Экспорт хийгдсэн гүүрний MIB-үүд нь IETF-ийн стандартуудыг хангаж байдаг. Тийм болохоор дурын SNMP клиент эсвэл монитор хийдэг багцыг өгөгдлийг хүлээн авахад хэрэглэж болно.

Гүүр машин дээр /etc/snmp.config файлд begemotSnmpdModulePath."bridge" = "/usr/lib/snmp_bridge.so" мөрийг тайлбар болгосныг болиулж bsnmpd дэмоныг эхлүүлэх хэрэгтэй. Community буюу нийгмийн нэрс болон хандалтын жагсаалтууд зэрэг бусад тохиргоонуудыг өөрчлөх шаардлагатай байж болно. Дэлгэрэнгүй мэдээллийг bsnmpd(1) болон snmp_bridge(3)-с үзнэ үү.

Доор дурдсан жишээнүүд нь гүүрэнд хүсэлт илгээхийн тулд Net-SNMP програм хангамжийг (net-mgmt/net-snmp) ашиглаж байгаа бөгөөд net-mgmt/bsnmptools портыг бас ашиглаж болно. Гүүрний MIB тодорхойлолтуудыг Net-SNMP руу оруулж ирэхийн тулд SNMP клиент хост дээрээ $HOME/.snmp/snmp.conf файлд доор дурдсан мөрүүдийг нэмэх хэрэгтэй:

```
mibdirs +/usr/shared/snmp/mibs
mibs +BRIDGE-MIB:RSTP-MIB:BEGEMOT-MIB:BEGEMOT-BRIDGE-MIB
```

Ганц гүүрийг IETF BRIDGE-MIB (RFC4188)-ээр монитор хийхийн тулд доор дурдсаныг хийнэ

```
% snmpwalk -v 2c -c public bridge1.example.com mib-2.dot1dBridge
BRIDGE-MIB::dot1dBaseBridgeAddress.0 = STRING: 66:fb:9b:6e:5c:44
BRIDGE-MIB::dot1dBaseNumPorts.0 = INTEGER: 1 ports
BRIDGE-MIB::dot1dStpTimeSinceTopologyChange.0 = Timeticks: (189959) 0:31:39.59 centi-
seconds
BRIDGE-MIB::dot1dStpTopChanges.0 = Counter32: 2
BRIDGE-MIB::dot1dStpDesignatedRoot.0 = Hex-STRING: 80 00 00 01 02 4B D4 50
...
BRIDGE-MIB::dot1dStpPortState.3 = INTEGER: forwarding(5)
BRIDGE-MIB::dot1dStpPortEnable.3 = INTEGER: enabled(1)
```

```
BRIDGE-MIB::dot1dStpPortPathCost.3 = INTEGER: 200000
BRIDGE-MIB::dot1dStpPortDesignatedRoot.3 = Hex-STRING: 80 00 00 01 02 4B D4 50
BRIDGE-MIB::dot1dStpPortDesignatedCost.3 = INTEGER: 0
BRIDGE-MIB::dot1dStpPortDesignatedBridge.3 = Hex-STRING: 80 00 00 01 02 4B D4 50
BRIDGE-MIB::dot1dStpPortDesignatedPort.3 = Hex-STRING: 03 80
BRIDGE-MIB::dot1dStpPortForwardTransitions.3 = Counter32: 1
RSTP-MIB::dot1dStpVersion.0 = INTEGER: rstp(2)
```

dot1dStpTopChanges.0-ийн утга хоёр бөгөөд энэ нь STP гүүрний бүтэц хоёр удаа өөрчлөгдсөн гэдгийг харуулж байна. Бүтцийн өөрчлөлт гэдэг нь сүлжээн дэх нэг буюу олон холбоосууд өөрчлөгдсөн юм уу эсвэл амжилтгүй болсон бөгөөд шинэ мод тооцоологдсон гэсэн үг юм. dot1dStpTimeSinceTopologyChange.0-ийн утга ийм зүйл хэзээ болсныг харуулах болно.

Олон гүүрний интерфэйсийг монитор хийхийн тулд хувийн BEGEMOT-BRIDGE-MIB-г ашиглаж болно:

```
% snmpwalk -v 2c -c public bridge1.example.com
enterprises.fokus.begemot.begemotBridge
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseName."bridge0" = STRING: bridge0
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseName."bridge2" = STRING: bridge2
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseAddress."bridge0" = STRING: e:ce:3b:5a:9e:13
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseAddress."bridge2" = STRING: 12:5e:4d:74:d:fc
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseNumPorts."bridge0" = INTEGER: 1
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseNumPorts."bridge2" = INTEGER: 1
...
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTimeSinceTopologyChange."bridge0" = Timeticks:
(116927) 0:19:29.27 centi-seconds
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTimeSinceTopologyChange."bridge2" = Timeticks:
(82773) 0:13:47.73 centi-seconds
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTopChanges."bridge0" = Counter32: 1
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTopChanges."bridge2" = Counter32: 1
BEGEMOT-BRIDGE-MIB::begemotBridgeStpDesignatedRoot."bridge0" = Hex-STRING: 80 00 00 40
95 30 5E 31
BEGEMOT-BRIDGE-MIB::begemotBridgeStpDesignatedRoot."bridge2" = Hex-STRING: 80 00 00 50
8B B8 C6 A9
```

mib-2.dot1dBridge дэд модоор монитор хийгдэж байгаа гүүрний интерфэйсийг солихын тулд доор дурдсаныг хийнэ:

```
% snmpset -v 2c -c private bridge1.example.com
BEGEMOT-BRIDGE-MIB::begemotBridgeDefaultBridgeIf.0 s bridge2
```

## 32.6. Холбоос нэгтгэлт ба ослыг тойрон гарах (Link Aggregation and Failover)

### 32.6.1. Танилцуулга

lagg(4) интерфэйс нь fault-tolerance буюу алдаанд тэсвэртэй байдал болон өндөр хурдны холбоосуудыг хангах зорилгоор сүлжээний олон интерфэйсүүдийг нэг виртуал интерфэйс болгон нэгтгэх боломжийг олгодог.

### 32.6.2. Ажиллагааны горимууд

**Failover**

Зөвхөн мастер портоор урсгалыг илгээж хүлээн авна. Хэрэв мастер порт байхгүй болвол дараагийн идэвхтэй порт ашиглагдана. Эхний нэмэгдсэн интерфэйс нь мастер порт болох бөгөөд үүний дараа нэмэгдсэнүүд нь ослыг тойрон гарах төхөөрөмжүүд болон ашиглагдах болно. Хэрэв мастер биш порт дээр тойрон гарах үйлдэл хийгдвэл анхдагч порт дахин боломжтой болонгуутаа мастер порт болдог.

**Cisco® Fast EtherChannel®**

Cisco® Fast EtherChannel® (FEC) нь статик тохируулга бөгөөд нөгөө талтайгаа нэгтгэлт хийхэд тохиролцдоггүй эсвэл холбоосыг монитор хийхийн тулд хүрээнүүд солилцдоггүй. Хэрэв шилжүүлэгч LACP-г дэмждэг бол түүнийг ашиглах ёстой.

FEC нь хэш хийгдсэн протоколын толгойн мэдээлэл дээр үндэслэн идэвхтэй портуудаар гарч байгаа урсгалыг тэнцвэржүүлж дурын идэвхтэй портоос ирж байгаа урсгалыг хүлээн авна. Хэш нь Ethernet эхлэл болон очих хаяг ба хэрэв байгаа бол VLAN хаяг болон IPv4/IPv6 эхлэл болон очих хаягийг багтаадаг.

**LACP**

IEEE® 802.3ad Link Aggregation Control Protocol (LACP) буюу Холбоос Нэгтгэлт Хянах Протокол болон Marker буюу Тэмдэглэгээний Протокол. LACP нь нэгтгэж болох холбоосуудыг нөгөө талтай нь нэг буюу хэд хэдэн Link Aggregated Group буюу Холбоос Нэгтгэгдсэн бүлгүүд (LAG) болгон тохиролцуулдаг. LAG бүр нь адил хурдтай, гүйцэд дуплекс ажиллагаанд тохируулсан портуудаас тогтоно. Урсгал нь LAG дахь портуудын дагуу хамгийн их нийт хурдаар тэнцвэржүүлэгдэх бөгөөд ихэнх тохиолдолд бүх портуудыг агуулсан зөвхөн нэг LAG байх болно. Физик холболт өөрчлөгдөхөд Холбоос Нэгтгэлт шинэ тохиргоо уруу хурдан шилжих болно.

LACP нь хэш хийгдсэн протоколын толгойн мэдээлэл дээр үндэслэн идэвхтэй портуудаар гарч байгаа урсгалыг тэнцвэржүүлж дурын идэвхтэй портоос ирж байгаа урсгалыг хүлээн авна. Хэш нь Ethernet эхлэл болон очих хаяг ба хэрэв байгаа бол VLAN хаяг болон IPv4/IPv6 эхлэл болон очих хаягийг багтаадаг.

**Loadbalance**

Энэ нь *FEC* горимын хуулбар юм.

**Round-robin**

Тойрон эргэх хуваарилалтыг ашиглан гарч байгаа урсгалыг бүх идэвхтэй портуудаар тарааж дурын идэвхтэй портоос ирж байгаа урсгалыг хүлээн авна. Энэ горим нь Ethernet хүрээний дарааллыг зөрчих бөгөөд болгоомжтой ашиглах шаардлагатай байдаг.

### 32.6.3. Жишээнүүд

*Жишээ 40. Cisco® шилжүүлэгчтэй LACP нэгтгэлт*

Энэ жишээ нь FreeBSD машины хоёр интерфэйсийг шилжүүлэгч рүү ачаалал тэнцвэржүүлсэн, алдаанд тэсвэртэй ганц холбоосоор холбож байна. Дамжуулах чанар болон алдаанд тэсвэртэй байдлыг сайжруулах зорилгоор илүү олон интерфэйсүүдийг нэмж болно. Ethernet холбоосууд дээр хүрээний дараалал чухал байдаг учраас хоёр станцын хоорондох ямар ч урсгал аль нэг интерфэйсийн хамгийн их хурдаар хязгаарлагдан тэр физик холбоосоор үргэлж урсаж байдаг. Дамжуулах алгоритм нь өөр өөр урсгалуудыг ялгаж, байгаа интерфэйсүүдээр тэдгээрийг тэнцвэржүүлэхийн тулд аль болох их мэдээллийг ашиглахыг оролддог.

Cisco® шилжүүлэгч дээр *FastEthernet0/1* болон *FastEthernet0/2* интерфэйсүүдийг *1* сувгийн бүлэгт нэмнэ.

```
interface FastEthernet0/1
channel-group 1 mode active
channel-protocol lacp
!
interface FastEthernet0/2
 channel-group 1 mode active
 channel-protocol lacp
```

*fxp0* болон *fxp1*-г ашиглан lagg(4) интерфэйсийг үүсгээд *10.0.0.3/24* гэсэн IP хаяг тавьж интерфэйсийг босгоно:

```
# ifconfig fxp0 up
# ifconfig fxp1 up
# ifconfig lagg0 create
# ifconfig lagg0 up laggproto lacp laggport fxp0 laggport fxp1 10.0.0.3/24
```

Доорх тушаалыг ажиллуулж интерфэйсийн төлвийг харна:

```
# ifconfig lagg0
```

*ACTIVE* гэж тэмдэглэгдсэн портууд нь алсын шилжүүлэгчтэй тохиролцсон, идэвхтэй нэгтгэлтийн бүлгийн хэсэг бөгөөд урсгал нь дамжуулагдаж хүлээн авагдах болно. LAG таних тэмдгүүдийг үзэхийн тулд ifconfig(8)-ийн дэлгэрэнгүй гаралтыг ашиглаарай.

```
lagg0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=8<VLAN_MTU>
        ether 00:05:5d:71:8d:b8
    inet 10.0.0.15 netmask 0xffffff00 broadcast 10.0.0.255
        media: Ethernet autoselect
        status: active
```

```
        laggproto lacp
        laggport: fxp1 flags=1c<ACTIVE,COLLECTING,DISTRIBUTING>
        laggport: fxp0 flags=1c<ACTIVE,COLLECTING,DISTRIBUTING>
```

Шилжүүлэгч дээр портын төлвийг харахын тулд `show lacp neighbor` тушаалыг ашиглах хэрэгтэй.

```
switch# show lacp neighbor
Flags:  S - Device is requesting Slow LACPDUs
        F - Device is requesting Fast LACPDUs
        A - Device is in Active mode       P - Device is in Passive mode

Channel group 1 neighbors

Partner's information:

                  LACP port                        Oper    Port     Port
Port      Flags   Priority  Dev ID         Age     Key     Number   State
Fa0/1     SA      32768     0005.5d71.8db8  29s    0x146   0x3      0x3D
Fa0/2     SA      32768     0005.5d71.8db8  29s    0x146   0x4      0x3D
```

Дэлгэрэнгүйг харахыг хүсвэл `show lacp neighbor detail` тушаалыг ашиглана.

*Жишээ 41. Failover горим*

Failover буюу ослыг тойрон гарах горимыг мастер интерфэйс дээр холбоос тасарсан тохиолдолд хоёр дахь интерфэйс уруу шилжихэд ашиглаж болно. *fxp0* нь мастер интерфэйс, *fxp1* нь хоёр дахь интерфэйс байхаар *lagg0* интерфэйсийг үүсгэж тохируулна:

```
# ifconfig lagg0 create
# ifconfig lagg0 up laggproto failover laggport fxp0 laggport fxp1
```

Интерфэйс нь үүнтэй төстэй байх бөгөөд гол ялгаа нь MAC хаяг болон төхөөрөмжийн нэрс байх болно:

```
# ifconfig lagg0
lagg0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=8<VLAN_MTU>
        ether 00:05:5d:71:8d:b8
        media: Ethernet autoselect
        status: active
        laggproto failover
        laggport: fxp1 flags=0<>
        laggport: fxp0 flags=5<MASTER,ACTIVE>
```

Урсгал *fxp0* дээр хүлээн авагдаж дамжих болно. Хэрэв холбоос *fxp0* дээр тасарсан бол *fxp1* нь идэвхтэй холбоос болно. Хэрэв холбоос нь мастер интерфэйс дээр сэргээгдсэн бол энэ нь эргээд идэвхтэй холбоос болно.

Дахин ачаалахад энэ тохиргоог хэвээр байлгахын тулд дараах мөрүүдийг /etc/rc.conf файлд нэмж өгнө:

```
ifconfig_fxp0="up"
ifconfig_fxp1="up"
cloned_interfaces="lagg0"
ifconfig_lagg0="laggproto failover laggport fxp0 laggport fxp1 10.0.0.15/24"
```

*Жишээ 42. Утастай болон утасгүй сүлжээний интерфэйсүүдийн хооронд Failover горим*

Зөөврийн компьютертай хэрэглэгчдийн хувьд утастай сүлжээний холболт байхгүй болсон үед ашиглахаар утасгүй сүлжээний интерфэйсийг хоёр дахь интерфэйс болгох нь зүйтэй байдаг. Өгөгдлийг утасгүй холболтоор дамжуулах боломжийг хангахын хажуугаар lagg(4)-ийн тусламжтайгаар нэг IP хаяг ашиглах, ажиллагаа болон аюулгүй байдлаас болоод утастай холболтыг илүүд үзэх боломжтой юм.

Энэ тохиргоонд утасгүй сүлжээний MAC хаягийг ашиглагдаж байгаа мастер интерфэйс болох утастай сүлжээний интерфэйсээс авсан lagg(4)-ийн хаягтай адилаар тавих хэрэгтэй.

Энэ тохиргоонд утастай сүлжээний интерфэйс *bge0*-ийг мастер, утасгүй сүлжээний интерфэйс *wlan0*-ийг failover интерфэйс гэж авч үзнэ. *wlan0*-ийг *iwn0*-ээс үүсгэсэн бөгөөд бид үүнд утастай сүлжээний холболтын MAC хаяг зааж өгөх болно. Эхний алхам бол MAC хаягийг утастай сүлжээний интерфэйсээс авах явдал юм:

```
# ifconfig bge0
bge0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
    options=19b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM,TSO4>
    ether 00:21:70:da:ae:37
    inet6 fe80::221:70ff:feda:ae37%bge0 prefixlen 64 scopeid 0x2
    nd6 options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL>
    media: Ethernet autoselect (1000baseT <full-duplex>)
    status: active
```

Та *bge0*-ийг өөрт байгаагаар солих хэрэгтэй бөгөөд ингэхэд таны утастай сүлжээний интерфэйсийн MAC хаяг бүхий өөр `ether` хаягийг авах болно. Одоо утасгүй сүлжээний интерфэйс *iwn0*-ийг солино:

```
# ifconfig iwn0 ether 00:21:70:da:ae:37
```

Утасгүй сүлжээний интерфэйсийг босгоно, гэхдээ ямар ч IP хаяг тавих хэрэггүй:

```
# ifconfig wlan0 create wlandev iwn0 ssid my_router up
```

*bge0* интерфэйсийг босгоно. *bge0*-ийг мастер маягаар ашиглан хэрэв шаардлагатай бол *wlan0* руу failover хийх боломжтойгоор lagg(4) интерфэйс үүсгэнэ:

```
# ifconfig bge0 up
# ifconfig lagg0 create
# ifconfig lagg0 up laggproto failover laggport bge0 laggport wlan0
```

Интерфэйс нь үүнтэй төстэй байх бөгөөд гол ялгаа нь MAC хаяг ба төхөөрөмжийн нэрс байх болно:

```
# ifconfig lagg0
lagg0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=8<VLAN_MTU>
        ether 00:21:70:da:ae:37
        media: Ethernet autoselect
        status: active
        laggproto failover
        laggport: wlan0 flags=0<>
        laggport: bge0 flags=5<MASTER,ACTIVE>
```

Дараа нь IP хаяг авахын тулд DHCP клиентийг эхлүүлнэ:

```
# dhclient lagg0
```

Дахин ачаалахад энэ тохиргоог хэвээр байлгахын тулд дараах мөрүүдийг /etc/rc.conf файлд нэмж өгнө:

```
ifconfig_bge0="up"
ifconfig_iwn0="ether 00:21:70:da:ae:37"
wlans_iwn0="wlan0"
ifconfig_wlan0="WPA"
cloned_interfaces="lagg0"
ifconfig_lagg0="laggproto failover laggport bge0 laggport wlan0 DHCP"
```

## 32.7. Дискгүй ажиллагаа

FreeBSD машин сүлжээгээр ачаалан локал дискгүйгээр NFS сервер дээр холбогдсон файлын системүүд ашиглаад ажиллаж чаддаг. Стандарт тохиргооны файлуудаас өөр системийн өөрчлөлт шаардлагагүй. Ийм системийг харьцангуй хялбараар тохируулж болдог. Учир нь шаардлагатай бүх элементүүд бэлэн байдаг:

- Цөмийг сүлжээгээр ачаалах хоёр боломжит арга хамгийн багаар бодоход байдаг:
  - PXE: Intel®-ийн Preboot eXecution Environment буюу ачаалалт хийгдэхээс өмнөх ажиллуулах орчны систем нь зарим сүлжээний картууд эсвэл эх хавтангуудад цуг бүтээгдсэн байдаг ухаалаг ачаалах ROM-ийн нэг хэлбэр юм. Илүү дэлгэрэнгүйг pxeboot(8)-с үзнэ үү.
  - Etherboot порт (net/etherboot) нь цөмийг сүлжээгээр ачаалах ROM хийгдэх боломжтой код үүсгэдэг. Код нь сүлжээний картан дээр ачаалах ROM уруу шарагдсан байх юм уу эсвэл локал уян (эсвэл хатуу) дискний хөтчөөс эсвэл ажиллаж байгаа MS-DOS® системээс дуудагдахаар байдаг. Олон сүлжээний карт дэмжигдсэн байгаа.
- Жишээ скрипт (/usr/shared/examples/diskless/clone_root) нь сервер дээр ажлын станцын root файлын системийг үүсгэх болон түүний ажиллагааг хангах үйлдлийг хөнгөвчилдөг. Скрипт нь магадгүй жижиг өөрчлөлтийг шаардаж болох боловч энэ нь таныг хурдан эхлүүлэх болно.
- Дискгүй системийн эхлүүлэлтийг илрүүлж дэмжих стандарт системийн эхлүүлэгч файлууд /etc санд байдаг.
- swap хийх хэрэв шаардлагатай бол NFS файл эсвэл локал диск уруу хийж болох юм.

Дискгүй ажлын станцуудыг тохируулах олон арга байдаг. Олон элементүүд үүнд хамрагддаг бөгөөд өөрийн сонирхлоор ихэнхийг нь өөрчилж болно. Дараах нь бүрэн системийг тохируулах талаар хувилбаруудыг тайлбарлах болно. Ингэхдээ стандарт FreeBSD эхлүүлэх скриптүүдтэй хялбар, нийцтэй байхыг чухалчлах болно. Тайлбарласан систем нь дараах шинжүүдтэй байна:

- Дискгүй ажлын станцууд нь хуваалцсан, зөвхөн уншигдах / файлын систем болон хуваалцсан, зөвхөн уншигдах /usr-г ашигладаг.

  root файлын систем нь зөвхөн дискгүй ажиллагаатай холбоотой эсвэл тэдгээрийн харьяалагдах ажлын станцтай холбоотой зарим тохиргооны файлууд нь өөрчлөгдсөн стандарт FreeBSD root-ийн (ихэвчлэн серверийн) хуулбар юм.

  Бичигдэх боломжтой байх root-ийн хэсэг нь md(4) файлын системээр дээр тавигддаг (overlaid). Систем дахин ачаалахад хийгдсэн өөрчлөлтүүд алга болох болно.

- Цөм дамжуулагдаж Etherboot эсвэл PXE-ийн аль нэгээр дуудагддаг бөгөөд зарим тохиолдолд зөвхөн аль нэг аргыг ашиглахыг шаардаж болох юм.

Тайлбарласны дагуу энэ систем нь аюултай юм. Энэ нь сүлжээний хамгаалагдсан талбарт байх ёстой бөгөөд бусад хостуудаас хандахааргүй байх ёстой.

Энэ хэсгийн бүх мэдээллийг FreeBSD 5.2.1-RELEASE ашиглан тест хийсэн болно.

### 32.7.1. Үндсэн мэдээлэл

Дискгүй ажлын станцуудыг тохируулах нь харьцангуй амархан боловч алдаанд өртөх хандлагатай байдаг. Хэд хэдэн шалтгаанаас болоод эдгээрийг заримдаа оношлох төвөгтэй байдаг. Жишээ нь:

- Эмхэтгэлтийн үеийн тохируулгууд нь ажиллах үед өөрөөр ажиллахыг тодорхойлж болох юм.
- Алдааны мэдэгдлүүд нь ихэвчлэн нуугдмал эсвэл бүр байхгүй ч байж болно.

Энд гарч болзошгүй асуудлуудыг шийдэхэд үндсэн арга замуудын зарим нэг мэдлэгийг ашиглах нь маш ашигтай байдаг.

Амжилттайгаар эхлүүлэхийн тулд хэд хэдэн үйлдлүүдийг хийх хэрэгтэй:

- Машин өөрийн IP хаяг, ажиллах файлын нэр, серверийн нэр, root зам зэрэг эхний параметрүүдийг авах хэрэгтэй. Үүнийг DHCP эсвэл BOOTP протоколуудыг ашиглан хийдэг. DHCP нь BOOTP-ийн нийцтэй өргөтгөл бөгөөд адил портын дугаарууд болон үндсэн пакетийн хэлбэршүүлэлтийг ашигладаг.

  Зөвхөн BOOTP ашиглахаар системийг тохируулах боломжтой. bootpd(8) серверийн програм нь үндсэн FreeBSD системд орсон байдаг.

  Гэхдээ DHCP нь BOOTP-ээс хэд хэдэн давуу талтай байдаг (илүү сайхан тохиргооны файлууд, PXE ашиглах боломж, дискгүй ажиллагаатай шууд холбоогүй бусад олон давуу талууд) бөгөөд бид голчлон DHCP тохиргоог боломжтой бол bootpd(8) ашиглан таарах жишээнүүдтэй нь хамт тайлбарлах болно. Жишээ тохиргоо нь ISC DHCP програм хангамжийн багцыг (тест сервер дээр 3.0.1.r12 хувилбар суулгагдсан) ашиглах болно.

- Машин нь нэг буюу хэд хэдэн програмуудыг локал санах ой уруугаа дамжуулах хэрэгтэй болно. TFTP юм уу эсвэл NFS ашиглагдана. TFTP болон NFS-ийн сонголтыг хэд хэдэн газар эмхэтгэх үеийн тохируулгаар хийдэг. Алдааны түгээмэл эх үүсвэр нь буруу протоколд зориулж файлын нэрүүдийг заах явдал юм: TFTP нь сервер дээрх ганц сангаас бүх файлуудыг ихэвчлэн дамжуулдаг бөгөөд энэ сантай харьцангуй файлын нэрсийг хүлээдэг. NFS-д файлын туйлын (абсолют) замууд хэрэгтэй байдаг.
- Боломжит дундын эхлүүлэх програмууд болон цөм нь эхлэн тохируулагдаж ажиллах шаардлагатай. Энд хэд хэдэн чухал хувилбарууд байдаг:
  - PXE нь FreeBSD-ийн гурав дахь шатны дуудагчийн өөрчлөгдсөн хувилбар болох pxeboot(8)-ийг ачаална. pxeboot(8) нь системийг эхлүүлэхэд шаардлагатай ихэнх параметрүүдийг авч хяналтыг дамжуулахаасаа өмнө тэдгээрийг цөмийн орчинд үлдээх болно. Энэ тохиолдолд GENERIC цөмийг ашиглах боломжтой.
  - Etherboot нь бага бэлтгэлтэйгээр цөмийг шууд ачаалах болно. Та цөмийг тусгай тохируулгуудтай бүтээх хэрэгтэй болно.

    PXE болон Etherboot нь адилхан ажиллана; гэхдээ цөмүүд нь loader(8)-ийн тэдэнд зориулж илүү ажил хийх боломжийг олгодог болохоор PXE нь зохимжтой арга юм.

    Хэрэв таны BIOS болон сүлжээний картууд нь PXE-г дэмждэг бол та үүнийг ашиглах хэрэгтэй байж болох юм.

- Эцэст нь машин өөрийн файлын системүүдэд хандах хэрэгтэй болно. NFS нь бүх тохиолдолд хэрэглэгдэнэ.

diskless(8) гарын авлагын хуудсыг бас үзнэ үү.

## 32.7.2. Тохируулах зааврууд

### 32.7.2.1. ISC DHCP ашиглах тохиргоо

ISC DHCP сервер BOOTP болон DHCP хүсэлтүүдэд хариулж чаддаг.

ISC DHCP 4.2 нь үндсэн системийн хэсэг биш юм. Та эхлээд net/isc-dhcp42-server порт юм уу эсвэл тохирох багцыг суулгах хэрэгтэй.

ISC DHCP суулгагдсаны дараа ажиллахын тулд тохиргооны файл хэрэгтэй болно (ихэвчлэн /usr/local/etc/dhcpd.conf гэж нэрлэгддэг). Доор `margaux` хост Etherboot-ийг ашигласан ба `corbieres` хост PXE-г ашиглаж байгаа тайлбар хийгдсэн жишээ байна:

```
default-lease-time 600;
max-lease-time 7200;
authoritative;

option domain-name "example.com";
option domain-name-servers 192.168.4.1;
option routers 192.168.4.1;

subnet 192.168.4.0 netmask 255.255.255.0 {
  use-host-decl-names on; ①
  option subnet-mask 255.255.255.0;
  option broadcast-address 192.168.4.255;

  host margaux {
    hardware ethernet 01:23:45:67:89:ab;
    fixed-address margaux.example.com;
    next-server 192.168.4.4; ②
    filename "/data/misc/kernel.diskless"; ③
    option root-path "192.168.4.4:/data/misc/diskless"; ④
  }
  host corbieres {
    hardware ethernet 00:02:b3:27:62:df;
    fixed-address corbieres.example.com;
    next-server 192.168.4.4;
    filename "pxeboot";
    option root-path "192.168.4.4:/data/misc/diskless";
  }
}
```

① Энэ тохируулга нь `host` зарлалтууд дахь утгыг дискгүй хостын хувьд хостын нэр болгон илгээхийг dhcpd-д хэлнэ. Өөр нэг арга нь `option host-name margaux` мөрийг `host` зарлалтуудын дотор нэмэх явдал юм.

② `next-server` тохируулга нь дуудагч эсвэл цөмийн файлыг дуудахад зориулж TFTP эсвэл NFS серверийг ашиглахыг зааж өгнө (анхдагчаар DHCP сервер байгаа хостыг ашиглана).

③ `filename` тохируулга нь Etherboot эсвэл PXE-ийн дараагийн ажиллуулах алхамд дуудах файлыг тодорхойлно. Үүнийг ашиглах дамжуулалтын аргын дагуу заасан байх ёстой.

Etherboot-ийг NFS эсвэл TFTP ашиглахаар эмхэтгэж болно. FreeBSD порт нь NFS-ийг анхдагчаар тохируулдаг. PXE нь TFTP-г ашигладаг. Энд яагаад харьцангуй файлын нэр хэрэглэгддэгийн учир нь тэр юм (энэ нь TFTP серверийн тохиргооноос хамаарах боловч ерөнхийдөө нэлээн энгийн байх болно). PXE нь цөмийг биш бас pxeboot-г дууддаг. FreeBSD CD-ROM /boot сангаас pxeboot-ийг дуудах зэрэг бусад сонирхолтой боломжууд байдаг (pxeboot(8) нь GENERIC цөмийг ачаалж чаддаг бөгөөд энэ нь алсын CD-ROM-с ачаалахын тулд PXE-г ашиглах боломжтой болгодог юм).

④ `root-path` тохируулга нь root файлын систем хүрэх замыг энгийн NFS бичлэгээр тодорхойлдог. PXE ашиглаж байгаа үед цөмийн тохиргоо BOOTP-г идэвхжүүлээгүй байхад хостын IP-г орхигдуулах боломжтой байдаг. NFS сервер дараа нь TFTP сервертэй адил болох болно.

#### 32.7.2.2. BOOTP ашиглах тохиргоо

Энд bootpd-ийн адил тохиргоо (нэг клиент болгож багасгасан) байна. Үүнийг /etc/bootptab файлаас олж болно.

BOOTP-г ашиглахын тулд Etherboot-ийг анхдагч биш тохируулга `NO_DHCP_SUPPORT`-тойгоор эмхэтгэх ёстой бөгөөд PXE-д DHCP *хэрэгтэй* гэдгийг санаарай. bootpd-ийн цорын ганц илэрхий давуу тал бол үндсэн системд орсон байдаг явдал юм.

```
.def100:\
  :hn:ht=1:sa=192.168.4.4:vm=rfc1048:\
  :sm=255.255.255.0:\
  :ds=192.168.4.1:\
  :gw=192.168.4.1:\
  :hd="/tftpboot":\
  :bf="/kernel.diskless":\
  :rp="192.168.4.4:/data/misc/diskless":

margaux:ha=0123456789ab:tc=.def100
```

#### 32.7.2.3. Ачаалах програмыг Etherboot-ээр бэлдэх

Etherboot-ийн вэб сайт нь Линукс системд голчлон зориулагдсан боловч ашигтай мэдээлэл агуулсан өргөттөсөн баримтыг агуулдаг. Дараах нь FreeBSD систем дээр Etherboot-г хэрхэн ашиглах талаар тайлбарлах болно.

Та эхлээд net/etherboot багц юм уу эсвэл портыг суулгах ёстой.

Etherboot-ийн тохиргоог Etherboot-ийн эх байгаа сан дахь Config файлыг засварлаж (өөрөөр хэлбэл NFS-ийн оронд TFTP-г ашиглахын тулд) өөрчилж болно.

Бидний тохиргооны хувьд бид ачаалах уян дискийг ашиглах болно. Бусад аргуудын хувьд (PROM, эсвэл MS-DOS® програм) Etherboot-ийн баримтаас лавлана уу.

Ачаалах уян дискийг хийхийн тулд Etherboot суулгасан машин дээрх хөтөч уруугаа уян дискээ хийгээд Etherboot мод дахь src сан уруу болгож одоогийн сангаа солиод доор дурдсан тушаалыг бичнэ:

```
# gmake bin32/devicetype.fd0
```

*devicetype* нь дискгүй ажлын станц дахь Ethernet картын төрлөөс хамаарна. Зөв *devicetype* буюу төхөөрөмжийн төрлийг тодорхойлохын тулд тэр сан дахь NIC файлд хандана.

#### 32.7.2.4. PXE-ээр ачаалах

Анхдагчаар pxeboot(8) дуудагч нь NFS-ээр цөмийг дууддаг. /etc/make.conf файлд `LOADER_TFTP_SUPPORT` тохируулга заахын оронд TFTP-г ашиглахаар эмхэтгэгдэж болно. Зааврууд талаар /usr/shared/examples/etc/make.conf файл дахь тайлбаруудыг үзнэ үү.

Цуваа консол дискгүй машин тохируулахад ашигтай байж болох make.conf файлын өөр хоёр тохируулга байдаг: эдгээр нь `BOOT_PXELDR_PROBE_KEYBOARD`, болон `BOOT_PXELDR_ALWAYS_SERIAL` юм.

Машин эхлэхэд PXE-г ашиглахын тулд та өөрийн BIOS-ийн тохируулгаас `Boot from network` буюу `Сүлжээнээс ачаалах` сонголтыг сонгох юм уу эсвэл PC-г эхлүүлж байх үед функцийн товчлуурыг дарах хэрэгтэй.

#### 32.7.2.5. TFTP болон NFS серверүүдийг тохируулах

Хэрэв та TFTP ашиглахаар тохируулсан PXE юм уу эсвэл Etherboot-г ашиглаж байгаа бол файл сервер дээрээ tftpd-г идэвхжүүлэх хэрэгтэй:

1. tftpd-д зориулж файлууд байх сан үүсгэнэ, өөрөөр хэлбэл /tftpboot.
2. Энэ мөрийг өөрийн /etc/inetd.conf файлд нэмнэ:

   ```
   tftp    dgram   udp wait    root    /usr/libexec/tftpd  tftpd -l -s /tftpboot
   ```

   Зарим нэгэн PXE хувилбарууд нь TFTP-ийн TCP хувилбарыг хүсдэг байна. Энэ тохиолдолд `dgram udp` мөрийг `stream tcp` болгон сольж хоёр дахь мөрийг нэмнэ.

3. inetd-д өөрийн тохиргооны файлаа дахин уншихыг хэлээрэй. Энэ тушаалыг зөв ажиллуулахын тулд /etc/rc.conf файлд `inetd_enable="YES"` тохируулга заагдсан байх ёстой:

   ```
   # service inetd restart
   ```

Та tftpboot санг серверийнхээ хаана ч байрлуулж болно. inetd.conf болон dhcpd.conf файлууд дээрээ байрлалаа заасан эсэхээ шалгаарай.

Аль ч тохиолдолд та бас NFS-ээ идэвхжүүлж NFS сервер дээрээ тохирох файлын системээ экспорт хийх хэрэгтэй юм.

1. Үүнийг /etc/rc.conf-д нэмнэ:

```
nfs_server_enable="YES"
```

2. Доор дурдсаныг /etc/exports файл уруу нэмж дискгүй root сан байгаа файлын системээ экспорт хийнэ (эзлэхүүний холбох цэгийг тааруулж *margaux corbieres* гэснийг дискгүй ажлын станцуудын нэрсээр солих хэрэгтэй):

```
/data/misc -alldirs -ro margaux corbieres
```

3. mountd-д өөрийн тохиргооны файлаа дахин уншихыг хэлнэ. Хэрэв та эхний шатандаа NFS-г /etc/rc.conf файлд идэвхжүүлэх хэрэгтэй болсон бол харин дахин ачаалахыг хүсэж болох юм.

```
# service mountd restart
```

#### 32.7.2.6. Дискгүй цөмийг бүтээх

Хэрэв Etherboot-г ашиглаж байгаа бол дискгүй клиентдээ зориулж дараах тохируулгуудтайгаар (ердийн тохируулгуудын хамтаар) цөмийн тохиргооны файл үүсгэх хэрэгтэй:

```
options     BOOTP          # Use BOOTP to obtain IP address/hostname
options     BOOTP_NFSROOT  # NFS mount root file system using BOOTP info
```

Та бас BOOTP_NFSV3, BOOT_COMPAT болон BOOTP_WIRED_TO (NOTES файлаас лавлана уу) тохируулгуудыг ашиглахыг хүсэж болох юм.

Эдгээр тохируулгуудын нэрс нь уламжлалт бөгөөд тэдгээр нь яг үнэндээ DHCP болон BOOTP-ийн өөр хэрэглээг цөм дотор идэвхжүүлдэг болохоор нэлээн төөрөгдөлд хүргэж болох юм (зөвхөн BOOTP эсвэл DHCP-ийн хэрэглээг ашиглахаар болгох бас боломжтой байдаг).

Цөмийг бүтээж (FreeBSD цөмийг тохируулах нь-г үзнэ үү), dhcpd.conf файлд заасан байрлал уруу хуулах хэрэгтэй.

PXE-г ашиглаж байгаа үед дээрх тохируулгуудтайгаар цөмийг бүтээх нь заавал шаардлагагүй (хэдийгээр зөвлөдөг боловч). Тэдгээрийг идэвхжүүлэх нь цөмийг эхлүүлэх явцад илүү олон DHCP хүсэлтүүдийг үүсгэж шинэ утгууд болон зарим тусгай тохиолдлуудад pxeboot(8)-ийн авсан утгуудын хооронд нийцэмжгүй байдлын жижиг эрсдэл гаргахад хүргэдэг. Тэдгээрийг ашиглахын давуу тал нь буруу нөлөөллөөс болоод хостын нэр тохируулагддаг явдал юм. Үгүй бол та хостын нэрийг өөр аргаар, жишээ нь клиентэд тусгайлан зориулсан rc.conf файлд тохируулах хэрэгтэй болно.

Etherboot-ээр дуудагдах боломжтой байхын тулд цөмд төхөөрөмжийн бяцхан зааврууд эмхэтгэгдсэн байх хэрэгтэй байдаг. Ингэхдээ та дараах тохируулгыг тохиргооны файлдаа тохируулах болно (NOTES тохиргооны тайлбаруудын файлыг үзнэ үү):

```
hints       "GENERIC.hints"
```

#### 32.7.2.7. Root файлын системийг бэлдэх

Дискгүй ажлын станцад зориулж dhcpd.conf файлд `root-path` гэж жагсаагдсан байрлалд root файлын системийг та үүсгэх хэрэгтэй.

##### 32.7.2.7.1. root-ийг суурьшуулахын тулд `make world` тушаал ашиглах

Энэ арга нь хурдан байх бөгөөд бүрэн онгон системийг (зөвхөн root файлын систем биш) `DESTDIR` уруу суулгах болно. Та дараах скриптийг л ердөө ажиллуулах хэрэгтэй болно:

```
#!/bin/sh
export DESTDIR=/data/misc/diskless
mkdir -p ${DESTDIR}
cd /usr/src; make buildworld && make buildkernel
make installworld && make installkernel
cd /usr/src/etc; make distribution
```

Үүнийг хийсний дараа танд өөрийнхөө хэрэгцээний дагуу `DESTDIR`-д байрлуулагдсан өөрийн /etc/rc.conf болон /etc/fstab файлуудаа өөрчлөн тохируулах шаардлага гарч болох юм.

#### 32.7.2.8. Swap-ийг тохируулах

Шаардлагатай бол сервер дээр байрласан swap файлд NFS-ээр хандаж болно.

##### 32.7.2.8.1. NFS Swap

Цөм нь ачаалах үед NFS swap-ийг идэвхжүүлэхийг дэмждэггүй. Бичигдэх файлын системийг холбож swap файлыг үүсгэн идэвхжүүлэх замаар эхлүүлэх скриптүүдээр swap нь идэвхжүүлэгдэх ёстой. Зохих хэмжээ бүхий swap файлыг үүсгэхдээ та ингэж хийж болно:

```
# dd if=/dev/zero of=/path/to/swapfile bs=1k count=1 oseek=100000
```

Үүнийг идэвхжүүлэхийн тулд та дараах мөрийг өөрийн rc.conf файлд нэмэх хэрэгтэй:

```
swapfile=/path/to/swapfile
```

#### 32.7.2.9. Бусад асуудлууд

##### 32.7.2.9.1. Зөвхөн уншигдах /usr сантайгаар ажиллах

Хэрэв дискгүй ажлын станц нь X-ийг ажиллуулахаар тохируулагдсан бол анхдагчаар /usr-т алдааны бүртгэлийг хийдэг XDM тохиргооны файлыг та тааруулах хэрэгтэй болно.

##### 32.7.2.9.2. FreeBSD биш сервер ашиглах

root файлын системд зориулсан сервер нь FreeBSD-г ажиллуулаагүй бол та FreeBSD машин дээр root файлын систем үүсгээд хэрэгтэй байрлал уруу нь `tar` эсвэл `cpio` тушаал ашиглан хуулах хэрэгтэй болно.

Энэ тохиолдолд бүхэл тооны их/бага (major/minor) хэмжээнүүдийн ялгаанаас болоод /dev дахь тусгай файлуудад заримдаа асуудал гардаг. Энэ асуудлын шийдэл нь FreeBSD биш серверээс уг санг экспорт хийж энэ сангаа FreeBSD машин дээр холбон хэрэглэгчийн хувьд төхөөрөмжийн цэгүүдийг харагдахгүйгээр (transparently) хуваарилахын тулд devfs(5)-ийг ашиглах явдал юм.

# 32.8. NFS root файлын систем ашиглан PXE ачаалалт хийх

Intel®-н Preboot eXecution Environment (PXE) буюу Ачаалалтаас өмнө ажиллуулах орчин нь үйлдлийн системийг сүлжээгээр ачаалах боломжийг олгодог. PXE дэмжлэг орчин үеийн эх хавтангуудын BIOS дээр ихэвчлэн хийгдсэн байдаг бөгөөд BIOS тохиргоонуудад үүнийг идэвхжүүлэн сүлжээгээр ачаалах боломжтой болгодог. Бүрэн хэмжээгээр ажиллах PXE тохиргоотой байхын тулд зөв тохируулсан DHCP болон TFTP сервертэй байхыг шаарддаг.

Хост компьютер ачаалахдаа DHCP-с TFTP ашиглан эхний ачаалагч дуудагчийг хаанаас авч болох мэдээллийг авдаг. Энэ мэдээллийг хост компьютер хүлээж авсныхаа дараа ачаалагч дуудагчийг TFTP-р татаж аван ажиллуулдаг. Үүнийг Preboot Execution Environment (PXE)-ийн тодорхойлолт хаягийн 2.2.1 хэсэг бичсэн байгаа. FreeBSD дээр PXE процессын явцад хүлээн авсан ачаалагч дуудагч нь /boot/pxeboot байна. /boot/pxeboot ажилласны дараа FreeBSD-н цөм дуудагддаг бөгөөд FreeBSD-н бусад ачаалалт хийгддэг. FreeBSD-н ачаалах процессын талаар дэлгэрэнгүйг FreeBSD-ийн Ачаалах процесс хэсгээс үзнэ үү.

## 32.8.1. NFS root файлын системд зориулж `chroot` орчин тохируулах

1. FreeBSD суулгац байх NFS-р холбож болох санг сонгоно. Жишээ нь /b/tftpboot/FreeBSD/install гэх мэт санг ашиглаж болно.

   ```
   # export NFSROOTDIR=/b/tftpboot/FreeBSD/install
   # mkdir -p ${NFSROOTDIR}
   ```

2. NFS-н тохиргоог хийх хэсэг дэх зааврыг даган NFS серверийг идэвхжүүлнэ.
3. Дараахийг /etc/exports файлд нэмэн NFS-р санг экспорт хийнэ:

```
/b -ro -alldirs
```

4. NFS серверийг дахин асаана:

```
# service nfsd restart
```

5. Тохиргоо хэсэгт заасан алхмуудын дагуу inetd(8)-г идэвхжүүлнэ.
6. /etc/inetd.conf-д дараах мөрийг нэмнэ:

```
tftp dgram udp wait root /usr/libexec/tftpd tftpd -l -s /b/tftpboot
```

7. inetd-г дахин асаана:

```
# service inetd restart
```

8. FreeBSD-н цөм болон хэрэглэгчийн талбарыг дахин бүтээнэ:

```
# cd /usr/src
# make buildworld
# make buildkernel
```

9. NFS-р холбогдсон санд FreeBSD-г суулгана:

```
# make installworld DESTDIR=${NFSROOTDIR}
# make installkernel DESTDIR=${NFSROOTDIR}
# make distribution DESTDIR=${NFSROOTDIR}
```

10. TFTP ажиллаж байгааг болон PXE-р дамжин ачаалагч дуудагчийг татаж авч байгаа эсэхийг шалгана:

```
# tftp localhost
tftp> get FreeBSD/install/boot/pxeboot
Received 264951 bytes in 0.1 seconds
```

11. ${NFSROOTDIR}/etc/fstab-г засварлан root файлын системийг NFS-р холбох мөрийг үүсгэнэ:

```
# Device                                    Mountpoint    FSType   Options
Dump Pass
myhost.example.com:/b/tftpboot/FreeBSD/install      /         nfs      ro
0    0
```

*myhost.example.com* гэдгийг таны өөрийн NFS серверийн IP хаяг эсвэл хостын нэрээр солих хэрэгтэй. Энэ жишээн дээр NFS хэрэглэгчид root файлын системийн агуулгыг устгачих магадлалаас хамгаалж root файлын системийг "зөвхөн унших" эрхээр холбосон байгаа.

12. chroot(8) орчинд root нууц үгийг тохируулна.

```
# chroot ${NFSROOTDIR}
# passwd
```

Энэ нь PXE-р ачаалж байгаа хэрэглэгч машинуудад зориулж root нууц үгийг тохируулах болно.

13. ${NFSROOTDIR}/etc/ssh/sshd_config файлыг засварлан `PermitRootLogin` гэдгийг идэвхжүүлж PXE-р ачаалж байгаа хэрэглэгч машинуудад зориулж ssh-р дамжин root хэрэглэгч нэвтрэх боломжийг бүрдүүлнэ. Үүнийг sshd_config(5)-д тайлбарласан байгаа.
14. ${NFSROOTDIR} дахь chroot(8) орчинд бусад өөрчлөлтүүдийг хийнэ. Эдгээр өөрчлөлтүүдэд pkg_add(1) ашиглан багцууд нэмэх, vipw(8), ашиглан нууц үгийн файлыг засварлах, эсвэл автоматаар холбохын тулд amd.conf(8) харгалзааг засварлах зэрэг орж болно. Жишээ нь:

```
# chroot ${NFSROOTDIR}
# pkg_add -r bash
```

### 32.8.2. /etc/rc.initdiskless ашигладаг Санах Ойн Файлын системийг тохируулах

Хэрэв та NFS root-с ачаалсан бол /etc/rc нь NFS-ээр ачаалсныг мэдэж /etc/rc.initdiskless скриптийг ажиллуулдаг. Юу болж байгааг энэ скрипт доторх тайлбаруудыг уншиж мэдээрэй. Бид /etc болон /var гэсэн санах ой дээр тулгуурласан файлын систем үүсгэх хэрэгтэй, учир нь NFS root сан нь зөвхөн уншигдах эрхтэй бөгөөд эдгээр сангууд нь бичигдэх боломжтой байх ёстой.

```
# chroot ${NFSROOTDIR}
# mkdir -p conf/base
# tar -c -v -f conf/base/etc.cpio.gz --format cpio --gzip etc
# tar -c -v -f conf/base/var.cpio.gz --format cpio --gzip var
```

Систем ачаалахад /etc болон /var-д зориулсан санах ойн файлын системүүд үүсч холбогдох бөгөөд cpio.gz файлуудын агуулга тэдгээр рүү хуулагдах болно.

### 32.8.3. DHCP сервер тохируулах

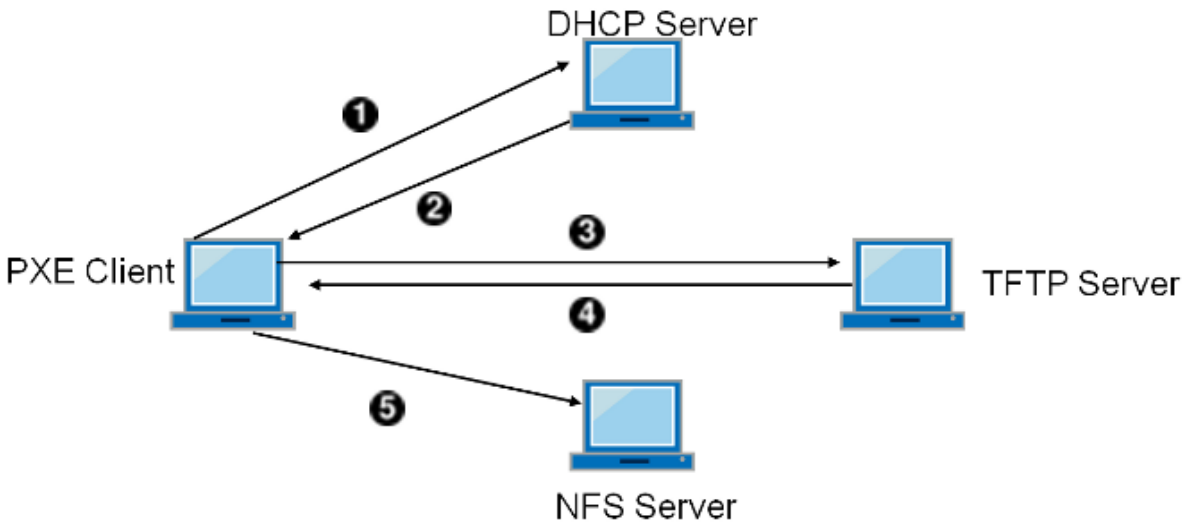
PXE нь TFTP болон DHCP сервер тохируулсан байхыг шаарддаг. DHCP сервер нь TFTP сервер байгаа машин дээр заавал байх шаардлагагүй бөгөөд таны сүлжээнээс хандаж болох боломжтой байх ёстой.

1. DHCP Серверийг Суулгах болон Тохируулах заасан зааврын дагуу DHCP серверийг суулгана. /etc/rc.conf болон /usr/local/etc/dhcpd.conf файлуудыг зөв тохируулсан эсэхийг шалгаарай.
2. /usr/local/etc/dhcpd.conf файлын `next-server`, `filename`, болон `option root-path` тохиргоонуудад TFTP-н IP хаяг, TFTP дэх /boot/pxeboot хүрэх зам болон NFS root файлын системийн замыг зааж өгнө. Жишээ dhcpd.conf файл ийм байна:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
   range 192.168.0.2 192.168.0.3 ;
   option subnet-mask 255.255.255.0 ;
   option routers 192.168.0.1 ;
   option broadcast-address 192.168.0.255 ;
   option domain-name-server 192.168.35.35, 192.168.35.36 ;
   option domain-name "example.com";

   # IP address of TFTP server
   next-server 192.168.0.1 ;

   # path of boot loader obtained
   # via tftp
   filename "FreeBSD/install/boot/pxeboot" ;

   # pxeboot boot loader will try to NFS mount this directory for root FS
   option root-path "192.168.0.1:/b/tftpboot/FreeBSD/install/" ;

}
```

### 32.8.4. PXE хэрэглэгчийг тохируулах ба холболтын үеийн асуудлуудад дибаг хийх

1. Хэрэглэгч машин ачаалахдаа BIOS тохиргооны цэс рүү орно. Сүлжээнээс ачаалахыг BIOS-д тохируулна. Хэрэв өмнөх бүх алхмууд зөв бол бүгд "зүгээр л ажиллах ёстой".
2. PXE ачаалах процессын явцад үүсэх сүлжээний урсгалыг дибаг хийхийн тулд net/wireshark-г ашиглаарай. Үүнийг доорх диаграм дээр дүрсэлсэн байгаа. DHCP сервер тохируулах хэсэгт жишээ тохиргоог харуулсан бөгөөд DHCP, TFTP ба NFS серверүүд нь нэг машин дээр байна. Гэхдээ эдгээр серверүүд нь өөр өөр машин дээр байж болно.




*Зураг 109. NFS root холболт ашиглан PXE ачаалах процесс*

3. pxeboot-г TFTP-р авч болж байгаа эсэхийг шалгаарай. TFTP сервер дээрээ /var/log/xferlog файлыг нээж харан pxeboot файл зөв газраас татагдаж байгаа эсэхийг шалгаарай. Дээрх dhcpd.conf файлын тохиргоог шалгахын тулд:

```
# tftp 192.168.0.1
tftp> get FreeBSD/install/boot/pxeboot
Received 264951 bytes in 0.1 seconds
```

tftpd(8) ба tftp(1) гарын авлагын хуудаснуудыг уншаарай. TFTP-н зарим нэг хязгаарлалтуудыг BUGS хэсэгт тайлбарласан байгаа.

4. root файлын системийг NFS-р холбож болж байгаа эсэхийг шалгаарай. Дээрх dhcpd.conf файлын тохиргоог шалгахын тулд:

```
# mount -t nfs 192.168.0.1:/b/tftpboot/FreeBSD/install /mnt
```

5. boot.nfsroot.server болон boot.nfsroot.path гэх мэт хувьсагчуудыг pxeboot дуудагч хэрхэн тохируулдагийг src/sys/boot/i386/libi386/pxe.c файлын кодыг уншиж ойлгоорой. Эдгээр хувьсагчууд дараагаар нь src/sys/nfsclient/nfs_diskless.c файлын NFS-р дискгүй root холбох кодод ашиглагддаг.
6. pxeboot(8) болон loader(8) гарын авлагын хуудаснуудыг уншина уу.

## 32.9. ISDN

ISDN технологи болон тоног төхөөрөмжийн тухай мэдээллийн сайн эх үүсвэр нь Дан Кегелийн ISDN хуудас юм.

ISDN-ий тухай хялбар зааврууд доор дурдав:

- Хэрэв та Европд амьдарч байгаа бол ISDN картын хэсгийг та магадгүй шалгахыг хүсэж болох юм.

- Хэрэв та Интернэт уруу Интернэтийн үйлчилгээ үзүүлэгчтэй dial-up-аар байнгын бишээр холбогдохдоо ISDN-ийг голчлон хэрэглэхээр төлөвлөж байгаа бол та Терминал Хувиргагчдыг үзэж болох юм. Хэрэв та үйлчилгээ үзүүлэгчдээ солих бол энэ нь асуудлууд хамгийн багатайгаар хамгийн уян хатан чанарыг танд өгөх болно.
- Хэрэв та хоёр LAN-ийг хооронд нь холбож байгаа бол эсвэл Интернэт уруу зориулагдсан ISDN холболтоор холбогдож байгаа бол зориулагдсан чиглүүлэгч/гүүр сонголтыг бодолцож болох юм.

Аль шийдлийг сонгохыг тодорхойлоход чухал хүчин зүйл нь өртөг юм. Дараах сонголтуудыг хамгийн бага өртөгтэйгөөс нь эхлээд хамгийн их өртөгтэй хүртэл жагсаав.

### 32.9.1. ISDN картууд

FreeBSD-ийн ISDN шийдэл нь идэвхгүй картыг ашиглаад зөвхөн DSS1/Q.931 (буюу Euro-ISDN) стандартыг дэмждэг. Firmware нь бас бусад дохионы протоколуудыг дэмждэг зарим идэвхтэй картуудыг дэмждэг; энэ нь бас хамгийн эхэнд дэмжигдсэн Primary Rate (PRI) буюу Анхдагч Хурд бүхий ISDN картыг оруулдаг.

isdn4bsd програм хангамж нь бусад ISDN чиглүүлэгчүүдтэй түүхий HDLC дээгүүрх IP юм уу эсвэл синхрон PPP: өөрчлөн засварласан sppp(4) драйвер `isppp`-тэй цөмийн PPP юм уу эсвэл хэрэглэгчийн талбарын ppp(8) ашиглан холбогдох боломжийг олгодог. Хэрэглэгчийн талбарын ppp(8) ашиглан хоёр буюу түүнээс дээш ISDN B-сувгуудын суваг холболтыг (channel bonding) хийх боломжтой байдаг. Утас хариулагч машины хэрэглээ бас байдаг байдаг бөгөөд бас програм хангамжийн 300 baud модем зэрэг олон хэрэгслүүд байдаг.

FreeBSD дээр дэмжигдсэн зарим PC ISDN картуудын тоо өсөн нэмэгдэж байгаа бөгөөд энэ нь бүх л Европ болон дэлхийн бусад олон хэсэгт амжилттайгаар ашиглагдаж байгаа мэдээллүүд байдаг.

Дэмжигдсэн идэвхгүй (пассив) ISDN картууд нь ихэвчлэн Infineon (хуучин Siemens) ISAC/HSCX/IPAC ISDN бичил схемтэй байдаг, гэхдээ бас Cologne Chip-ийн (зөвхөн ISA шугамд) бичил схемүүдтэй ISDN картууд, Winbond W6692 бичил схемүүдтэй PCI картууд, Tiger300/320/ISAC бичил схемийн хослолуудтой зарим картууд болон AVM Fritz!Card PCI V.1.0 болон AVM Fritz!Card PnP зэрэг зарим нэгэн үйлдвэрлэгчээс хамааралтай бичил схем бүхий картууд ордог.

Одоогоор идэвхтэй дэмжигдсэн ISDN картууд нь AVM B1 (ISA болон PCI) BRI картууд болон AVM T1 PCI PRI картууд юм.

isdn4bsd-ийн баримтын талаар isdn4bsd гарын авлага зэрэг илүү баримтыг заасан isdn4bsd-ийн эх хуудас хаягаас үзээрэй.

Хэрэв та өөр ISDN протокол, одоогоор дэмжигдээгүй байгаа ISDN PC картанд зориулж дэмжлэг нэмэх эсвэл isdn4bsd-ийг өргөжүүлэх сонирхолтой байгаа бол Hellmuth Michaelis <hm@FreeBSD.org>-тай холбоо барьна уу.

isdn4bsd-ийн суулгалт, тохиргоо болон алдааг олж засварлах тухай асуултуудад зориулагдсан freebsd-isdn захидлын жагсаалт байдаг.

### 32.9.2. ISDN Терминал Хувиргагчид

Терминал хувиргагчид (TA) нь модемууд нь ердийн утасны шугамд зориулагдсан байдаг шиг ISDN-ий хувьд зориулагдсан байдаг.

Ихэнх TA нь стандарт Hayes модемийн AT тушаалын олонлогийг ашигладаг бөгөөд модемийг шууд солих зориулалтаар ашиглаж болно.

TA нь ерөнхийдөө модемтой адил ажилладаг. Ялгаатай тал нь холболт болон дамжуулах хурдны хувьд таны хуучин модемоос хамаагүй хурдан байдаг явдал юм. Та модемийн тохиргоотой яг адил PPP-г тохируулах хэрэгтэй болно. Цуваа хурдыг хамгийн өндрөөр аль болох тавьсан эсэхээ шалгаарай.

TA-г ашиглахын гол давуу тал нь динамик PPP-ээр хийж болдогтой адил Интернэтийн үйлчилгээ үзүүлэгчтэй холбогдох явдал юм. IP хаягийн хэмжээ өдөр ирэх тутам багасаж байгаа өнөө үед ихэнх үйлчилгээ үзүүлэгчид танд статик IP өгөх хүсэлгүй байдаг. Ихэнх зориулагдсан чиглүүлэгчид динамик IP хуваарилалттай зохицож чаддаггүй.

TA-уудын боломжууд болон холболтын тогтвортой байдалд зориулан таны ажиллуулж байдаг PPP дэмонд TA-ууд нь бүрэн найддаг. Хэрэв та PPP-г аль хэдийн тохируулсан бол энэ нь FreeBSD машин дээр модемийг ашиглаж байгаад ISDN уруу хялбараар сайжруулах боломжийг олгодог. Гэхдээ танд тулгарсан PPP програм дээр байсан асуудлууд бас хэвээр үлдэх болно.

Хэрэв та максимум тогтвортой байдлыг хүсэж байгаа бол хэрэглэгчийн талбарын PPP биш цөмийн PPP тохируулгыг ашиглаарай.

Дараах TA-ууд нь FreeBSD-тэй ажилладаг:

- Motorola BitSurfer болон Bitsurfer Pro
- Adtran

Ихэнх бусад TA-ууд нь магадгүй бас ажиллах байх. TA үйлдвэрлэгчид нь өөрсдийнхөө бүтээгдэхүүнийг стандарт модемийн AT тушаалын олонлогийн ихэнхийг хүлээн авч чаддагаар хийхийг оролддог.

Гадаад TA-ийн жинхэнэ асуудал нь модемуудын адил таны компьютер дээр сайн цуваа карт хэрэгтэй явдал юм.

Та цуваа төхөөрөмжүүдийн тухай дэлгэрэнгүй ойлголт болон асинхрон ба синхрон цуваа портуудын ялгаануудын талаар FreeBSD Цуваа Тоног Төхөөрөмжийн зааврааc унших шаардлагатай.

Стандарт PC цуваа порт (асинхрон) дээр ажиллаж байгаа TA нь таныг 128 Kbs холболттой байсан ч гэсэн 115.2 Kbs хурдаар хязгаарладаг. ISDN-ий чаддаг 128 Kbs хурдыг бүрэн ашиглахын тулд та TA-г синхрон цуваа карт уруу шилжүүлэх шаардлагатай.

Дотоод TA худалдан авч синхрон/асинхрон асуудлаас тойрон гарчихлаа гэж битгий тэнэгтээрэй. Дотоод TA-ууд нь ердөө л өөр дээрээ бүтээгдсэн стандарт PC цуваа портын бичил схемтэй байдаг. Энэ нь ердөө л өөр цуваа кабель худалдан авч өөр хоосон цахилгаан сокет

олох асуудлыг хөнгөвчилдөг.

TA-тэй синхрон карт нь ядаж дангаараа зориулагдсан чиглүүлэгчтэй адил хурдан байдаг бөгөөд түүнийг энгийн FreeBSD хайрцаг удирдаж байгаа нь магадгүй илүү уян хатан юм.

Синхрон карт/TA болон зориулагдсан чиглүүлэгчийн алийг сонгох нь голчлон сүсэг бишрэлийн асуудал байдаг. Энэ талаар зарим хэлэлцүүлэг захидлын жагсаалтуудаар явагдсан байгаа. Бүрэн хэлэлцүүлгийг архивуудаас хайхыг бид санал болгож байна.

### 32.9.3. Зориулагдсан ISDN Гүүрнүүд/Чиглүүлэгчид

ISDN гүүрнүүд эсвэл чиглүүлэгчид нь зөвхөн FreeBSD эсвэл бусад үйлдлийн системийн хувьд биш юм. Чиглүүлэлт болон гүүр хийх технологийн талаар илүү дэлгэрэнгүй тайлбарыг сүлжээний ямар нэг номноос үзнэ үү.

Энэ хэсгийн хувьд чиглүүлэгч болон гүүр гэсэн ухагдахуунууд нь харилцан солигдож ашиглагдана.

Доод зэрэглэлийн ISDN чиглүүлэгчид/гүүрнүүдийн үнэ багасаж байгаа болохоор энэ нь улам илүү түгээмэл сонголт болох юм. ISDN чиглүүлэгч нь таны локал Ethernet сүлжээ уруу шууд залгагддаг жижиг хайрцаг байдаг бөгөөд бусад гүүр/чиглүүлэгчтэй хийх өөрийн холболтоо удирддаг. Энэ нь PPP болон бусад түгээмэл протоколуудын тусламжтай холболт хийх өөртөө цуг бүтээгдсэн програм хангамжтай байдаг.

Чиглүүлэгч нь бүрэн ISDN холболтыг ашиглах болохоор стандарт TA-аас хамаагүй илүү хурдан дамжуулах боломжийг олгодог.

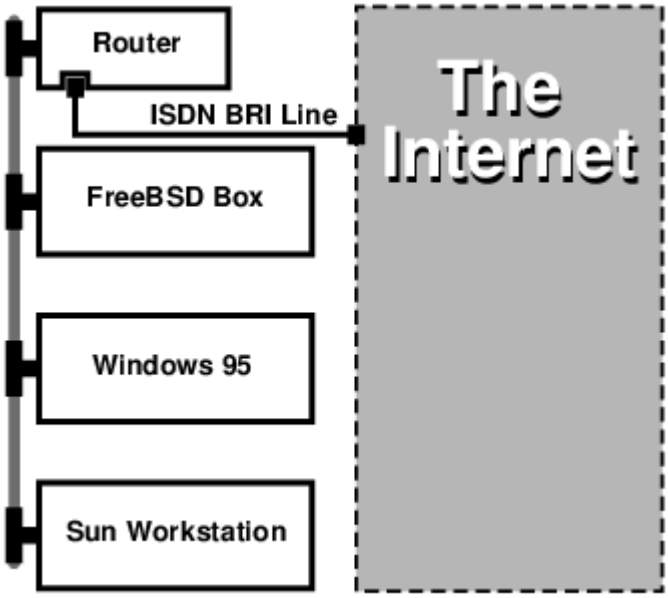
ISDN чиглүүлэгчид болон гүүрнүүдийн гол асуудал нь үйлдвэрлэгчид хоорондын харилцан ажиллах боломж асуудал хэвээр байгаа явдал юм. Хэрэв та Интернэтийн үйлчилгээ үзүүлэгчтэй холбогдохоор төлөвлөж байгаа бол тэд нартай өөрийн хэрэгцээний талаар хэлэлцэх нь зүйтэй юм.

Хэрэв та өөрийн гэрийн LAN-ийг оффисийн LAN-тай холбох зэргээр хоёр LAN сегментийг хооронд нь холбохоор төлөвлөж байгаа бол энэ нь хамгийн хялбар ажиллагаа багатай шийдэл юм. Та холболт хийгдэх гэж байгаа хоёр талдаа хоёуланд нь зориулан төхөөрөмж авах болохоор холболт ажиллах тал дээр итгэлтэй байж болно.

Жишээ нь гэрийн компьютер эсвэл салбар оффисийн сүлжээг гол оффисийн сүлжээтэй холбохын тулд дараах тохируулгыг ашиглаж болно:

*Жишээ 43. Салбар оффис эсвэл Гэрийн сүлжээ*

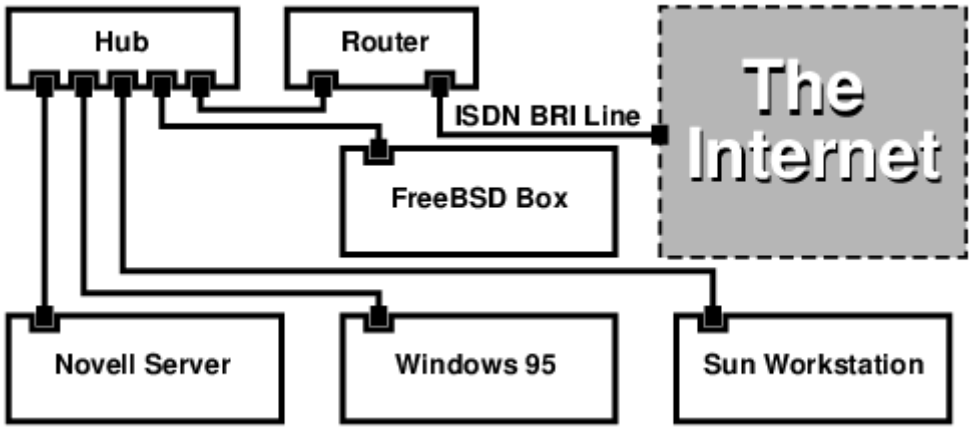
Сүлжээ нь 10 base 2 Ethernet ("thinnet") сүлжээний шугаман хэлбэрийг ашиглана. Шаардлагатай бол чиглүүлэгчийг сүлжээний кабельтай AUI/10BT хувиргагчаар холбоно.




Хэрэв таны гэр/салбар оффис нь зөвхөн нэг компьютер бол та зориулагдсан чиглүүлэгчийг холбохын тулд эрчилсэн хос зөрсөн кабелийг (twisted pair crossover cable) ашиглаж болно.

*Жишээ 44. Гол оффис эсвэл бусад LAN*

Сүлжээ нь 10 base T Ethernet ("Twisted Pair") сүлжээний од хэлбэрийг ашиглана.




Ихэнх чиглүүлэгчид/гүүрнүүдийн нэг том давуу тал нь *нэг зэрэг* 2 тусдаа сайт уруу 2 *тусдаа чөлөөтэй* PPP холболтуудтай байх боломжийг олгодог явдал юм. Хоёр цуваа порттой тусгай (ихэвчлэн үнэтэй) загваруудыг тооцохгүй бол энэ нь ихэнх ТА-ууд дээр дэмжигдээгүй байдаг. Үүнийг суваг холболт (channel bonding), MPP зэрэгтэй андуурч болохгүй.

Жишээ нь хэрэв та оффис дээрээ зориулагдсан ISDN холболттой бөгөөд түүнийг чагнахыг хүсэж байгаа боловч өөр нэг ISDN шугам авахыг хүсэхгүй байгаа бол энэ нь маш ашигтай боломж байж болох юм. Оффист байгаа чиглүүлэгч нь Интернэт уруу хийх зориулагдсан B сувгийн холболтыг (64 Kbps) удирдаж тусдаа өгөгдлийн холболтод зориулж өөр B сувгийг ашиглаж чадна. Хоёр дахь B суваг нь гаднаас залгах, гадагш залгах юм уу эсвэл илүү өргөн зурваст зориулж эхний B сувагтай цуг динамикаар холболт (bonding) хийхэд (MPP зэрэг)

ашиглагдаж болно.

Ethernet гүүр нь зөвхөн IP урсгал биш бас өөр илүүг дамжуулах боломжийг танд олгодог. Та IPX/SPX эсвэл өөрийн ашигладаг бусад протоколуудынхаа алийг ч бас илгээж болох юм.

# 32.10. Network Address Translation буюу Сүлжээний Хаягийн Хөрвүүлэлт

## 32.10.1. Тойм

FreeBSD-ийн Network Address Translation буюу Сүлжээний Хаягийн Хөрвүүлэлтийн дэмон нь natd(8) гэгддэг бөгөөд ирж байгаа түүхий IP пакетуудыг хүлээн авч эх хаягийг локал машиныхаар сольж эдгээр пакетуудыг гарч байгаа IP пакетийн урсгалд буцааж дахин оруулдаг. natd(8) нь өгөгдлийг буцаан хүлээн авахад өгөгдлийн анхны байрлалыг тодорхойлж чаддаг байхаар эхлэл IP хаяг болон портыг өөрчилж түүнийг анхлан хүсэгчид буцааж дамжуулдаг.

NAT-ийн хамгийн түгээмэл хэрэглээ бол Интернэтийн холболт хуваалцах гэгддэгийг хийж гүйцэтгэх явдал юм.

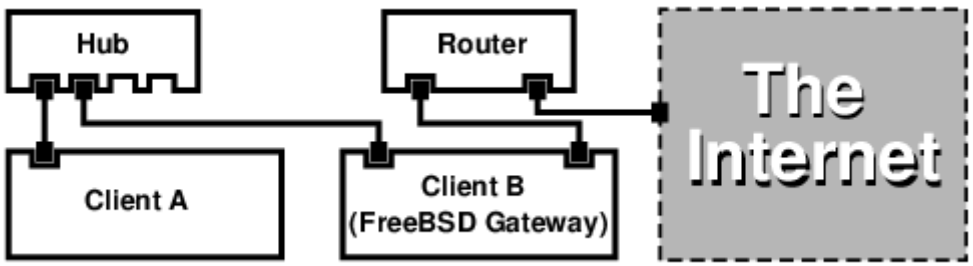
## 32.10.2. Анхны тохируулга

IP хаягийн хэмжээ багасаж байгаа болон кабель эсвэл DSL зэрэг өндөр хурдны хэрэглэгчийн шугамууд ашиглах хэрэглэгчдийн тоо ихэссэнээс болоод хүмүүст Интернэтийн холболт хуваалцах шийдлийн хэрэгцээ нэмэгдэх болсон билээ. Хэд хэдэн компьютеруудыг шууд нэг холболтоор холбох чадвар болон IP хаяг нь natd(8)-г боломжийн сонголт болгодог.

Хамгийн түгээмэл тохиолдол нь хэрэглэгч кабель эсвэл DSL шугаманд нэг IP хаягтай холбогдсон машинтай бөгөөд LAN-аар дамжуулан хэд хэдэн компьютеруудад Интернэтийн хандалт өгөхийн тулд энэ холбогдсон нэг компьютераа ашиглахыг хүсдэг явдал юм.

Үүнийг хийхийн тулд Интернэт дэх FreeBSD машин нь гарц маягаар ажиллах ёстой. Энэ гарц машин нь хоёр NIC-тэй байх ёстой-нэг нь Интернэтийн чиглүүлэгч уруу холбогдохын тулд, нөгөөх нь LAN уруу холбогдохын тулд ашиглагдах ёстой. LAN дахь бүх машинууд нь холбох төхөөрөмж (hub) юм уу эсвэл шилжүүлэгчээр (switch) холбогдсон байна.

FreeBSD гарцаар LAN-ийг Интернэттэй холбох олон арга замууд байдаг. Энэ жишээ нь хамгийн багаар бодоход хоёр NIC-тэй гарцыг зөвхөн тайлбарлах болно.



Үүнтэй адил тохиргоог Интернэтийн холболтыг хуваалцахын тулд түгээмэл ашигладаг. LAN машинуудын нэг нь Интернэтэд холбогдсон. Бусад машинууд нь тэр "гарц" машинаар Интернэтэд ханддаг.

### 32.10.3. Ачаалагч дуудагчийн тохиргоо

natd(8)-ийн тусламжтайгаар хийх сүлжээний хаягийн хөрвүүлэлтэд зориулсан цөмийн боломжууд GENERIC цөмд идэвхжүүлэгдээгүй байдаг боловч /boot/loader.conf файлд хоёр тохиргоо нэмж тэдгээрийг ачаалах явцад дуудахаар тохируулж болдог:

```
ipfw_load="YES"
ipdivert_load="YES"
```

Мөн `net.inet.ip.fw.default_to_accept` тохиргоог `1` болгож болно:

```
net.inet.ip.fw.default_to_accept="1"
```

Галт хана болон NAT гарцыг анх удаа тохируулахдаа энэ тохиргоог тохируулах нь зүйтэй байдаг. Ингэснээр ipfw(8)-ийн анхдагч бодлого нь илүү хаалттай `deny ip from any to any` гэсэн бодлого биш харин `allow ip from any to any` гэсэн илүү нээлттэй бодлого болох бөгөөд дахин ачаалалтын дараа системд хандах боломж хэвээр байх юм.

### 32.10.4. Цөмийн тохиргоо

Модулиудыг сонгодгоор биш эсвэл бүх шаардлагатай боломжуудыг цөмд оруулж бүтээхийг хүсэж байгаа бол дараах тохируулгууд цөмийн тохиргооны файлд байх ёстой:

```
options IPFIREWALL
options IPDIVERT
```

Мөн нэмэлтээр дараах нь бас тохиромжтой байж болох юм:

```
options IPFIREWALL_DEFAULT_TO_ACCEPT
options IPFIREWALL_VERBOSE
```

### 32.10.5. Системийн эхлүүлэх тохиргоо

Галт хана болон NAT дэмжлэгийг ачаалах явцад идэвхжүүлэхийн тулд дараах мөрүүдийг /etc/rc.conf файлд оруулж өгөх ёстой:

```
gateway_enable="YES" ①
firewall_enable="YES" ②
firewall_type="OPEN" ③
```

```
natd_enable="YES"
natd_interface="fxp0" ④
natd_flags="" ⑤
```

① Машиныг гарц маягаар ажиллахыг тохируулна. `sysctl net.inet.ip.forwarding=1` тушаалыг ажиллуулах нь бас адил үр дүнтэй. /etc/rc.firewall файл дахь галт ханын дүрмүүдийг ачаалах үед идэвхжүүлнэ.

② Энэ нь урьдчилан тодорхойлсон байдаг бүгдийг оруулахыг зөвшөөрдөг галт ханын дүрмийн олонлогийг заана. Нэмэлт төрлүүдийн талаар /etc/rc.firewall файлаас үзнэ үү.

③ Аль интерфэйс пакетуудыг нэвтрүүлэн дамжуулахыг заана (Интернэтэд холбогдсон интерфэйс).

④ Ачаалах үед natd(8)-д дамжуулах нэмэлт тохиргооны сонголтууд.

Өмнөх тохируулгуудыг /etc/rc.conf файлд тодорхойлсноор `natd -interface fxp0` тушаалыг ачаалах үед ажиллуулна. Үүнийг гараар бас ажиллуулж болно.

Их олон сонголтуудыг дамжуулах бол natd(8)-д зориулж тохиргооны файлыг ашиглаж бас болно. Энэ тохиолдолд дараах мөрийг /etc/rc.conf файлд нэмж тохиргооны файлыг тодорхойлох шаардлагатай:

```
natd_flags="-f /etc/natd.conf"
```

/etc/natd.conf файл нь тохиргооны сонголтуудын жагсаалтыг нэг мөрөнд нэгийг байхаар агуулдаг. Жишээ нь дараах хэсгийн хувьд доорх файлыг ашиглах болно:

```
redirect_port tcp 192.168.0.2:6667 6667
redirect_port tcp 192.168.0.3:80 80
```

Тохиргооны файлын тухай дэлгэрэнгүй мэдээллийг natd(8) гарын авлагын хуудасны `-f` тохируулгын талаар бичсэнээс лавлана уу.

LAN-ий ард байгаа машин болон интерфэйс бүр RFC 1918-д тодорхойлогдсон хувийн сүлжээний хаягаас заагдсан IP хаягтай, natd машины дотоод IP хаягийг анхдагч гарц болгон заасан байх ёстой.

Жишээ нь LAN-ий цаана байгаа клиент `A` болон `B` нь `192.168.0.2` болон `192.168.0.3` гэсэн IP хаягтай байх бол natd машины LAN интерфэйс нь `192.168.0.1` гэсэн IP хаягтай байна. Клиент `A` болон `B`-ийн анхдагч гарц natd машины `192.168.0.1`-ээр тохируулагдах ёстой. natd машины гадаад буюу Интернэтийн интерфэйс нь natd(8)-г ажиллуулахын тулд ямар нэгэн тусгайлсан өөрчлөлтийг шаарддаггүй.

### 32.10.6. Порт дахин чиглүүлэлт

natd(8)-ийн сул тал нь LAN клиентүүдэд Интернэтээс хандах боломжгүй байдаг. LAN дахь

клиентүүд нь гаднах ертөнц уруу чиглэсэн гадагшаа холболтуудыг хийж чадах боловч ирж байгааг хүлээн авч чаддаггүй. LAN клиент машинуудын аль нэгэн дээр Интернэтийн үйлчилгээнүүдийг ажиллуулахыг оролдох үед энэ нь асуудал үүсгэдэг. Үүнийг тойрох хялбар арга бол natd машины сонгосон Интернэтийн портуудыг LAN клиент уруу дахин чиглүүлэх явдал юм.

Жишээ нь A клиент дээр IRC сервер, B клиент дээр вэб сервер ажиллаж байна гэж үзье. Эдгээр нь зөв ажиллахын тулд 6667 (IRC) болон 80 (вэб) портууд дээр хүлээн авсан холболтууд нь тохирох машинууд уруу дахин чиглүүлэгдэх ёстой юм.

natd(8) уруу -redirect_port тохируулга зохих сонголтуудтай дамжуулагдах ёстой. Зөв бичих дүрэм нь ийм байна:

```
 -redirect_port proto targetIP:targetPORT[-targetPORT]
            [aliasIP:]aliasPORT[-aliasPORT]
            [remoteIP[:remotePORT[-remotePORT]]]
```

Дээрх жишээний хувьд нэмэлт өгөгдөл нь ийм байх ёстой:

```
-redirect_port tcp 192.168.0.2:6667 6667
-redirect_port tcp 192.168.0.3:80 80
```

```
Энэ нь зөв _tcp_ портуудыг LAN клиент машинууд уруу дахин чиглүүлэх болно.
```

-redirect_port нэмэлт өгөгдлийг порт тус бүрийг заахад ашиглахаас гадна бас бүлэг портуудыг заахад хэрэглэж болно. Жишээ нь *tcp 192.168.0.2:2000-3000 2000-3000* нь 2000-аас 3000 хүртэлх портууд дээр хүлээн авсан бүх холболтуудыг A клиентийн 2000-аас 3000 хүртэлх портууд уруу дахин чиглүүлэх болно.

Эдгээр тохируулгуудыг natd(8)-г шууд ажиллуулж байх үед /etc/rc.conf файлд natd_flags="" тохируулга дотор оруулж ашиглаж болохоос гадна тохиргооны файлын тусламжтай дамжуулж болно.

Тохиргооны сонголтуудын талаар natd(8)-с лавлана уу

### 32.10.7. Хаяг дахин чиглүүлэлт

Хаягийн дахин чиглүүлэлт нь хэд хэдэн IP хаягууд байгаа бөгөөд тэдгээр нь нэг машин дээр байх ёстой үед ашигтай байдаг. Үүнийг ашиглаад natd(8) нь өөрийн LAN клиент бүрдээ өөрийн гадаад IP хаягуудыг өгч чаддаг. Дараа нь natd(8) нь LAN клиентүүдээс ирэх гадагш чиглэсэн пакетуудыг зөв гадаад IP хаягаас болгон дахин бичиж тухайн IP хаяг дээр ирж байгаа бүх урсгалыг тохирох LAN клиент уруу дахин чиглүүлдэг. Энэ нь бас статик NAT гэж нэрлэгддэг. Жишээ нь 128.1.1.1, 128.1.1.2, болон 128.1.1.3 гэсэн IP хаягууд нь natd гарц машинд харьяалагдаж байна гэж үзье. 128.1.1.1 нь natd гарц машины гадаад IP хаяг болон ашиглагдаж болох бол 128.1.1.2 болон 128.1.1.3 нь A болон B LAN клиентүүд уруу буцааж дамжуулагдахаар байж болно.

`-redirect_address` бичлэг нь дараах маягийн байна:

```
-redirect_address localIP publicIP
```

| | |
|---|---|
| localIP | LAN клиентийн дотоод IP хаяг. |
| publicIP | LAN клиентэд тохирох гадаад IP хаяг. |

Энэ жишээний хувьд энэхүү нэмэлт өгөгдөл нь ингэж уншигдаж болно:

```
-redirect_address 192.168.0.2 128.1.1.2
-redirect_address 192.168.0.3 128.1.1.3
```

`-redirect_port` тохируулгын нэгэн адил эдгээр нэмэлт өгөгдлүүдийг /etc/rc.conf файлын `natd_flags=""` тохируулгад хийж өгч болохоос гадна тохиргооны файлаар дамжуулж болно. Хаягийн дахин чиглүүлэлт дээр тухайн нэг IP хаяг дээр хүлээн авсан бүх өгөгдөл дахин чиглүүлэгддэг учраас портын дахин чиглүүлэлт шаардлагагүй байдаг.

natd машины гадаад IP хаягууд нь идэвхтэй байж гадаад интерфэйсийн нэрээр нэрлэгдсэн байх ёстой. Үүнийг хийхийн тулд rc.conf(5)-г үзнэ үү.

## 32.11. IPv6

IPv6 (бас IPng "IP next generation" буюу дараа үеийн IP гэгддэг) нь олонд танигдсан IP протоколын (бас IPv4 гэгддэг) шинэ хувилбар юм. Одоогийн бусад *BSD системүүдийн адил FreeBSD нь KAME IPv6 лавлагаа шийдлийг агуулдаг. Тэгэхээр таны FreeBSD систем таныг IPv6-г туршихад хэрэгтэй бүхнийг агуулсан байдаг. Энэ хэсэг нь IPv6-г тохируулж ажиллуулах талаар анхаарлаа хандуулах болно.

1990 оны эхэн үеэс хүмүүс IPv4-ийн хаягийн хэмжээ хурдацтай багасаж байгааг мэдэх болсон. Интернэтийн өсөлтийн хурдаас хамаараад хоёр гол асуудал гарч ирсэн:

- Хаяггүй болж эхэлсэн. RFC1918 хувийн хаягийн талбар (`10.0.0.0/8`, `172.16.0.0/12`, болон `192.168.0.0/16`) болон Network Address Translation (NAT) буюу Сүлжээний хаягийн хөрвүүлэлтийг ашигладаг болсон болохоор өнөөдөр энэ нь тийм ч чухал асуудал байхаа больсон.
- Чиглүүлэгчийн хүснэгтийн оруулгууд хэтэрхий ихсэж байгаа явдал. Энэ нь өнөөдөр асуудал хэвээр байгаа билээ.

IPv6 нь эдгээр асуудлууд болон өөр бусад олныг шийддэг:

- 128 битийн хаягийн талбар. Өөрөөр хэлбэл онолын хувьд 340,282,366,920,938,463,463,374,607,431,768,211,456 хаяг байна. Энэ нь манай гариг дээрх нэг квадрат метр бүрт ойролцоогоор 6.67 * 10^27 IPv6 хаяг байна гэсэн үг юм.
- Чиглүүлэгчүүд нь зөвхөн сүлжээний нийлбэр хаягуудыг өөрсдийн чиглүүлэлтийн хүснэгтүүдэд хадгалах бөгөөд ингэснээр чиглүүлэлтийн хүснэгтийн дундаж хэмжээг 8192 хүртэл бууруулдаг.

IPv6-ийн бусад ашигтай олон боломжууд бас байдаг, тэдгээрээс дурдвал:

- Хаягийн автомат тохиргоо (RFC2462)
- Anycast хаягууд ("one-out-of many" буюу олноос ганц)
- Зайлшгүй шаардлагатай (mandatory) multicast хаягууд
- IPsec (IP security буюу нууцлаг)
- Хялбаршуулсан толгойн бүтэц
- Хөдөлгөөнт IP
- IPv6-to-IPv4 шилжилтийн арга замууд

Дэлгэрэнгүй мэдээллийн талаар доороос үзнэ үү:

- playground.sun.com дэх IPv6-ийн тухай тойм
- KAME.net

### 32.11.1. IPv6 хаягуудын тухай суурь мэдээлэл

IPv6 хаягуудын өөр өөр төрлүүд байдаг: Unicast, Anycast, болон Multicast.

Unicast хаягууд нь бидний мэддэг хаягууд юм. Unicast хаяг уруу илгээсэн пакет нь яг тэр хаягт харгалзах интерфэйс дээр ирдэг.

Anycast хаягуудыг unicast хаягуудаас ялгах боломжгүй байдаг боловч тэдгээр нь бүлэг интерфэйсүүдийг хаягладаг. Anycast хаяг уруу чиглэсэн пакет хамгийн ойрын (чиглүүлэгчийн хэмжилтийн хувьд) интерфэйс дээр ирэх болно. Anycast хаягуудыг зөвхөн чиглүүлэгчид ашиглаж болох юм.

Multicast хаягууд нь бүлэг интерфэйсүүдийг таньдаг. Multicast хаяг уруу чиглэсэн пакет нь multicast бүлэгт харьяалагдах бүх интерфэйсүүд дээр ирэх болно.

IPv4 цацах (broadcast) хаяг (ихэвчлэн `xxx.xxx.xxx.255`) нь IPv6-ийн multicast хаягуудаар илэрхийлэгддэг.

*Хүснэгт 14. Нөөц IPv6 хаягууд*

| IPv6 хаяг | Prefixlength буюу угтварын урт (битээр) | Тайлбар | Тэмдэглэл |
|---|---|---|---|
| `::` | 128 бит | тодорхойлогдоогүй | харьцуулбал IPv4 дэх `0.0.0.0` |
| `::1` | 128 бит | loopback буюу буцах хаяг | харьцуулбал IPv4 дэх `127.0.0.1` |
| `::00:xx:xx:xx:xx` | 96 бит | суулгагдсан IPv4 | Бага 32 бит нь IPv4 хаяг юм. Бас "IPv4 нийцтэй IPv6 хаяг" гэгддэг |

| IPv6 хаяг | Prefixlength буюу угтварын урт (битээр) | Тайлбар | Тэмдэглэл |
|---|---|---|---|
| `::ff:xx:xx:xx:xx` | 96 бит | IPv4-р илэрхийлэгдсэн IPv6 хаяг | Бага 32 бит нь IPv4 хаяг юм. IPv6-г дэмждэггүй хостуудад зориулсан. |
| `fe80::` - `feb::` | 10 бит | link-local | харьцуулбал IPv4 дэх loopback буюу буцах хаяг |
| `fec0::` - `fef::` | 10 бит | site-local | |
| `ff::` | 8 бит | multicast | |
| `001` (хоёртын системээр) | 3 бит | глобал unicast | Бүх глобал unicast хаягууд нь энэ цөөрмөөс заагдана. Эхний 3 бит нь "001" байна. |

### 32.11.2. IPv6 хаягуудыг унших нь

Дүрмийн дагуу ийм хэлбэртэй байна: `x:x:x:x:x:x:x:x` хэлбэрийн байх бөгөөд "x" бүр нь 16 битийн арван зургаат утга байна. Жишээ нь `FEBC:A574:382B:23C1:AA49:4592:4EFE:9982`

Хаяг нь ихэнхдээ дандаа тэгүүдээс тогтох урт дэд хэсгүүдээс тогтож болох бөгөөд хаягийн тийм хэсгийг "::" гэж хураангуйлж болно. Дөрвөн арван зургаат тооны гурав хүртэлх эхний "0"-үүдийг орхиж бас болно. Жишээ нь `fe80::1` нь дүрэмд заасан хэлбэрээр бол `fe80:0000:0000:0000:0000:0000:0000:0001` байна.

Гурав дахь хэлбэр нь сүүлийн 32 бит хэсгийг бидний мэддэг (аравтын) IPv4 загварын цэгүүдээр "." тусгаарлаж бичих явдал юм. Жишээ нь `2002::10.0.0.1` нь (арван зургаатын) дүрэмд зааснаар `2002:0000:0000:0000:0000:0000:0a00:0001` болох бөгөөд үүнийг бас `2002::a00:1` гэж бичиж болох юм.

Энэ хүртэл уншигч доор дурдсаныг ойлгож чадаж байх ёстой:

```
# ifconfig
```

```
rl0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        inet 10.0.0.10 netmask 0xffffff00 broadcast 10.0.0.255
        inet6 fe80::200:21ff:fe03:8e1%rl0 prefixlen 64 scopeid 0x1
        ether 00:00:21:03:08:e1
        media: Ethernet autoselect (100baseTX )
        status: active
```

`fe80::200:21ff:fe03:8e1%rl0` нь автоматаар тохируулагдсан link-local хаяг юм. Энэ нь MAC

хаягаас автомат тохиргооны хэсэг болон үүсгэгддэг.

IPv6 хаягуудын бүтцийн тухай дэлгэрэнгүй мэдээллийг RFC3513 хаягаас үзнэ үү.

### 32.11.3. Холбогдох нь

Одоогоор бусад IPv6 хостууд болон сүлжээнүүдтэй холбогдох дөрвөн арга байдаг:

- IPv6 үйлчилгээ үзүүлдэг эсэхийг мэдэхийн тулд өөрийн Интернэтийн үйлчилгээ үзүүлэгчид хандах хэрэгтэй.
- SixXS нь дэлхий даяар эцсийн хэрэглэгчдийн хооронд туннелийн үйлчилгээ үзүүлдэг.
- 6-аас-4 уруу туннель хийх (RFC3068)
- Хэрэв та dial-up холболт дээр байгаа бол net/freenet6 портыг ашиглах хэрэгтэй.

### 32.11.4. IPv6 ертөнц дэх DNS

IPv6-д зориулсан хоёр төрлийн DNS бичлэг байдаг. IETF нь A6 бичлэгийг хуучин, хэрэгцээнээс гарсан гэж зарласан. AAAA бичлэгүүд нь одоо стандарт болсон байгаа.

AAAA бичлэгүүдийг ашиглах нь амархан. Өөрийн хостын нэрийг дөнгөж авсан шинэ IPv6 хаягт зааж доор дурдсаныг:

```
MYHOSTNAME           AAAA    MYIPv6ADDR
```

өөрийн анхдагч бүсийн DNS файл уруу нэмж өгнө. Та өөрийн DNS бүсүүдийг удирддаггүй бол өөрийн DNS үйлчилгээ үзүүлэгчээс асуух хэрэгтэй. bind (8.3 ба 9 хувилбар) болон dns/djbdns-ийн (IPv6 засвартай) одоогийн хувилбарууд AAAA бичлэгүүдийг дэмждэг.

### 32.11.5. /etc/rc.conf файлд хэрэгцээтэй өөрчлөлтүүдийг хийх нь

#### 32.11.5.1. IPv6 клиентийн тохиргоонууд

Эдгээр тохиргоонууд нь таны LAN-д байх машиныг чиглүүлэгч биш клиент маягаар ажиллуулахаар тохируулахад танд туслах болно. Таны интерфэйсийг ачаалах үед автоматаар rtsol(8)-оор тохируулуулахын тулд FreeBSD 9.*x* ба түүнээс хойшхи хувилбарууд дээр дараах мөрийг:

```
ipv6_prefer="YES"
```

rc.conf файлд нэмнэ.

FreeBSD 8.*x* болон түүнээс өмнөх дээр дараахийг нэмнэ:

```
ipv6_enable="YES"
```

2001:471:1f11:251:290:27ff:fee0:2093 зэрэг IP хаягийг өөрийн fxp0 интерфэйсд статикаар өгөхийн тулд FreeBSD 9.*x*-н хувьд доор дурдсаныг нэмнэ:

```
ifconfig_fxp0_ipv6="inet6 2001:471:1f11:251:290:27ff:fee0:2093 prefixlen 64"
```

Компьютерийн сүлжээнд холбогдсон тэр дэд сүлжээний зөв утгыг оруулж *prefixlen 64*-г солихоо мартуузай.

FreeBSD 8*x* болон түүнээс өмнөхийн хувьд дараахийг нэмнэ:

```
ipv6_ifconfig_fxp0="2001:471:1f11:251:290:27ff:fee0:2093"
```

2001:471:1f11:251::1 анхдагч чиглүүлэгчийг зааж өгөхдөө доор дурдсаныг /etc/rc.conf файлд нэмнэ:

```
ipv6_defaultrouter="2001:471:1f11:251::1"
```

#### 32.11.5.2. IPv6 чиглүүлэгч/гарцын тохиргоонууд

Энэ нь таны туннелийн үйлчилгээ үзүүлэгчийн өгсөн чиглэлүүдийг авч тэдгээрийг дахин ачаалсан ч гэсэн үлдэхээр болгож тохиргоо болгон хөрвүүлэхэд танд туслах болно. Эхлүүлэх үед өөрийн туннелийг сэргээхийн тулд доор дурдсантай төстэйг /etc/rc.conf файлд ашиглана:

Тохируулах Ердийн Туннель хийх интерфэйсүүдийг жагсаана. Жишээ нь gif0:

```
gif_interfaces="gif0"
```

*MY_IPv4_ADDR* гэсэн локал төгсгөлтэй интерфэйсийг *REMOTE_IPv4_ADDR* гэсэн алсын төгсгөл уруу тохируулахын тулд:

```
gifconfig_gif0="MY_IPv4_ADDR REMOTE_IPv4_ADDR"
```

Өөрийн IPv6 туннелийн төгсгөл болгон ашиглахаар таны зааж өгсөн IPv6 хаягийг хамааруулахын тулд FreeBSD 9.*x* болон түүнээс хойшхийн хувьд доор дурдсаныг нэмнэ:

```
ifconfig_gif0_ipv6="inet6 MY_ASSIGNED_IPv6_TUNNEL_ENDPOINT_ADDR"
```

FreeBSD 8.*x* ба түүнээс өмнөхийн хувьд дараахийг нэмнэ:

```
ipv6_ifconfig_gif0="MY_ASSIGNED_IPv6_TUNNEL_ENDPOINT_ADDR"
```

Тэгээд таны хийх зүйл чинь IPv6-д зориулж анхдагч чиглүүлэлт тохируулах явдал юм. Энэ

нь IPv6 туннелийн нөгөө тал байх болно:

```
ipv6_defaultrouter="MY_IPv6_REMOTE_TUNNEL_ENDPOINT_ADDR"
```

#### 32.11.5.3. IPv6 туннелийн тохиргоонууд

Хэрэв сервер нь таны бусад сүлжээ болон гадаад ертөнцийн хооронд IPv6-г чиглүүлэх бол дараах /etc/rc.conf тохируулга бас хэрэгтэй болно:

```
ipv6_gateway_enable="YES"
```

### 32.11.6. Чиглүүлэгчийн зарлалт ба Хостын автомат тохиргоо

Энэ хэсэг нь IPv6 анхдагч чиглүүлэлтийг зарлахын тулд rtadvd(8)-г тохируулахад тань туслах болно.

rtadvd(8)-г идэвхжүүлэхийн тулд таны /etc/rc.conf файлд дараах тохиргоо хэрэгтэй болно:

```
rtadvd_enable="YES"
```

IPv6 чиглүүлэгчийн төлөөлөлт хийгдэх интерфэйсийг заах нь чухал юм. Жишээ нь rtadvd(8) -д fxp0-г ашиглахыг хэлж өгөхийн тулд доор дурдсаныг нэмнэ:

```
rtadvd_interfaces="fxp0"
```

Одоо бид /etc/rtadvd.conf тохиргооны файл үүсгэх ёстой. Энд жишээ байна:

```
fxp0:\
    :addrs#1:addr="2001:471:1f11:246::":prefixlen#64:tc=ether:
```

fxp0-г өөрийн ашиглах интерфэйсээр солих хэрэгтэй.

Дараа нь `2001:471:1f11:246::`-г өөрийн хуваарилалтын угтвараар солиорой.

Хэрэв та `/64` дэд сүлжээг зориулсан бол өөр бусад ямар ч өөрчлөлт хийх хэрэггүй. Хэрэв тийм биш бол та `prefixlen#` буюу угтварын уртыг зөв утгаар өөрчлөх шаардлагатай.

## 32.12. Asynchronous Transfer Mode (ATM) буюу Асинхрон Дамжуулах Горим

### 32.12.1. ATM дээгүүр классик IP тохируулах нь (PVC-нүүд)

Classical IP over ATM (CLIP) буюу ATM дээгүүрх классик IP нь Asynchronous Transfer Mode

(ATM) буюу Асинхрон Дамжуулах Горимыг IP-тай ашиглах хамгийн энгийн арга юм. Энэ нь switched буюу шилжүүлэгдсэн (SVC-нүүд) ба permanent буюу байнгын (PVC-нүүд) холболтуудтай ашиглагдаж болдог. Энэ хэсэг нь сүлжээн дээр тулгуурласныг PVC-нүүд дээр хэрхэн тохируулах талаар тайлбарладаг.

#### 32.12.1.1. Бүрэн тор хэлбэрийн тохиргоонууд

CLIP-ийг PVC-нүүдтэй тохируулах эхний арга нь машин бүрийг сүлжээн дэх бусад машин бүртэй зориулагдсан PVC-ээр холбох явдал юм. Үүнийг тохируулах хялбар байдаг боловч их олон тооны машинууд дээр бүтэшгүй байдаг. Энэ жишээ нь сүлжээнд бидэнд дөрвөн машин байгаа бөгөөд машин бүр ATM сүлжээнд ATM хувиргагч картын тусламжтай холбогдсон гэж үздэг. Эхний алхам нь IP хаягууд болон машинуудын хооронд ATM холболтуудыг төлөвлөх явдал юм. Бид доор дурдсаныг ашиглана:

| Хост | IP хаяг |
|---|---|
| hostA | 192.168.173.1 |
| hostB | 192.168.173.2 |
| hostC | 192.168.173.3 |
| hostD | 192.168.173.4 |

Бүрэн тор хэлбэрийн сүлжээг бүтээхийн тулд машинуудын хослол бүрийн хооронд нэг ATM холболт бидэнд хэрэгтэй:

| Машинууд | VPI.VCI хослол |
|---|---|
| hostA - hostB | 0.100 |
| hostA - hostC | 0.101 |
| hostA - hostD | 0.102 |
| hostB - hostC | 0.103 |
| hostB - hostD | 0.104 |
| hostC - hostD | 0.105 |

Холболтын төгсгөл бүр дэх VPI болон VCI утгууд нь мэдээж ялгаатай байх боловч амархан байлгах үүднээс бид тэдгээрийг адил гэж үзнэ. Дараа нь бид ATM интерфэйсүүдийг хост бүр дээр тохируулах хэрэгтэй болно:

```
hostA# ifconfig hatm0 192.168.173.1 up
hostB# ifconfig hatm0 192.168.173.2 up
hostC# ifconfig hatm0 192.168.173.3 up
hostD# ifconfig hatm0 192.168.173.4 up
```

Ингэхдээ бүх хостууд дээр ATM интерфэйс нь hatm0 гэж үзэх болно. Одоо PVC-нүүд нь hostA дээр тохируулагдах шаардлагатай (бид тэдгээрийг ATM шилжүүлэгч дээр аль хэдийн тохируулагдсан гэж үзнэ, үүнийг хэрхэн хийхийг шилжүүлэгчийн гарын авлагаас лавлах хэрэгтэй).

```
hostA# atmconfig natm add 192.168.173.2 hatm0 0 100 llc/snap ubr
hostA# atmconfig natm add 192.168.173.3 hatm0 0 101 llc/snap ubr
hostA# atmconfig natm add 192.168.173.4 hatm0 0 102 llc/snap ubr

hostB# atmconfig natm add 192.168.173.1 hatm0 0 100 llc/snap ubr
hostB# atmconfig natm add 192.168.173.3 hatm0 0 103 llc/snap ubr
hostB# atmconfig natm add 192.168.173.4 hatm0 0 104 llc/snap ubr

hostC# atmconfig natm add 192.168.173.1 hatm0 0 101 llc/snap ubr
hostC# atmconfig natm add 192.168.173.2 hatm0 0 103 llc/snap ubr
hostC# atmconfig natm add 192.168.173.4 hatm0 0 105 llc/snap ubr

hostD# atmconfig natm add 192.168.173.1 hatm0 0 102 llc/snap ubr
hostD# atmconfig natm add 192.168.173.2 hatm0 0 104 llc/snap ubr
hostD# atmconfig natm add 192.168.173.3 hatm0 0 105 llc/snap ubr
```

Мэдээж хэрэг UBR-с бусад урсгалын хэлэлцээнүүд хэрэв ATM хувиргагч тэдгээрийг дэмждэг бол ашиглагдаж болно. Энэ тохиолдолд урсгалын хэлэлцээний нэр нь урсгалын нэмэлт өгөгдлүүдийн дараа орж байна. atmconfig(8) хэрэгсэлд зориулсан тусламжийг дараах тушаалыг ашиглан:

```
# atmconfig help natm add
```

авч болох бөгөөд эсвэл atmconfig(8) гарын авлагаас авч болно.

Үүнтэй адил тохиргоог /etc/rc.conf ашиглан бас хийж болно. `hostA`-ийн хувьд энэ нь иймэрхүү харагдана:

```
network_interfaces="lo0 hatm0"
ifconfig_hatm0="inet 192.168.173.1 up"
natm_static_routes="hostB hostC hostD"
route_hostB="192.168.173.2 hatm0 0 100 llc/snap ubr"
route_hostC="192.168.173.3 hatm0 0 101 llc/snap ubr"
route_hostD="192.168.173.4 hatm0 0 102 llc/snap ubr"
```

Бүх CLIP чиглүүлэлтүүдийн тухайн үеийн төлвийг дараах тушаалаар авч болно:

```
hostA# atmconfig natm show
```

## 32.13. Common Address Redundancy Protocol (CARP)

Common Address Redundancy Protocol буюу CARP нь олон хостуудад нэг IP хаяг хуваалцах боломжийг олгодог. Зарим тохиргоонуудын хувьд үүнийг байнгын бэлэн байдал юм уу эсвэл ачаалал тэгш хуваарилахад ашиглаж болох юм. Энд дурдсан жишээний адилаар хостууд нь бас өөр өөр IP хаягууд ашиглаж болох юм.

CARP-ийн дэмжлэгийг идэвхжүүлэхийн тулд FreeBSD цөмийг FreeBSD цөмийг тохируулах нь бүлэгт заасны дагуу дараах тохируулгатайгаар бүтээсэн байх хэрэгтэй:

```
device  carp
```

Мөн өөрөөр if_carp.ko модулийг ачаалах үед дуудаж болно. Ингэхийн тулд дараах мөрийг /boot/loader.conf нэмж өгнө:

```
if_carp_load="YES"
```

CARP боломж нь одоо хүртээмжтэй болсон байх ёстой бөгөөд түүнийг хэд хэдэн `sysctl`-ийн OID-уудын тусламжтайгаар тааруулж болно:

| OID | Тайлбар |
| --- | --- |
| `net.inet.carp.allow` | Ирж байгаа CARP пакетуудыг хүлээж авах. Анхдагчаар идэвхжсэн байдаг. |
| `net.inet.carp.preempt` | Энэ тохируулга нь хост дээрх бүх CARP интерфэйсүүдийг аль нэг нь зогссон тохиолдолд зогсоодог. Анхдагчаар идэвхжээгүй байдаг. |
| `net.inet.carp.log` | `0` гэсэн утга нь бүртгэлийг идэвхжүүлдэггүй. `1` гэсэн утга нь муу CARP пакетуудын бүртгэлийг идэвхжүүлдэг. `1`-ээс их утгууд нь CARP интерфэйсүүдийн хувьд төлвийн өөрчлөлтүүдийн бүртгэлийг идэвхжүүлдэг. Анхдагч утга нь `1` байна. |
| `net.inet.carp.arpbalance` | ARP ашиглан дотоод сүлжээний урсгалыг тэнцвэржүүлнэ. Анхдагчаар идэвхжээгүй байдаг. |
| `net.inet.carp.suppress_preempt` | Preemption буюу өөр илүү өндөр давуу эрх бүхий бодлого/урсгалаар солигдох боломжийг хаасан хориглолтын төлвийг үзүүлэх зөвхөн уншигдах OID. Хэрэв интерфэйс дээрх холбоос босоогүй бол илүү өндөр давуу эрх бүхий бодлого/урсгалаар солигдох боломжийг хааж болно. `0` утга нь илүү өндөр давуу эрх бүхий бодлого/урсгалаар солигдох боломж хаагдаагүй гэсэн утгатай. Асуудал бүр энэ OID-ийг нэгээр ахиулж нэмэгдүүлнэ. |

CARP төхөөрөмжүүдийг `ifconfig` тушаалаар үүсгэж болно:

```
# ifconfig carp0 create
```

Жинхэнэ амьдрал дээр эдгээр интерфэйсүүдэд VHID гэгддэг давхардахгүй дугаарууд хэрэгтэй болно. Энэ VHID буюу Виртуал Хостын Үнэмлэх нь сүлжээн дэх хостыг ялгахад хэрэглэгдэх болно.

### 32.13.1. Серверт хандах боломжийг байнга хангахын тулд CARP-ийг ашиглах нь (CARP)

Дээр дурдсан CARP-ийн нэг хэрэглээ нь серверийн бэлэн байдалд зориулагджээ. Энэ жишээ нь өөр өөр IP хаягуудтай, ижил вэбийн агуулга бүхий гурван хостын хувьд байнгын бэлэн байдалд байх дэмжлэгийг харуулах болно. Эдгээр машинууд нь Тойрон Эргэх DNS тохиргоотойгоор ажиллах болно. Амжилттүй болсон үед хандах нөөц машин нь хоёр нэмэлт CARP интерфэйстэй байх бөгөөд агуулгын сервер бүрийн IP-ний хувьд ашиглагдана. Хандалт амжилттүй болоход амжилттүй болсон машины IP хаягийг нөөц сервер авах ёстой. Энэ нь хэрэглэгчид амжилттүй болсон нь мэдэгдэхгүй гэсэн үг юм. Нөөц сервер нь ачааллыг авах ёстой бол бусад агуулгын серверүүдтэй ижил агуулга болон үйлчилгээнүүдтэй байх ёстой.

Хоёр машин нь өгөгдсөн хостын нэр болон VHID-үүдээс бусдаараа адилхан тохируулагдсан байх ёстой. Энэ жишээн дээр эдгээр машинуудыг `hosta.example.org` болон `hostb.example.org` гэж нэрлэсэн байгаа. Эхлээд CARP-ийн тохиргоонд шаардлагатай мөрүүдийг rc.conf файлд нэмэх хэрэгтэй. `hosta.example.org` хостын хувьд rc.conf файл доор дурдсан мөрүүдийг агуулсан байх ёстой:

```
hostname="hosta.example.org"
ifconfig_fxp0="inet 192.168.1.3 netmask 255.255.255.0"
cloned_interfaces="carp0"
ifconfig_carp0="vhid 1 pass testpass 192.168.1.50/24"
```

`hostb.example.org` дээр доор дурдсан мөрүүдийг rc.conf файл агуулсан байх ёстой:

```
hostname="hostb.example.org"
ifconfig_fxp0="inet 192.168.1.4 netmask 255.255.255.0"
cloned_interfaces="carp0"
ifconfig_carp0="vhid 2 pass testpass 192.168.1.51/24"
```

Хамгийн чухал нь `ifconfig` тушаалд `pass` тохируулгаар заагдсан нууц үгс нь адил байх ёстой. carp төхөөрөмжүүд нь зөвхөн зөв нууц үг бүхий машинуудаас ирэх зарлалтуудыг сонсох болно. VHID нь машин бүрийн хувьд өөр өөр байх ёстой.

Гурав дахь `provider.example.org` машиныг аль нэг хост руу хандахад амжилттүй болоход хандаж болохоор болгож бэлдэх хэрэгтэй. Энэ машинд хоёр carp төхөөрөмж хэрэгтэй болох бөгөөд нэг нэгээрээ хоёр хосттой ажиллах юм. rc.conf-ийн зөв тохиргооны мөрүүд доор дурдсантай төсөөтэй байх болно:

```
hostname="provider.example.org"
ifconfig_fxp0="inet 192.168.1.5 netmask 255.255.255.0"
cloned_interfaces="carp0 carp1"
ifconfig_carp0="vhid 1 advskew 100 pass testpass 192.168.1.50/24"
ifconfig_carp1="vhid 2 advskew 100 pass testpass 192.168.1.51/24"
```

Хоёр carp төхөөрөмжтэй байх нь хариу өгөхөө байсан аль нэг машиныг мэдэж IP хаягийг нь авах боломжийг `provider.example.org` хостод олгох болно.

Анхдагч FreeBSD цөм дээр preemption буюу өөр илүү өндөр давуу эрх бүхий бодлого/урсгалаар солигдох боломж идэвхжсэн байж *болох* юм. Хэрэв тийм бол `provider.example.org` нь IP хаягийг анхдагч агуулгын сервер рүү суллахгүй байж болох юм. Энэ тохиолдолд администратор мастер уруу буцааж IP-г хүчилж болох юм. Дараах тушаалыг `provider.example.org` дээр ажиллуулах шаардлагатай:

```
# ifconfig carp0 down && ifconfig carp0 up
```

Хостод тохирох carp интерфэйс дээр үүнийг хийх ёстой.

Энэ хүрэхэд CARP бүрмөсөн идэвхжиж тест хийхэд бэлэн болсон байх ёстой. Сүлжээг дахин эхлүүлэх буюу эсвэл машинуудыг дахин ачаалаад тест хийгээрэй.

Дэлгэрэнгүй мэдээллийг carp(4) гарын авлагын хуудаснаас олж болно.

# Chapter 33. Хавсралтууд

# Appendix A: Se procurer FreeBSD

## A.1. CDROM болон DVD Нийтлэгчид

### A.1.1. CD болон DVD цуглуулгууд

FreeBSD CD болон DVD цуглуулгуудыг олон жижиглэн худалдаалагчдаас шууд худалдан авах боломжтой байдаг:

- FreeBSD Mall, Inc.
  2420 Sand Creek Rd C-1 #347
  Brentwood, CA 94513
  USA
  Утас: +1 925 240-6652
  Факс: +1 925 674-0821
  Email: <info@freebsdmall.com>
  WWW: http://www.freebsdmall.com/
- Dr. Hinner EDV
  Kochelseestr. 11
  D-81371 München
  Germany
  Утас: (0177) 428 419 0
  WWW: http://www.hinner.de/linux/freebsd.html
- Linux Distro UK
  42 Wharfedale Road
  Margate
  CT9 2TB
  United Kingdom
  WWW: https://linux-distro.co.uk/
- The Linux Emporium
  The Techno Centre, Puma Way
  Parkside
  CV1 2TT
  United Kingdom
  Утас: +44 (0)247 615 8121
  Факс: +44 1491 837016
  WWW: http://www.linuxemporium.co.uk/products/bsd/
- LinuxCenter.Ru
  Galernaya Street, 55
  Saint-Petersburg
  190000
  Russia
  Утас: +7-812-3125208
  Email: <info@linuxcenter.ru>
  WWW: http://linuxcenter.ru/shop/freebsd

# A.2. FTP сайтууд

FreeBSD-д зориулсан абан ёсны эхүүдийг дэлхий даяар байрласан толин тусгал сайтуудаас авч болно. ftp://ftp.FreeBSD.org/pub/FreeBSD/ сайт нь сайн холбогдсон бөгөөд маш олон тооны холболтуудыг зөвшөөрдөг. Гэхдээ та "илүү ойр" толин тусгал сайтыг хайж олсон нь дээр биз ээ (ялангуяа хэрэв та ямар нэгэн толин тусгал маягийн сайт тохируулж ажиллуулахыг хүсвэл).

Мөн FreeBSD нь дараах толин тусгал сайтууд дахь нэргүй FTP-ээс авч болохоор байдаг. Хэрэв та нэргүй FTP-ээс FreeBSD-г авахыг сонговол өөртөө аль болох ойр сайтыг ашиглаад үзээрэй. "Primary Mirror Sites" буюу үндсэн толин тусгал сайтууд дахь толин тусгал сайтууд нь бүхэл FreeBSD архивыг (архитектур болгонд зориулсан одоо байгаа бүх хувилбарууд) ихэвчлэн агуулдаг боловч танай улс юм уу эсвэл бүсэд байрлаж байгаа сайтаас татаж авахад танд илүү хурдан байх болов уу. Бүсийн сайтууд нь хамгийн түгээмэл архитектуруудад зориулсан хамгийн сүүлийн үеийн хувилбаруудыг агуулдаг боловч FreeBSD-ийн бүх архивыг агуулаагүй байж болох юм. Бүх сайтууд нь нэргүй FTP-ээр хандах боломжийг олгодог. Зарим сайтууд нь бас бусад аргуудаар хандах боломжийг олгодог. Сайт болгоны хувьд хандах аргууд нь хостын нэрийн дараа хаалтанд өгөгдсөн байгаа болно.

Central Servers, Primary Mirror Sites, Armenia, Australia, Austria, Brazil, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hong Kong, Ireland, Japan, Korea, Latvia, Lithuania, Netherlands, New Zealand, Norway, Poland, Russia, Saudi Arabia, Slovenia, South Africa, Spain, Sweden, Switzerland, Taiwan, Ukraine, United Kingdom, United States of America.

(as of UTC)

**Central Servers**

ftp://ftp.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp.FreeBSD.org/pub/FreeBSD/ / http://ftp.FreeBSD.org/pub/FreeBSD/)

**Primary Mirror Sites**

In case of problems, please contact the hostmaster <mirror-admin@FreeBSD.org> for this domain.

- ftp://ftp1.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp4.FreeBSD.org/pub/FreeBSD/ / http://ftp4.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp5.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp7.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp10.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp10.FreeBSD.org/pub/FreeBSD/ / http://ftp10.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp11.FreeBSD.org/pub/FreeBSD/ (ftp)

- ftp://ftp13.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp14.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp14.FreeBSD.org/pub/FreeBSD/)

**Armenia**

In case of problems, please contact the hostmaster <hostmaster@am.FreeBSD.org> for this domain.

- ftp://ftp1.am.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp1.am.FreeBSD.org/pub/FreeBSD/ / rsync)

**Australia**

In case of problems, please contact the hostmaster <hostmaster@au.FreeBSD.org> for this domain.

- ftp://ftp.au.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.au.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.au.FreeBSD.org/pub/FreeBSD/ (ftp)

**Austria**

In case of problems, please contact the hostmaster <hostmaster@at.FreeBSD.org> for this domain.

- ftp://ftp.at.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp.at.FreeBSD.org/pub/FreeBSD/ / http://ftp.at.FreeBSD.org/pub/FreeBSD/)

**Brazil**

In case of problems, please contact the hostmaster <hostmaster@br.FreeBSD.org> for this domain.

- ftp://ftp2.br.FreeBSD.org/FreeBSD/ (ftp / http://ftp2.br.FreeBSD.org/)
- ftp://ftp3.br.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)
- ftp://ftp4.br.FreeBSD.org/pub/FreeBSD/ (ftp)

**Czech Republic**

In case of problems, please contact the hostmaster <hostmaster@cz.FreeBSD.org> for this domain.

- ftp://ftp.cz.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp.cz.FreeBSD.org/pub/FreeBSD/ / http://ftp.cz.FreeBSD.org/pub/FreeBSD/ / http://ftp.cz.FreeBSD.org/pub/FreeBSD/ / rsync / rsyncv6)
- ftp://ftp2.cz.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.cz.FreeBSD.org/pub/FreeBSD/)

**Denmark**

In case of problems, please contact the hostmaster <staff@dotsrc.org> for this domain.

- ftp://ftp.dk.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp.dk.FreeBSD.org/pub/FreeBSD/ / http://ftp.dk.FreeBSD.org/pub/FreeBSD/)

**Estonia**

In case of problems, please contact the hostmaster <hostmaster@ee.FreeBSD.org> for this domain.

- ftp://ftp.ee.FreeBSD.org/pub/FreeBSD/ (ftp)

**Finland**

In case of problems, please contact the hostmaster <hostmaster@fi.FreeBSD.org> for this domain.

- ftp://ftp.fi.FreeBSD.org/pub/FreeBSD/ (ftp)

**France**

In case of problems, please contact the hostmaster <hostmaster@fr.FreeBSD.org> for this domain.

- ftp://ftp.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp1.fr.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp1.fr.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp3.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.fr.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)
- ftp://ftp7.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.fr.FreeBSD.org/pub/FreeBSD/ (ftp)

**Germany**

In case of problems, please contact the hostmaster <de-bsd-hubs@de.FreeBSD.org> for this domain.

- ftp://ftp.de.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp1.de.FreeBSD.org/freebsd/ (ftp / http://www1.de.FreeBSD.org/freebsd/ / rsync://rsync3.de.FreeBSD.org/freebsd/)
- ftp://ftp2.de.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.de.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp4.de.FreeBSD.org/FreeBSD/ (ftp / http://ftp4.de.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp5.de.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp7.de.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp7.de.FreeBSD.org/pub/FreeBSD/)

**Greece**

In case of problems, please contact the hostmaster <hostmaster@gr.FreeBSD.org> for this domain.

- ftp://ftp.gr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.gr.FreeBSD.org/pub/FreeBSD/ (ftp)

**Hong Kong**

ftp://ftp.hk.FreeBSD.org/pub/FreeBSD/ (ftp)

**Ireland**

In case of problems, please contact the hostmaster <hostmaster@ie.FreeBSD.org> for this domain.

- ftp://ftp3.ie.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)

**Japan**

In case of problems, please contact the hostmaster <hostmaster@jp.FreeBSD.org> for this domain.

- ftp://ftp.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp7.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp9.jp.FreeBSD.org/pub/FreeBSD/ (ftp)

**Korea**

In case of problems, please contact the hostmaster <hostmaster@kr.FreeBSD.org> for this domain.

- ftp://ftp.kr.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)
- ftp://ftp2.kr.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.kr.FreeBSD.org/pub/FreeBSD/)

**Latvia**

In case of problems, please contact the hostmaster <hostmaster@lv.FreeBSD.org> for this domain.

- ftp://ftp.lv.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.lv.FreeBSD.org/pub/FreeBSD/)

**Lithuania**

In case of problems, please contact the hostmaster <hostmaster@lt.FreeBSD.org> for this domain.

- ftp://ftp.lt.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.lt.FreeBSD.org/pub/FreeBSD/)

**Netherlands**

In case of problems, please contact the hostmaster <hostmaster@nl.FreeBSD.org> for this domain.

- ftp://ftp.nl.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.nl.FreeBSD.org/os/FreeBSD/ / rsync)
- ftp://ftp2.nl.FreeBSD.org/pub/FreeBSD/ (ftp)

**New Zealand**

- ftp://ftp.nz.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.nz.FreeBSD.org/pub/FreeBSD/)

**Norway**

In case of problems, please contact the hostmaster <hostmaster@no.FreeBSD.org> for this domain.

- ftp://ftp.no.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)

**Poland**

In case of problems, please contact the hostmaster <hostmaster@pl.FreeBSD.org> for this domain.

- ftp://ftp.pl.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp2.pl.FreeBSD.org

**Russia**

In case of problems, please contact the hostmaster <hostmaster@ru.FreeBSD.org> for this domain.

- ftp://ftp.ru.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.ru.FreeBSD.org/FreeBSD/ / rsync)
- ftp://ftp2.ru.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.ru.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp5.ru.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp5.ru.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp6.ru.FreeBSD.org/pub/FreeBSD/ (ftp)

**Saudi Arabia**

In case of problems, please contact the hostmaster <ftpadmin@isu.net.sa> for this domain.

- ftp://ftp.isu.net.sa/pub/ftp.freebsd.org (ftp)

**Slovenia**

In case of problems, please contact the hostmaster <hostmaster@si.FreeBSD.org> for this domain.

- ftp://ftp.si.FreeBSD.org/pub/FreeBSD/ (ftp)

**South Africa**

In case of problems, please contact the hostmaster <hostmaster@za.FreeBSD.org> for this domain.

- ftp://ftp.za.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.za.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.za.FreeBSD.org/pub/FreeBSD/ (ftp)

**Spain**

In case of problems, please contact the hostmaster <hostmaster@es.FreeBSD.org> for this domain.

- ftp://ftp.es.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.es.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp3.es.FreeBSD.org/pub/FreeBSD/ (ftp)

**Sweden**

In case of problems, please contact the hostmaster <hostmaster@se.FreeBSD.org> for this domain.

- ftp://ftp.se.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.se.FreeBSD.org/pub/FreeBSD/ (ftp / rsync://ftp2.se.FreeBSD.org/)
- ftp://ftp3.se.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.se.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp4.se.FreeBSD.org/pub/FreeBSD/ / http://ftp4.se.FreeBSD.org/pub/FreeBSD/ / http://ftp4.se.FreeBSD.org/pub/FreeBSD/ / rsync://ftp4.se.FreeBSD.org/pub/FreeBSD/ / rsync://ftp4.se.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp6.se.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp6.se.FreeBSD.org/pub/FreeBSD/)

**Switzerland**

In case of problems, please contact the hostmaster <hostmaster@ch.FreeBSD.org> for this domain.

- ftp://ftp.ch.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.ch.FreeBSD.org/pub/FreeBSD/)

**Taiwan**

In case of problems, please contact the hostmaster <hostmaster@tw.FreeBSD.org> for this domain.

- ftp://ftp.ch.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp.tw.FreeBSD.org/pub/FreeBSD/ / rsync / rsyncv6)
- ftp://ftp2.tw.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp2.tw.FreeBSD.org/pub/FreeBSD/ / http://ftp2.tw.FreeBSD.org/pub/FreeBSD/ / http://ftp2.tw.FreeBSD.org/pub/FreeBSD/ / rsync / rsyncv6)
- ftp://ftp4.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.tw.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp6.tw.FreeBSD.org/ / rsync)
- ftp://ftp7.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp11.tw.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp11.tw.FreeBSD.org/FreeBSD/)
- ftp://ftp12.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp13.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp14.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp15.tw.FreeBSD.org/pub/FreeBSD/ (ftp)

**Ukraine**

- ftp://ftp.ua.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.ua.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp6.ua.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp6.ua.FreeBSD.org/pub/FreeBSD / rsync://ftp6.ua.FreeBSD.org/FreeBSD/)
- ftp://ftp7.ua.FreeBSD.org/pub/FreeBSD/ (ftp)

**United Kingdom**

In case of problems, please contact the hostmaster <hostmaster@uk.FreeBSD.org> for this domain.

- ftp://ftp.uk.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.uk.FreeBSD.org/pub/FreeBSD/ (ftp / rsync://ftp2.uk.FreeBSD.org/ftp.freebsd.org/pub/FreeBSD/)
- ftp://ftp3.uk.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.uk.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.uk.FreeBSD.org/pub/FreeBSD/ (ftp)

**United States of America**

In case of problems, please contact the hostmaster <hostmaster@us.FreeBSD.org> for this domain.

- ftp://ftp1.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.us.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp4.us.FreeBSD.org/pub/FreeBSD/ / http://ftp4.us.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp5.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp10.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp11.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp13.us.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp13.us.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp14.us.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp14.us.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp15.us.FreeBSD.org/pub/FreeBSD/ (ftp)

# A.3. Нэргүй CVS(хуучирсан)

## A.3.1. Анхааруулга

CVS нь төслийн хувьд хуучирсан бөгөөд ашиглахыг зөвлөдөггүй. Түүний оронд Subversion-г ашиглах ёстой.

# A.4. CTM-г ашиглах нь

CTM нь алсын сангийн модыг төвтэй синхрончлол хийхэд зориулсан арга юм. Энэ нь FreeBSD-ийн эх моднуудтай ашиглахад зориулагдаж хөгжүүлэгдсэн бөгөөд цаг хугацаа өнгөрөх тусам бусад хүмүүс бусад зорилгоор хэрэглэхэд ашигтайг мэдэж болох юм. Дельтанууд үүсгэх процессийн талаар бараг баримт байхгүй болохоор бас хэрэв та CTM-г бусад зүйлсэд зориулж ашиглахыг хүсэж байгаа бол дэлгэрэнгүй мэдээллийн талаар ctm-users-desc захидлын жагсаалтад хандана уу.

### A.4.1. CTM-г би яагаад ашиглах ёстой вэ?

CTM нь танд FreeBSD эх моднуудын локал хуулбарыг өгдөг. Модны төрөл бүрийн "хэлбэрүүд" байдаг. Таны бүхэл CVS мод эсвэл аль нэг салбаруудын нэгийг дагахыг хүсэж байгаагаас хамаарч CTM нь танд мэдээлэл өгч чадна. Та FreeBSD-ийн идэвхтэй хөгжүүлэгч боловч олиггүй юм уу эсвэл TCP/IP холболт байхгүй эсвэл өөрчлөлтүүдийг автоматаар тан руу илгээгддэг байлгахыг хүсэж байгаа бол CTM нь танд зориулагджээ. Та хамгийн идэвхтэй салбаруудын хувьд өдөрт гурав хүртэлх дельтануудыг авах хэрэгтэй болно. Гэхдээ та тэдгээрийг автомат цахим захидлаар илгээгддэг байхыг бодох хэрэгтэй. Шинэчлэлтүүдийн хэмжээг үргэлж аль болох бага байлгадаг. Энэ ихэвчлэн 5K-с бага байдаг, хааяа (арваас нэгт) 10-50K хэмжээтэй байдаг бөгөөд заримдаа том 100K+ эсвэл илүү бас байдаг.

Та урьдчилан багцалсан хувилбарын оронд хөгжүүлэлтийн эхүүд дээр шууд ажиллахтай холбоотой гардаг төрөл бүрийн анхааруулгуудын талаар бас мэдэж байх хэрэгтэй болно. Энэ нь таныг "current" эхүүдийг сонгох тохиолдолд ялангуяа үнэн байдаг. Таныг FreeBSD-ийн одоо үеийн хэлбэрт байх нь хэсгийг уншихыг зөвлөдөг.

### A.4.2. CTM-г ашиглахад надад юу хэрэгтэй вэ?

Танд хоёр зүйл хэрэгтэй: CTM програм болон түүнийг тэжээх эхний дельтанууд ("current" түвшингүүдэд хүрэхийн тулд).

CTM програм нь хувилбар 2.0 гарснаас хойш FreeBSD-ийн хэсэг болж ирсэн бөгөөд хэрэв танд эх байгаа бол энэ нь /usr/src/usr.sbin/ctm санд байдаг.

CTM-г тэжээх "дельтануудыг" нь FTP эсвэл цахим захидал гэсэн хоёр аргаар авч болно. Хэрэв танд Интернет уруу хандах ерөнхий FTP хандалт байгаа бол дараах FTP сайтууд нь CTM-д хандахыг дэмждэг:

ftp://ftp.FreeBSD.org/pub/FreeBSD/CTM/

эсвэл толин тусгалууд хаягийг үзнэ үү.

тохирох сан руу FTP хийгээд README файлыг татаж аваад тэндээс эхлээрэй.

Хэрэв та өөрийн дельтануудыг цахим захидлаар авахыг хүсэж байгаа бол:

CTM түгээлтийн жагсаалтуудын аль нэгэнд бүртгүүлэх хэрэгтэй. ctm-cvs-cur-desc нь бүхэл CVS модыг дэмждэг. ctm-src-cur-desc нь хөгжүүлэлтийн салбарын толгойг дэмждэг. ctm-src-9-desc нь 9.X хувилбарыг дэмждэг, гэх мэт. (Хэрэв та өөрийгөө жагсаалтад хэрхэн бүртгүүлэхийг мэдэхгүй байгаа бол дээрх жагсаалтын нэрэн дээр дарах юм уу эсвэл https://lists.freebsd.org уруу очоод өөрийн хүсэж байгаа жагсаалтан дээрээ дарах хэрэгтэй. Бүртгэхэд шаардлагатай бүгдийг жагсаалтын хуудас агуулсан байх ёстой.)

Өөрийн CTM шинэчлэлтүүдийг хүлээн авч эхлэхдээ та тэдгээрийг задалж ашиглахдаа `ctm_rmail` тушаал ашиглаж болох юм. Хэрэв та процессийг бүрэн автомат загвараар ажилладаг байлгахыг хүсэж байгаа бол /etc/aliases файл дахь оруулгаас `ctm_rmail` програмыг шууд ашиглаж болох юм. Илүү дэлгэрэнгүйг `ctm_rmail`-ийн гарын авлагын хуудаснаас шалгана уу.

CTM дельтануудыг авах ямар аргыг ашиглаж байгаагаас үл хамааран та ctm-announce-desc захидлын жагсаалтад бүртгүүлэх хэрэгтэй. Ирээдүйд CTM системийн ажиллагааны тухай зарлалууд илгээгдэх газар нь зөвхөн энэ болох юм. Дээрх жагсаалтын нэрэн дээр дарж жагсаалтад бүртгүүлэхийн тулд зааврууды дагах хэрэгтэй.

### A.4.3. CTM-г эхний удаа хэрэглэх

CTM дельтануудыг ашиглаж эхлэхээсээ өмнө дельтанууд дараа дараагийн удаа үүсгэгдэхэд нь зориулж та эхлэх цэг дээр ирэх хэрэгтэй болно.

Танд юу байгааг эхлээд тодорхойлох хэрэгтэй. Бүгд "хоосон" сангаас эхэлж болно. Өөрийн CTM-ээр дэмжигдсэн модыг эхлүүлэхийн тулд та эхний "Empty" буюу хоосон дельтаг ашиглах ёстой. Тодорхой нэг цэгт эдгээр "started" буюу эхэлсэн дельтануудын аль нэгийг өөрийн хүслээр CD дээр түгээж болохоор энэ нь зориулагдсан боловч одоогоор тийм юм болоогүй байгаа билээ.

Моднууд нь хэдэн арван мегабайт байдаг болохоор өөртөө байгаагаас эхлэхийг эрхэмлэх ёстой юм. Хэрэв танд -RELEASE CD байгаа бол та эхний эхийг түүнэс хуулж эсвэл задалж болох юм. Энэ нь өгөгдлийн чухал дамжуулалтыг хэмнэх болно.

Та эдгээр "starter" буюу эхлэх дельтануудыг дугаарт нэмэгдсэн X-ээр таньж болно (жишээ нь src-cur.3210XEmpty.gz). X-ийн дараах тэмдэглээс нь таны эхний "seed" буюу үрийн үүсэлд харгалзана. Дүрмээр бол Empty буюу хоосноос шилжих үндсэн шилжилт нь 100 дельта бүрээс үүсгэгдэнэ. Ингэхэд тэдгээр нь том байна! XEmpty дельтануудын хувьд 70-аас 80 мегабайт gzip хийгдсэн өгөгдөл нь нийтлэг байдаг.

Эхлэх анхдагч дельтаг сонгосны дараа танд түүний дараах өндөр дугааруудтай бүх дельтанууд хэрэгтэй болно.

### A.4.4. Өөрийн өдөр тутмын амьдралдаа CTM-г ашиглах нь

Дельтануудыг өгөхийн тулд ердөө л доор дурдсаныг хийнэ:

```
# cd /where/ever/you/want/the/stuff
# ctm -v -v /where/you/store/your/deltas/src-xxx.*
```

CTM нь gzip-ээр тавигдсан дельтануудыг ойлгодог болохоор та тэдгээрийг эхлээд gunzip хийх хэрэггүй юм. Энэ нь дискний зай хэмнэх болно.

Бүх процессийн талаар маш нууцлаг гэж бодоогүй л бол CTM нь таны модыг хөндөхгүй байх болно. Дельтаг шалгахын тулд та -c тугийг бас ашиглаж болох бөгөөд CTM нь таны модыг үнэндээ хөндөхгүй; энэ нь зөвхөн дельтаны бүрэн бүтэн байдлыг шалгаж таны одоогийн модонд цэвэр хийгдэх эсэхийг л үздэг.

CTM-д өгөгдөх бусад сонголтууд бас байдаг бөгөөд эдгээрийг гарын авлагын хуудаснуудаас үзнэ үү, эсвэл илүү дэлгэрэнгүйг эхүүдээс хайна уу.

Ингээд л бүгд бараг болж байна. Шинэ дельта авах бүртээ өөрийн эхүүдийг хамгийн сүүлийн хувилбарт аваачихын тулд түүнийг CTM-ээр ажиллуулах хэрэгтэй.

Дельтануудыг дахин татаж авахад хэцүү байгаа бол битгий устгаарай. Ямар нэг муу юм болж болзошгүй гэж үзээд тэдгээрийг та хаа нэгтэй хадгалахыг хүсэж болох юм. Танд зөвхөн уян диск байсан ч гэсэн `fdwrite` тушаалыг ашиглан хуулбар хийх нь зүйтэй.

## A.4.5. Өөрийн локал өөрчлөлтүүдийг хадгалах нь

Хөгжүүлэгчийн хувьд эх модыг өөрчлөх эсвэл түүн дээр турших хүсэл төрж болох юм. CTM нь локал өөрчлөлтүүдийг хязгаарлалттайгаар дэмждэг: foo файлыг байгаа эсэхийг шалгахын өмнө foo.ctm файлыг эхлээд хайдаг. Хэрэв энэ файл байвал CTM нь foo-ийн оронд энэ файл дээр ажилладаг.

Энэ нь бидэнд локал өөрчлөлтүүдийг хялбараар арчлах боломжийг олгодог: өөрийн өөрчлөх төлөвлөгөөтэй байгаа файлуудаа .ctm өргөтгөлтэй болгон хуулах хэрэгтэй. Дараа нь та кодыг чөлөөтэй оролдож болох бөгөөд CTM нь .ctm файлыг хамгийн сүүлийн хэлбэрт байлгах болно.

## A.4.6. CTM-ийн бусад сонирхолтой сонголтууд

### A.4.6.1. Шинэчлэлтээр яг юу хөндөгдөхийг олох

Та өөрийн эх архивт CTM-ийн хийх өөрчлөлтүүдийн жагсаалтыг `-l` сонголтыг CTM-д өгөн тодорхойлж болно.

Та өөрчлөлтүүдийн бүртгэлүүдийг хадгалах, өөрчлөгдсөн файлуудад урьдчилан эсвэл дараа нь боловсруулалт хийх эсвэл ердөө л хэтэрхий зовниж байгаа бол энэ нь ашигтай юм.

### A.4.6.2. Шинэчлэхээсээ өмнө нөөцлөлтүүд хийх

Заримдаа та CTM шинэчлэлтээр өөрчлөгдөх бүх файлуудыг нөөцөлж авахыг хүсэж болох юм.

`-B backup-file` сонголтыг зааснаар CTM-ийг өгөгдсөн CTM дельтагаар хөндөгдөх бүх файлуудыг нөөцөлж backup-file руу хадгалахад хүргэнэ.

### A.4.6.3. Шинэчлэлтээр хөндөгдөх файлуудыг хязгаарлах

Заримдаа та өгөгдсөн CTM шинэчлэлтийн хүрээг хязгаарлах сонирхолтой эсвэл дельтануудын дарааллаас цөөн хэдэн файлуудыг задлах сонирхолтой байж болох юм.

CTM-ийн ажилладаг файлуудын жагсаалтыг та `-e` болон `-x` сонголтуудыг ашиглан шүүлт хийх регулар илэрхийллүүдийг заан хянаж болно.

Жишээ нь өөрийн хадгалсан CTM дельтануудын цуглуулгаас lib/libc/Makefile-ийн хамгийн сүүлийн хуулбарыг задалж авахын тулд дараах тушаалуудыг ажиллуулна:

```
# cd /where/ever/you/want/to/extract/it/
```

```
# ctm -e '^lib/libc/Makefile' ~ctm/src-xxx.*
```

CTM дельтанд заагдсан файл бүрийн хувьд тушаалын мөрөнд өгөгдсөн дарааллаар `-e` болон `-x` сонголтуудыг өгсөн. Файл нь бүх `-e` болон `-x` сонголтууд түүнд өгөгдсөний дараа зөвхөн сонгогдох эрхтэй гэж тэмдэглэгдсэн бол CTM-ээр боловсруулагддаг.

### A.4.7. CTM-ийн ирээдүйн төлөвлөгөөнүүд

Маш их:

- CTM-ийн хуурамч шинэчлэлтүүдийг илрүүлэхийн тулд CTM системд ямар нэгэн нэвтрэлт танилт ашиглах.
- CTM-ийн сонголтуудыг цэвэрлэх, тэдгээр нь толгой эргүүлэм, зөн совингийн эсрэг болсон.

### A.4.8. Бусад зүйлс

`ports` цуглуулгад зориулсан дельтануудын дараалал бас байдаг боловч сонирхол тийм ч өндөр байдаггүй.

### A.4.9. CTM толин тусгалууд

CTM/FreeBSD нь дараах толин тусгал сайтууд дахь нэргүй FTP-ээр хандаж авах боломжтой байдаг. Хэрэв та CTM-г нэргүй FTP-ээр авахыг сонгосон бол өөртөө ойр сайтыг оролдож үзнэ үү.

Асуудал гарсан тохиолдолд ctm-users-desc захидлын жагсаалтад хандана уу.

**Калифорни, Bay Area, албан ёсны эх**

- ftp://ftp.FreeBSD.org/pub/FreeBSD/development/CTM/

**Өмнөд Африк, хуучин дельтануудад зориулсан нөөц сервер**

- ftp://ftp.za.FreeBSD.org/pub/FreeBSD/CTM/

**Тайвань/R.O.C.**

- ftp://ctm.tw.FreeBSD.org/pub/FreeBSD/development/CTM/
- ftp://ctm2.tw.FreeBSD.org/pub/FreeBSD/development/CTM/
- ftp://ctm3.tw.FreeBSD.org/pub/FreeBSD/development/CTM/

Хэрэв та өөртөө ойрхон толин тусгалыг олоогүй бол эсвэл толин тусгал нь бүрэн биш байгаа бол alltheweb зэрэг хайлтын системийг ашиглаад үзээрэй.

## A.5. Subversion ашиглах нь

### A.5.1. Танилцуулга

2012 оны 7 сараас FreeBSD-н бүх эх код, баримтжуулалт болон портын цуглуулгыг хадгалахдаа Subversion (*svn*)-г үндсэн хувилбар удирдах систем болгон ашиглаж эхэлсэн.

Subversion хөгжүүлэлтийн хэрэгсэл юм. Ихэнх хэрэглэгчид FreeBSD-г шинэчлэх хэсэгт зааснаар FreeBSD-н үндсэн системийг шинэчлэх ёстой бөгөөд FreeBSD-н портын цуглуулгыг шинэчлэхдээ Portsnap-г ашиглах ёстой.

Subversion-д URL-г ашиглан репозиторыг заадаг бөгөөд `protocol://hostname/path` хэлбэрийн байна. Доор заасан шиг толин тусгалууд өөр өөр протоколуудыг дэмжиж болно. Замын эхний хэсэг нь хандаж болох FreeBSD-н репозитор юм. Гурван өөр репозитор байдаг: FreeBSD-н үндсэн системийн эх кодод зориулсан `base`, Портын цуглуулгад зориулсан `ports`, баримтжуулалтад зориулсан `doc` байдаг. Жишээ нь URL `svn://svn0.us-east.FreeBSD.org/ports/head/` нь svn протоколыг ашиглан `svn0.us-east.FreeBSD.org` толин дээрх портын репозиторын гол салбарыг зааж байна.

### A.5.2. Суулгах

Subversion-г аль ч репозиторын агуулгыг татаж авахаас өмнө суулгасан байх шаардлагатай. Хэрэв портын мод байгаа бол Subversion-г ингэж суулгаж болно:

```
# cd /usr/ports/devel/subversion
# make install clean
```

Хэрэв портын мод байхгүй бол Subversion-г багц хэлбэрээр суулгаж болно:

```
# pkg_add -r subversion
```

Хэрэв pkgng ашиглан багцыг удирддаг бол Subversion-г ингэж суулгана:

```
# pkg install devel/subversion
```

### A.5.3. Subversion-г ажиллуулах

Эхийн цэвэр хувилбарыг татаж локал сан руу авахад `svn` тушаалыг ашиглаж болно. Энэ санд байгаа файлуудыг *локал ажлын хувилбар* гэдэг.

Локал сан нь үүссэн боловч `svn` тушаалаар үүсгэгдээгүй бол татаж авахаасаа өмнө нэрийг солих юм уу устгаарай. `svn`-ээр үүсгэгдээгүй сан дээр татаж авах тохиолдолд татаж авч байгаа файлууд нь байгаа файлуудтай зөрөх асуудал үүсч болзошгүй.

Өгөгдсөн репозитороос татаж авахдаа энэ маягийн тушаалыг:

```
# svn checkout svn-mirror/repository/branch lwcdir
```

гэж ажиллуулах бөгөөд:

- *svn-mirror* нь Subversion толин сайтууд дахь аль нэгийн URL юм.
- *repository* нь төслийн аль нэг репозитор бөгөөд жишээ нь `base`, `ports`, эсвэл `doc`.
- *branch* нь ашиглаж байгаа репозитороос хамаарна. `ports` ба `doc` нь `head` салбарт ихэвчлэн шинэчлэгддэг бол `base` нь -CURRENT-н хамгийн сүүлийн хувилбарыг `head` дотор агуулж байдаг бөгөөд -STABLE салбарын сүүлийн хувилбарууд нь `stable/8` ( 8.*x*) болон `stable/9` (9.*x*) дотор байдаг.
- *lwcdir* нь зааж өгсөн салбарын агуулга байрлах сан юм. Энэ нь ихэвчлэн `ports`-н хувьд /usr/ports, `base`-н хувьд /usr/src, `doc`-н хувьд /usr/doc байдаг.

Энэ нь Портын цуглуулгыг АНУ-н баруунд байрлах репозитороос HTTPS протокол ашиглан локал ажлын хуулбарыг /usr/ports-д санд байршуулж байгаа жишээ юм. Хэрэв /usr/ports сан байгаа бөгөөд `svn`-ээр үүсгэгдээгүй бол өөрчлөх юм уу устгахаа мартуузай.

```
# svn checkout https://svn0.us-west.FreeBSD.org/ports/head /usr/ports
```

Алсын репозиторын бүрэн салбарыг эхний удаа татаж авах учраас удаж магадгүй. Тэвчээртэй байгаарай.

Эхний удаа татсаныхаа дараа локал ажлын хуулбарыг шинэчлэхдээ:

```
# svn update lwcdir
```

Дээрх жишээн дээр үүсгэсэн /usr/ports-г шинэчлэхдээ:

```
# svn update /usr/ports
```

Татаж авахаас шинэчлэх нь хурдан бөгөөд өөрчлөгдсөн файлуудыг л татаж авдаг.

Татаж авсныхаа дараа локал ажлын хуулбарыг шинэчлэх өөр нэг арга нь /usr/ports, /usr/src, ба /usr/doc сангууд дахь Makefile-д байдаг. `SVN_UPDATE`-г зааж өгөөд `update` гэж хэрэглэнэ. Жишээ нь /usr/src-г шинэчлэхийн тулд:

```
# cd /usr/src
# make update SVN_UPDATE=yes
```

### A.5.4. Дэлгэрэнгүй мэдээлэл

Subversion ашиглах талаар бусад мэдээллийг Version Control with Subversion нэртэй "Subversion ном", эсвэл Subversion баримтжуулалтаас үзнэ үү.

## A.6. Subversion толин сайтууд

Бүх толин тусгалууд бүх репозиторыг агуулдаг.

FreeBSD-н мастер Subversion сервер svn.FreeBSD.org-д олон нийт хандах боломжтой байх бөгөөд зөвхөн уншигдах боломжтой байдаг. Энэ нь ирээдүйд өөрчлөгдөж болох бөгөөд хэрэглэгчдийг аль болох албан ёсны толин тусгалуудыг ашиглахыг зөвлөдөг. FreeBSD Subversion репозиториудыг хөтөч ашиглан харахын тулд http://svnweb.FreeBSD.org/ хаягийг ашиглаарай.

FreeBSD svn толин тусгалын сүлжээ нь эхний шатандаа байгаа бөгөөд өөрчлөгдөх магадлалтай. Эдгээр толин тусгалуудыг статик гэж тооцоолж болохгүй. Ялангуяа серверүүдийн SSL сертификатууд хэзээ нэг цагт шинэчлэгдэх болно.

| Нэр | Протоколууд | Байрлал | SSL хэш |
|---|---|---|---|
| svn0.us-west.FreeBSD.org | svn, http, https | USA, California | SHA1 79:35:8F:CA:6D:34:D9:30:44:D1:00:AF:33:4D:E6:11:44:4D:15:EC |
| svn0.us-east.FreeBSD.org | svn, http, https | USA, New Jersey | SHA1 06:D1:23:DE:5E:7A:F7:2B:7A:7E:74:95:5F:54:8D:5C:B0:D6:2E:8F |

HTTPS нь FreeBSD толин тусгалыг хуурамчаар дуурайх ("man in the middle" халдлага гэгддэг) эсвэл эцсийн хэрэглэгч рүү муу агуулга шидэх зэргээс хамгаалах боломжтой ашиглахыг зөвлөдөг протокол юм.

HTTPS толин тусгал руу хийх эхний холболтын үеэр хэрэглэгчээс серверийн *fingerprint*-г асуудаг:

```
Error validating server certificate for 'https://svn0.us-west.freebsd.org:443':
 - The certificate is not issued by a trusted authority. Use the
   fingerprint to validate the certificate manually!
Certificate information:
 - Hostname: svnmir.ysv.FreeBSD.org
 - Valid: from Fri, 24 Aug 2012 22:04:04 GMT until Sat, 24 Aug 2013 22:04:04 GMT
 - Issuer: clusteradm, FreeBSD.org, CA, US
 - Fingerprint: 79:35:8f:ca:6d:34:d9:30:44:d1:00:af:33:4d:e6:11:44:4d:15:ec
(R)eject, accept (t)emporarily or accept (p)ermanently?
```

Fingerprint-г дээрх хүснэгттэй харьцуулж хараарай. Хэрэв хурууны хээ таарч байгаа бол серверийн аюулгүй байдлын сертификатыг түр эсвэл бүрмөсөн зөвшөөрч болно. Түр зуурын сертификат нь сервертэй хийх нэг удаагийн сессээр дуусгавар болох бөгөөд дараагийн удаа шалгах алхам дахин хийгдэх болно. Сертификатыг бүрмөсөн зөвшөөрснөөр нэвтрэх эрхийг ~/.subversion/auth/ санд хадгалж сертификатын хугацаа дуустал хэрэглэгчээс хурууны хээг дахин шалгахыг асуудаггүй.

Хэрэв HTTPS-г галт хана эсвэл бусад асуудлаас болоод хэрэглэх боломжгүй бол арай илүү хурдан SVN дараагийн сонголт байж болох юм. Алийг ч ашиглах боломжгүй бол HTTP-г ашиглаарай.

# A.7. CVSup-г ашиглах нь (хуучирсан)

## A.7.1. Танилцуулга

CVS нь төслийн хувьд хуучирсан бөгөөд ашиглахыг зөвлөдөггүй. Түүний оронд Subversion-г ашиглах ёстой.

CVSup нь алсын сервер хост дээр байгаа мастер CVS архиваас эх модыг шинэчлэх болон түгээхэд зориулагдсан програм хангамжийн багц юм. FreeBSD-ийн эхүүд Калифорни дахь төв хөгжүүлэлтийн машин дээрх CVS архивт арчлагдаж байдаг. CVSup-ийн тусламжтай FreeBSD хэрэглэгчид өөрсдийн эх модыг хялбараар хамгийн сүүлийн хэлбэрт аваачиж чаддаг байна.

CVSup нь шинэчлэлтийн *pull* буюу *mamax* загварыг ашигладаг. Татах загвар дээр клиент бүр серверээс шинэчлэлтийг хүссэн үедээ эсвэл хэрэгтэй үедээ асуудаг. Сервер нь өөрийн хэрэглэгчдээс ирэх хүсэлтүүдийг идэвхгүйгээр хүлээж байдаг. Сервер хэзээ ч хүсээгүй шинэчлэлтүүдийг илгээдэггүй. Хэрэглэгчид нь шинэчлэлтийг авахын тулд CVSup клиентийг гараар ажиллуулах юм уу эсвэл түүнийг автоматаар давтамжтайгаар ажиллуулахын тулд `cron` ажил тохируулах ёстой болно.

Ингэж том үсгээр бичсэн CVSup ухагдахуун нь бүхэл програм хангамжийн багцыг хэлнэ. Түүний гол бүрэлдэхүүн хэсгүүд нь хэрэглэгч бүрий машин дээр ажилладаг `cvsup` клиент, ба FreeBSD-ийн толин тусгал сайтууд дээр ажиллаж байдаг `cvsupd` сервер юм.

csup хэрэгсэл нь CVSup-г C дээр бичсэн хувилбар юм. Үүний хамгийн том давуу тал бол илүү хурдан бөгөөд Modula-3 хэлээс хамаардаггүй, тиймээс та түүнийг суулгах шаардлагагүй байдаг. Мөн түүнийг шууд ашиглаж болно, учир нь энэ нь үндсэн системд орсон байдаг. Хэрэв та csup-г ашиглахаар шийдсэн бол CVSup-ийн суулгалт дахь алхмуудыг алгасаж энэ нийтлэлийн үлдсэн хэсгийг дагаж байхдаа CVSup гэснийг csup болгож солиорой.

## A.7.2. Суулгалт

CVSup-г суулгах хамгийн хялбар арга бол FreeBSD-ийн багцын цуглуулгаас урьдчилан эмхэтгэсэн net/cvsup багцыг ашиглах явдал юм. Хэрэв та CVSup-г эхээс бүтээхийг эрхэмлэж байгаа бол net/cvsup портыг ашиглаарай. Гэхдээ урьдчилан анхааруулъя: net/cvsup порт нь Modula-3 системээс хамаардаг бөгөөд үүнийг татан авч суулгахад ихээхэн хэмжээний дискний зай болон хугацаа шаарддаг.

Хэрэв та Xorg суулгагдаагүй сервер машин дээр CVSup-г ашиглах гэж байгаа бол CVSupGUI-г агуулаагүй net/cvsup-without-gui портыг ашиглах хэрэгтэй.

## A.7.3. CVSup тохиргоо

CVSup-ийн ажиллагаа supfile гэгддэг тохиргооны файлаар хянагддаг. /usr/shared/examples/cvsup/ санд зарим жишээ supfiles байдаг.

supfile дахь мэдээлэл CVSup-д зориулж дараах асуултуудад хариулдаг:

- Ямар файлуудыг та хүлээн авахыг хүсэж байна вэ?
- Тэдгээрийн ямар хувилбаруудыг та хүсэж байна вэ?
- Тэдгээрийг та хаанаас авахыг хүсэж байна вэ?
- Та тэдгээрийг өөрийн машиныхаа хаана байрлуулахыг хүсэж байна вэ?
- Та өөрийн төлвийн файлуудыг хаана байрлуулахыг хүсэж байна вэ?

Дараах хэсгүүдэд бид эдгээр асуултуудад хариулан түгээмэл supfile-г бүтээх болно. Эхлээд бид supfile-ийн ерөнхий бүтцийг тайлбарлах болно.

supfile нь текст файл юм. Тайлбарууд нь `#` тэмдэгтээс эхэлж мөр дуустал үргэлжилнэ. Хоосон мөрнүүд болон зөвхөн тайлбараас тогтох мөрүүдийг орхидог.

Үлдэж байгаа мөр болгон хэрэглэгчийн авахыг хүсэж байгаа файлуудын цуглуулгыг тайлбарладаг. Мөр нь файлуудын "цуглуулга" буюу файлуудын логик бүлэглэлийн нэрээс эхэлдэг бөгөөд эдгээрийг сервер тодорхойлдог. Цуглуулгын нэр таны аль файлуудыг хүсэж байгааг серверт хэлдэг. Цуглуулгын нэрийн дараа талбарууд байхгүй эсвэл байж болох бөгөөд зайгаар тусгаарлагдсан байна. Эдгээр талбарууд нь дээрх асуултуудад хариулдаг. Хоёр төрлийн талбар байдаг: тугны талбарууд ба утгын талбарууд. Тугны талбар нь ганцаараа байгаа түлхүүр үгнээс тогтдог, өөрөөр хэлбэл `delete` эсвэл `compress` байна. Утгын талбар нь бас түлхүүр үгээс эхлэх боловч хоосон зайгаар тусгаарлагдалгүйгээр `=` болон хоёр дахь үгээс тогтоно. Жишээ нь `release=cvs` нь утгын талбар юм.

supfile ихэвчлэн нэгээс илүү цуглуулгыг хүлээн авахаар заасан байдаг. supfile-г зохион байгуулах нэг арга нь цуглуулга бүрийн хувьд харгалзах бүх талбаруудыг зааж өгөх явдал юм. Гэхдээ энэ нь supfile-ий мөрүүдийг нэлээн урт болгодог бөгөөд supfile дахь цуглуулгуудын бүгдийн хувьд ихэнх талбарууд нь ижил байдаг болохоор таагүй юм. CVSup нь эдгээр асуудлуудыг тойрон гарах анхдагч болгох арга замтай байдаг. Тусгай псевдо цуглуулгын нэр `*default`-р эхэлсэн мөрүүд нь тугнууд болон supfile дахь дараа дараачийн цуглуулгуудын хувьд анхдагч болж ашиглагдах утгуудыг заахад хэрэглэгддэг. Анхдагч утгыг цуглуулгатай цуг өөр утгыг заан өгч тухайн цуглуулгын хувьд өөрчилж болдог. Анхдагчуудыг дундын supfile-д нэмэлт `*default` мөрүүдээр өөрчилж эсвэл дарж болдог.

Ийм мэдээлэлтэйгээр бид одоо FreeBSD-CURRENT-ийн гол эх модыг хүлээн авч шинэчлэхэд зориулсан supfile-г бүтээх болно.

- Ямар файлуудыг та хүлээн авахыг хүсэж байна вэ?

  CVSup-аар авах боломжтой файлууд нь "collections" буюу цуглуулгууд гэгддэг бүлгүүдэд хуваагдан зохион байгуулагддаг. Байгаа цуглуулгууд нь дараах хэсэгт тайлбарлагдсан байгаа. Энэ жишээн дээр бид FreeBSD-ийн хувьд бүх гол эхийг авахыг хүсэж байна. `src-all` гэсэн ганц том цуглуулга бидэнд үүнийг өгөх болно. supfile-аа бүтээх эхний алхам болгож бид цуглуулгуудыг нэг мөрөнд нэгийг тавьж жагсаана (энэ тохиолдолд зөвхөн нэг мөр байна):

  ```
  src-all
  ```

- Тэдгээрийн ямар хувилбаруудыг та хүсэж байна вэ?

CVSup-ийн тусламжтай та хэзээ ч юм байсан эхүүдийн бараг ямар ч хувилбарыг хүлээн авч чадна. cvsupd сервер нь бүх хувилбаруудыг агуулсан CVS архиваас шууд ажилладаг болохоор энэ нь боломжтой юм. Та тэдгээрийн алийг хүсэж байгаагаа `tag=` болон `date=` утга талбаруудыг ашиглан зааж өгч болно.

`tag=` талбарыг зөв зааж өгөхдөө анхааралтай байх хэрэгтэй. Зарим хаягууд нь зөвхөн файлуудын зарим цуглуулгуудын хувьд зөв байдаг. Хэрэв та буруу эсвэл буруу бичсэн хаягийг зааж өгөх юм бол магадгүй таны устгахыг хүсээгүй файлуудыг CVSup нь устгах юм. Ялангуяа `ports-*` цуглуулгуудын хувьд *зөвхөн* `tag=.`-г ашиглаарай.

`tag=` талбар нь симболын хаягийг архивт нэрлэдэг. Хоёр төрлийн хаяг байдаг бөгөөд эдгээр нь залруулалтын хаягууд болон салбарын хаягууд юм. Залруулалтын хаяг нь тухайн нэг залруулалтыг заадаг. Үүний утга нь өдрөөс өдөрт ижил хэвээр байна. Нөгөө талаас салбарын хаяг нь ямар ч үед хөгжүүлэлтийн өгөгдсөн шугам дахь хамгийн сүүлийн залруулалтыг заадаг. Салбарын хаяг нь тухайн нэг залруулалтыг заадаггүй болохоор үүний маргаашийн утга нь өнөөдрийнхөөс өөр байж болох юм.

CVS Tags буюу хаягууд нь хэрэглэгчдийн сонирхож болох салбарын хаягуудыг агуулдаг. CVSup-ийн тохиргооны файлд хаягийг зааж өгөхдөө өмнө нь `tag=` гэж өгөх ёстой (`RELENG_8` нь `tag=RELENG_8` болно). Зөвхөн `tag=.` нь портын цуглуулгад хамааралтай болохыг санаарай.

Хаягийн нэрийг яг үзүүлсэн шиг бичихдээ анхааралтай байгаарай. CVSup нь зөв болон буруу хаягуудыг ялгаж чаддаггүй. Хэрэв та хаягийг буруу бичвэл CVSup нь таныг зөв хаяг өгсөн гэж ойлгон ажиллах бөгөөд энэ нь ямар ч файлуудыг заагаагүйд хүргэж болох юм. Энэ тохиолдолд энэ нь танд байгаа эхүүдийг устгах болно.

Салбарын хаягийг заахад та тэр хөгжүүлэлтийн шугамын файлуудын хамгийн сүүлийн хувилбаруудыг авах болно. Хэрэв та ямар нэг өнгөрсөн хувилбарыг авахыг хүсвэл `date=` талбарт огноог заан хийж өгч болно. cvsup(1) гарын авлагын хуудас үүнийг хэрхэн хийхийг тайлбарладаг.

Бидний жишээний хувьд FreeBSD-CURRENT-г авахыг бид хүсэж байна. Бид энэ мөрийг өөрсдийн supfile-ийн эхэнд нэмнэ:

```
*default tag=.
```

Хэрэв та `tag=` талбар эсвэл `date=` талбарын алийг ч зааж өгөөгүй бол чухал тусгай тохиолдол гарч ирнэ. Энэ тохиолдолд тодорхой нэг хувилбарыг хүлээн авахын оронд серверийн CVS архиваас RCS файлуудыг шууд хүлээн авдаг. Ажиллагааны ийм горимыг хөгжүүлэгчид ерөнхийдөө эрхэмлэдэг. Өөрсдийн системүүд дээр архивын хуулбарыг арчилснаар тэд залруулалтын түүхийг хайж файлуудын өнгөрсөн хувилбаруудыг шалгах боломжтой болох юм. Гэхдээ энэ хонжоо нь дискний зайны хувьд их

өртөгтэйгээр орж ирдэг.

- Тэдгээрийг та хаанаас авахыг хүсэж байна вэ?

  Бид `host=` талбарыг ашиглан `cvsup`-д өөрийн шинэчлэлтүүдийг хаанаас авахыг хэлж өгдөг. CVSup толин тусгал сайтуудын аль нь ч болох боловч та өөрийн кибер ертөнцдөө ойрхныг турших хэрэгтэй. Энэ жишээн дээр бид зохиомол FreeBSD түгээлтийн сайт `cvsup99.FreeBSD.org`-г ашиглах болно:

  ```
  *default host=cvsup99.FreeBSD.org
  ```

  Та CVSup-г ашиглахаасаа өмнө үнэн хэрэг дээрээ байгаа жинхэнэ хостоор солих хэрэгтэй болно. `cvsup`-г ажиллуулах үедээ та хостын тохируулгыг тушаалын мөрөөс `-h hostname` сонголтоор өөрчлөн дарж болно.

- Та тэдгээрийг өөрийн машиныхаа хаана байрлуулахыг хүсэж байна вэ?

  `prefix=` талбар нь хүлээн авсан файлуудыг хаана байрлуулахыг `cvsup` хэлнэ. Энэ жишээн дээр бид эх файлуудыг өөрсдийн гол эх модны /usr/src байрлалд шууд байрлуулах болно. src сан нь бидний хүлээн авахаар сонгосон цуглуулгад далд байгаа болохоор энэ нь зөв юм:

  ```
  *default prefix=/usr
  ```

- `cvsup` өөрийн төлвийн файлуудыг хаана арчлах ёстой вэ?

  CVSup клиент нь "base" буюу үндсэн гэгддэг сан дахь зарим төлвийн файлуудыг арчилж байдаг. Эдгээр файлууд нь ямар шинэчлэлтүүдийг та аль хэдийн хүлээн авсныг хянаж CVSup-ийг илүү үр ашигтай ажиллахад тусалдаг. Бид стандарт үндсэн сан /var/db-г ашиглах болно:

  ```
  *default base=/var/db
  ```

  Хэрэв таны үндсэн сан байхгүй байгаа бол одоо түүнийг үүсгэх хугацаа болсон байна. Хэрэв үндсэн сан байхгүй бол `cvsup` ажиллахаас татгалзах болно.

- supfile-ийн бусад тохиргоонууд:

  supfile-д ерөнхийдөө байх ёстой бас нэг мөр байдаг:

  ```
  *default release=cvs delete use-rel-suffix compress
  ```

  `release=cvs` нь сервер гол FreeBSD CVS архиваас өөрийн мэдээллийг авах ёстойг зааж байна. Энэ нь бараг үргэлж тохиолддог, хэдийгээр бусад боломжууд байдаг боловч энэ хэлэлцүүлгийн хүрээнээс хальдаг.

`delete` нь CVSup-д файлууд устгах зөвшөөрөл өгдөг. CVSup нь таны эх модыг байнга хамгийн сүүлийн хэлбэрт байлгадаг байхын тулд та үүнийг үргэлж зааж өгөх хэрэгтэй. CVSup нь өөрийн хариуцдаг зөвхөн тэр файлуудыг устгах тал дээр их анхааралтай ажилладаг. Танд байж болох нэмэлт файлууд тэр хэвээрээ үлддэг.

`use-rel-suffix` нь ... ойлгоход төвөгтэй юм. Хэрэв та үүний тухай жинхэнээсээ мэдье гэж хүсэж байгаа бол cvsup(1) гарын авлагын хуудаснаас үзнэ үү. Хэрэв тийм биш бол ердөө заагаад л тэгээд санаа зовох хэрэггүй юм.

`compress` нь холбооны суваг дээр gzip загварын шахалтын хэрэглээг идэвхжүүлдэг. Хэрэв таны сүлжээний холболт T1 эсвэл түүнээс хурдан бол та магадгүй шахалтыг ашиглах хэрэггүй юм. Хэрэв үгүй бол энэ нь бодитойгоор тусалдаг.

- Бүгдийг нийлүүлбэл:

  Энд бидний жишээнд зориулсан бүхэл supfile байна:

  ```
  *default tag=.
  *default host=cvsup99.FreeBSD.org
  *default prefix=/usr
  *default base=/var/db
  *default release=cvs delete use-rel-suffix compress

  src-all
  ```

#### A.7.3.1. refuse файл

Дээр дурдсанаар CVSup нь *маmах аргыг* ашигладаг. Энэ нь юу гэсэн үг вэ гэхээр та CVSup сервер руу холбогдох бөгөөд энэ нь "Над дээрээс татаж авах боломжтой зүйлс энэ байна..." гэж хэлэх бөгөөд таны клиент "OK, Би энэ,энэ, энийг авна." гэж хариулна гэсэн үг юм. Анхдагч тохиргоон дээр CVSup клиент нь таны тохиргооны файлд сонгосон хаяг болон цуглуулгатай холбоотой бүх файлыг авна. Модыг хэсэгчлэн татахын тулд refuse-г ашиглана.

refuse файл нь үндсэндээ CVSup-д цуглуулгаас файл болгоныг татаж авахгүй гэдгийг хэлдэг; өөрөөр хэлбэл клиентэд серверээс зарим файлууд татаж авахаас *татгалзахыг* хэлж өгдөг. refuse файл нь base/sup/ сангаас олдож болно (эсвэл хэрэв танд ийм файл байхгүй байгаа бол түүнийг дээрх санд байрлуулах хэрэгтэй). *base* нь таны supfile-д тодорхойлогдсон байдаг; бидний тодорхойлсон *base* нь /var/db бөгөөд энэ нь анхдагчаар refuse файл нь /var/db/sup/refuse гэсэн үг юм.

refuse файл нь маш хялбар хэлбэршилттэй байдаг; энэ нь таны татаж авахыг хүсээгүй файлууд эсвэл сангуудын нэрийг агуулдаг. Жишээ нь:

```
bin/
usr.bin/
```

Удаан холболттой юм уу эсвэл өөрсдийн Интернэтийн холболтод минутаар төлдөг хэрэглэгчид үнэ цэнэтэй хугацаагаа хэмнэх болно. Учир нь тэд өөрсдийн хэзээ ч ашиглахгүй

файлаа татаж авах шаардлагагүй болох юм. CVSup-ийн refuse файлууд болон бусад сайхан боломжуудын талаар илүү мэдээллийг түүний гарын авлагын хуудаснаас үзнэ үү.

### A.7.4. CVSup-г ажиллуулах нь

Одоо та шинэчлэлт хийж үзэхэд бэлэн боллоо. Үүнийг хийх тушаалын мөр их амархан:

```
# cvsup supfile
```

энд байгаа supfile нь таны дөнгөж үүсгэсэн supfile-ийн нэр юм. Таныг X11 дээр ажиллаж байна гэж үзвэл `cvsup` нь энгийн зүйлсийг хийх зарим товчлууруудтай GUI цонхыг үзүүлэх болно. btn[go] товчлуурыг дарж ажиллахыг нь хараарай.

Та энэ жишээн дээр өөрийн жинхэнэ /usr/src модыг шинэчилж байгаа болохоор `cvsup` нь таны файлуудыг шинэчлэхэд шаардлагатай зөвшөөрлүүдтэй байхын тулд та програмыг `root` эрхээс ажиллуулах хэрэгтэй болно. Өөрийн тохиргооны файлыг дөнгөж үүсгэсэн бөгөөд урьд нь энэ програмыг хэзээ ч ашиглаж байгаагүй бол таныг бухимдуулж магадгүй юм. Өөрийн файлуудыг хөндөлгүйгээр туршилтын журмаар ажиллуулах хялбар арга байдаг. Хаа нэгтээ хоосон сан үүсгээд түүнийг тушаалын мөрөнд нэмэлт өгөгдлөөр оруулж өгнө:

```
# mkdir /var/tmp/dest
# cvsup supfile /var/tmp/dest
```

Таны заасан сан бүх файлын шинэчлэлтүүдийн хувьд очих сан болдог. CVSup нь /usr/src сан доторх таны энгийн файлуудыг шалгадаг, гэхдээ тэдгээрийн алийг ч өөрчлөх буюу устгахгүй. Файлын шинэчлэлтүүд харин /var/tmp/dest/usr/src санд хийгдэнэ. CVSup нь ингэж ажиллахдаа бас өөрийн үндсэн сангийн төлвийн файлуудыг өөрчлөлгүйгээр үлдээдэг. Тэдгээр файлуудын шинэ хувилбарууд заагдсан сан уруу бичигдэх болно. Танд /usr/src сан уруу унших эрх л байхад иймэрхүү туршилтын ажиллагааг гүйцэтгэхэд `root` эрх байх заавал шаардлагагүй юм.

Хэрэв та X11-г ажиллуулахгүй байгаа юм уу эсвэл танд GUI таалагддаггүй бол `cvsup`-г ажиллуулахдаа тушаалын мөрөн дээр хоёр сонголтыг нэмж өгөх хэрэгтэй:

```
# cvsup -g -L 2 supfile
```

`-g` сонголт нь CVSup-д өөрийн GUI-г ашиглахгүйг хэлнэ. Хэрэв та X11-ийг ажиллуулахгүй байгаа бол энэ автомат байдаг, үгүй бол та үүнийг зааж өгөх хэрэгтэй.

`-L 2` нь CVSup-д хийж байгаа бүх шинэчлэлтүүдийнхээ талаар дэлгэрэнгүй мэдээллийг үзүүлэхийг хэлж өгнө. Гурван түвшний харуулах горим байдаг бөгөөд энэ нь `-L 0`-с `-L 2` хүртэл байна. Анхдагч нь 0 байх бөгөөд энэ нь алдааны мэдэгдлээс бусдыг харуулахгүй.

Бусад олон сонголтууд байдаг. Тэдгээрийн товч жагсаалтыг үзэхийн тулд `cvsup -H` гэж бичнэ. Илүү дэлгэрэнгүй тайлбаруудын талаар гарын авлагын хуудсыг үзнэ үү.

Шинэчлэлт ажилладаг аргад сэтгэл ханамжтай болсныхоо дараа та CVSup-г давтамжтайгаар ажиллуулахын тулд cron(8) ашиглан хийж өгч болно. Мэдээж та cron(8)-с CVSup-г ажиллуулахдаа өөрийнхөө GUI-г ашиглахгүй болгох хэрэгтэй.

### A.7.5. CVSup-ийн файлын цуглуулгууд

CVSup-ийн файлын цуглуулгууд нь шатлалтайгаар зохион байгуулагдсан байдаг. Цөөн том цуглуулгууд байдаг бөгөөд тэдгээр нь арай жижиг дэд цуглуулгуудад хуваагддаг. Том цуглуулгыг хүлээн авах нь түүний дэд цуглуулгууд тус бүрийг хүлээн авахтай адил юм. Цуглуулга хоорондын шаталсан холбоонууд нь доорх жагсаалт дахь догол ашиглалтаар тусгагддаг.

Хамгийн түгээмэл хэрэглэгддэг цуглуулгууд нь `src-all` юм.

`cvs-all release=cvs`

Криптограф кодыг оролцуулаад FreeBSD-ийн гол CVS архив.

`distrib release=cvs`

FreeBSD-ийн түгээлт болон толин тусгал хийхтэй холбоотой файлууд.

`projects-all release=cvs`

FreeBSD төслийн архив дахь эхүүд.

`src-all release=cvs`

Криптограф кодыг оролцуулаад FreeBSD-ийн гол эхүүд.

`src-base release=cvs`

/usr/src сангийн дээр байрлах бусад файлууд.

`src-bin release=cvs`

Ганц хэрэглэгчийн горимд хэрэг болох хэрэглэгчийн багажууд (/usr/src/bin).

`src-cddl release=cvs`

CDDL лицензийн доор байдаг хэрэгслүүд болон сангууд (/usr/src/cddl).

`src-contrib release=cvs`

Харьцангуй өөрчлөлтгүйгээр хэрэглэгддэг FreeBSD төслөөс гаднах хэрэгслүүд ба сангууд (/usr/src/contrib).

`src-crypto release=cvs`

Харьцангуй өөрчлөлтгүйгээр хэрэглэгддэг FreeBSD төслөөс гаднах криптограф хэрэгслүүд ба сангууд (/usr/src/crypto).

`src-eBones release=cvs`

Kerberos ба DES (/usr/src/eBones). FreeBSD-ийн одоогийн хувилбаруудад ашигладаггүй.

`src-etc release=cvs`

Системийн тохиргооны файлууд (/usr/src/etc).

`src-games release=cvs`

Тоглоомууд (/usr/src/games).

`src-gnu release=cvs`

GNU Нийтийн Лицензтэй хэрэгслүүд (/usr/src/gnu).

`src-include release=cvs`

Толгой файлууд (/usr/src/include).

`src-kerberos5 release=cvs`

Kerberos5 аюулгүй байдлын багц (/usr/src/kerberos5).

`src-kerberosIV release=cvs`

KerberosIV аюулгүй байдлын багц (/usr/src/kerberosIV).

`src-lib release=cvs`

Сангууд (/usr/src/lib).

`src-libexec release=cvs`

Бусад програмуудаар ажилладаг системийн програмууд (/usr/src/libexec).

`src-release release=cvs`

FreeBSD хувилбар гаргахад шаардагдах файлууд (/usr/src/release).

`src-rescue release=cvs`

Яаралтай сэргээлт хийхэд зориулсан статикаар холболт хийгдсэн програмууд; rescue(8)-г үзнэ үү (/usr/src/rescue).

`src-sbin release=cvs`

Ганц хэрэглэгчийн горимд зориулсан системийн хэрэгслүүд (/usr/src/sbin).

`src-secure release=cvs`

Криптограф сангууд ба тушаалууд (/usr/src/secure).

`src-share release=cvs`

Олон системүүдийн хооронд хуваалцаж болох файлууд (/usr/src/share).

`src-sys release=cvs`

Цөм (/usr/src/sys).

`src-sys-crypto release=cvs`

Цөмийн криптограф код (/usr/src/sys/crypto).

`src-tools release=cvs`

FreeBSD-г арчлахад зориулсан төрөл бүрийн хэрэгслүүд (/usr/src/tools).

`src-usrbin release=cvs`

Хэрэглэгчийн хэрэгслүүд (/usr/src/usr.bin).

`src-usrsbin release=cvs`

Системийн хэрэгслүүд (/usr/src/usr.sbin).

`distrib release=self`

CVSup серверийн өөрийн тохиргооны файлууд. CVSup толин тусгал сайтуудад хэрэглэгддэг.

`gnats release=current`

GNATS алдаа мөрдөх мэдээллийн бааз.

`mail-archive release=current`

FreeBSD захидлын жагсаалтын архив.

### A.7.6. Дэлгэрэнгүй мэдээллийг

CVSup FAQ болон бусад CVSup-ийн тухай мэдээллийг CVSup гэрийн хуудас хаягаас үзнэ үү.

Ихэнх FreeBSD-тэй холбоотой CVSup-ийн хэлэлцүүлэг FreeBSD техникийн хэлэлцүүлгүүдийн захидлын жагсаалт-д болдог. Програмын шинэ хувилбар тэнд, бас FreeBSD зарлалын захидлын жагсаалт-д зарлагддаг.

CVSup-ийн талаар асуултууд эсвэл алдааны тайлангуудыг CVSup FAQ холбоосоос үзнэ үү.

### A.7.7. CVSup сайтууд

FreeBSD-д зориулсан CVSup серверүүд дараах хаягууд дээр ажиллаж байгаа:

## A.8. CVS Tags буюу хаягууд

CVS нь төслийн хувьд хуучирсан бөгөөд ашиглахыг зөвлөдөггүй. Түүний оронд Subversion-г ашиглах ёстой.

cvs эсвэл CVSup ашиглан эхийг авах юм уу эсвэл шинэчилж байгаа үед revision tag буюу залруулалтын хаягийг зааж өгөх ёстой байдаг. Залруулалтын хаяг нь FreeBSD хөгжүүлэлтийн тодорхой нэг байх юм уу эсвэл хугацааны тодорхой нэг цэг байдаг. Эхнийх нь "branch tags" буюу "салбарын хаягууд" гэгддэг бөгөөд хоёр дахь нь "хувилбарын хаягууд" гэгддэг.

### A.8.1. Салбарын хаягууд

HEAD-с (энэ нь үргэлж зөв хаяг байдаг) бусад бүх хаягууд зөвхөн src/ модонд хамаардаг. ports/, doc/, болон www/ моднууд нь салбарладаггүй.

**HEAD**

Гол шугамын симболын нэр, эсвэл FreeBSD-CURRENT. Залруулалт заагаагүй тохиолдолд

бас анхдагч байдаг.

CVSup-д энэ хаяг нь `.` гэсэн тэмдэгтээр илэрхийлэгддэг (цэг биш харин `.` тэмдэгт).

CVS-д энэ нь залруулалт заагаагүй тохиолдолд анхдагч байдаг. Хэрэв та өөрөө хүсээгүй л бол STABLE машин дээр CURRENT эхийг татан авч шинэчлэх нь ихэвчлэн тийм ч сайн санаа *биш* юм.

**RELENG_9**

FreeBSD-9.X-д зориулсан хөгжүүлэлтийн шугам, бас FreeBSD 8-STABLE гэгддэг

**RELENG_9_0**

FreeBSD-9.0-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_8**

FreeBSD-8.X-д зориулсан хөгжүүлэлтийн шугам, бас FreeBSD 8-STABLE гэгддэг

**RELENG_8_3**

FreeBSD-8.3-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_8_2**

FreeBSD-8.2-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_8_1**

FreeBSD-8.1-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_8_0**

FreeBSD-8.0-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_7**

FreeBSD-7.X-д зориулсан хөгжүүлэлтийн шугам, бас FreeBSD 7-STABLE гэгддэг

**RELENG_7_4**

FreeBSD-7.4-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_7_3**

FreeBSD-7.3-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_7_2**

FreeBSD-7.2-т зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_7_1**

FreeBSD-7.1-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_7_0**

FreeBSD-7.0-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_6**

FreeBSD-6.X-д зориулсан хөгжүүлэлтийн шугам, бас FreeBSD 6-STABLE гэгддэг

**RELENG_6_4**

FreeBSD-6.4-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_6_3**

FreeBSD-6.3-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_6_2**

FreeBSD-6.2-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_6_1**

FreeBSD-6.1-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_6_0**

FreeBSD-6.0-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_5**

FreeBSD-5.X-д зориулсан хөгжүүлэлтийн шугам, бас FreeBSD 5-STABLE гэгддэг.

**RELENG_5_5**

FreeBSD-5.5-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_5_4**

FreeBSD-5.4-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_5_3**

FreeBSD-5.3-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_5_2**

FreeBSD-5.2 болон FreeBSD-5.2.1-д зориулсан хувилбарын салбар, аюулгүй байдлын

зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_5_1**

FreeBSD-5.1-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_5_0**

FreeBSD-5.0-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4**

FreeBSD-4.X-д зориулсан хөгжүүлэлтийн шугам, бас FreeBSD 4-STABLE гэгддэг.

**RELENG_4_11**

FreeBSD-4.11-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_10**

FreeBSD-4.10-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_9**

FreeBSD-4.9-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_8**

FreeBSD-4.8-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_7**

FreeBSD-4.7-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_6**

FreeBSD-4.6 болон FreeBSD-4.6.2-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_5**

FreeBSD-4.5-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_4**

FreeBSD-4.4-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_4_3**

FreeBSD-4.3-д зориулсан хувилбарын салбар, аюулгүй байдлын зөвлөгөөнүүд болон бусад чухал засваруудад зориулж зөвхөн ашиглагддаг.

**RELENG_3**

FreeBSD-3.X-д зориулсан хөгжүүлэлтийн шугам, бас 3.X-STABLE гэгддэг.

**RELENG_2_2**

FreeBSD-2.2.X-д зориулсан хөгжүүлэлтийн шугам, бас 2.2-STABLE гэгддэг. Энэ салбар нь гол төлөв хуучирсан.

## A.8.2. Хувилбарын хаягууд

Эдгээр хаягууд нь FreeBSD-ийн тодорхой нэг хувилбар гарсан үеийн цагийн тодорхой цэгийг заадаг. Хувилбар инженерчлэлийн процессийн талаар Хувилбар инженерчлэлийн мэдээлэл болон Хувилбарын процесс баримтуудад илүү дэлгэрэнгүй баримтжуулагдсан байдаг. src мод нь `RELENG_` гэж эхэлсэн хаягийн нэрсийг ашигладаг. ports болон doc моднууд нь `RELEASE` гэж эхэлсэн хаягийн нэрсийг ашигладаг. Төгсгөлд нь www мод нь хувилбаруудад зориулсан ямар нэг тусгай нэрээр хаяглагддаггүй.

**RELENG_9_0_0_RELEASE**

FreeBSD 9.0

**RELENG_8_3_0_RELEASE**

FreeBSD 8.3

**RELENG_8_2_0_RELEASE**

FreeBSD 8.2

**RELENG_8_1_0_RELEASE**

FreeBSD 8.1

**RELENG_8_0_0_RELEASE**

FreeBSD 8.0

**RELENG_7_4_0_RELEASE**

FreeBSD 7.4

**RELENG_7_3_0_RELEASE**

FreeBSD 7.3

**RELENG_7_2_0_RELEASE**

FreeBSD 7.2

**RELENG_7_1_0_RELEASE**

FreeBSD 7.1

**RELENG_7_0_0_RELEASE**

FreeBSD 7.0

**RELENG_6_4_0_RELEASE**

FreeBSD 6.4

**RELENG_6_3_0_RELEASE**
FreeBSD 6.3

**RELENG_6_2_0_RELEASE**
FreeBSD 6.2

**RELENG_6_1_0_RELEASE**
FreeBSD 6.1

**RELENG_6_0_0_RELEASE**
FreeBSD 6.0

**RELENG_5_5_0_RELEASE**
FreeBSD 5.5

**RELENG_5_4_0_RELEASE**
FreeBSD 5.4

**RELENG_4_11_0_RELEASE**
FreeBSD 4.11

**RELENG_5_3_0_RELEASE**
FreeBSD 5.3

**RELENG_4_10_0_RELEASE**
FreeBSD 4.10

**RELENG_5_2_1_RELEASE**
FreeBSD 5.2.1

**RELENG_5_2_0_RELEASE**
FreeBSD 5.2

**RELENG_4_9_0_RELEASE**
FreeBSD 4.9

**RELENG_5_1_0_RELEASE**
FreeBSD 5.1

**RELENG_4_8_0_RELEASE**
FreeBSD 4.8

**RELENG_5_0_0_RELEASE**
FreeBSD 5.0

**RELENG_4_7_0_RELEASE**
FreeBSD 4.7

**RELENG_4_6_2_RELEASE**
FreeBSD 4.6.2

**RELENG_4_6_1_RELEASE**
FreeBSD 4.6.1

**RELENG_4_6_0_RELEASE**
FreeBSD 4.6

**RELENG_4_5_0_RELEASE**
FreeBSD 4.5

**RELENG_4_4_0_RELEASE**
FreeBSD 4.4

**RELENG_4_3_0_RELEASE**
FreeBSD 4.3

**RELENG_4_2_0_RELEASE**
FreeBSD 4.2

**RELENG_4_1_1_RELEASE**
FreeBSD 4.1.1

**RELENG_4_1_0_RELEASE**
FreeBSD 4.1

**RELENG_4_0_0_RELEASE**
FreeBSD 4.0

**RELENG_3_5_0_RELEASE**
FreeBSD-3.5

**RELENG_3_4_0_RELEASE**
FreeBSD-3.4

**RELENG_3_3_0_RELEASE**
FreeBSD-3.3

**RELENG_3_2_0_RELEASE**
FreeBSD-3.2

**RELENG_3_1_0_RELEASE**
FreeBSD-3.1

**RELENG_3_0_0_RELEASE**
FreeBSD-3.0

**RELENG_2_2_8_RELEASE**

FreeBSD-2.2.8

**RELENG_2_2_7_RELEASE**

FreeBSD-2.2.7

**RELENG_2_2_6_RELEASE**

FreeBSD-2.2.6

**RELENG_2_2_5_RELEASE**

FreeBSD-2.2.5

**RELENG_2_2_2_RELEASE**

FreeBSD-2.2.2

**RELENG_2_2_1_RELEASE**

FreeBSD-2.2.1

**RELENG_2_2_0_RELEASE**

FreeBSD-2.2.0

# A.9. rsync сайтууд

Дараах сайтууд нь FreeBSD-г rsync протоколоор түгээгдэх боломжийг бүрдүүлдэг. rsync хэрэгсэл нь rcp(1) тушаалтай бараг төстэйгөөр ажилладаг боловч илүү олон тохируулгуудтай бөгөөд хоёр талын файлуудын зөвхөн ялгаатайг нь дамжуулдаг. Ингэснээр сүлжээгээр хийх хамгийн сүүлийн хэлбэрт аваачих үйлдлийг ихээхэн хурдасгадаг байна. Хэрэв та FreeBSD FTP сервер юм уу эсвэл CVS архивын толин тусгал сайт бол энэ нь их ашигтай байдаг. rsync цуглуулга нь олон үйлдлийн системүүд дээр байдаг. FreeBSD-ийн хувьд net/rsync порт эсвэл багцыг үзнэ үү.

**Бүгд Найрамдах Чех Улс**

rsync://ftp.cz.FreeBSD.org/

Байгаа цуглуулгууд:

- ftp: FreeBSD FTP серверийн хэсэгчилсэн толин тусгал.
- FreeBSD: FreeBSD FTP серверийн бүрэн толин тусгал.

**Нидерланд**

rsync://ftp.nl.FreeBSD.org/

Байгаа цуглуулгууд:

- FreeBSD: FreeBSD FTP серверийн бүрэн толин тусгал.

**Орос**

rsync://ftp.mtu.ru/

Байгаа цуглуулгууд:

- FreeBSD: FreeBSD FTP серверийн бүрэн толин тусгал.
- FreeBSD-gnats: GNATS цох хянах мэдээллийн сан.
- FreeBSD-Archive: FreeBSD архив FTP серверийн бүрэн толин тусгал.

**Швед**

rsync://ftp4.se.freebsd.org/

Байгаа цуглуулгууд:

- FreeBSD: FreeBSD FTP серверийн бүрэн толин тусгал.

**Тайвань**

rsync://ftp.tw.FreeBSD.org/

rsync://ftp2.tw.FreeBSD.org/

rsync://ftp6.tw.FreeBSD.org/

Байгаа цуглуулгууд:

- FreeBSD: FreeBSD FTP серверийн бүрэн толин тусгал.

**Их Британи**

rsync://rsync.mirrorservice.org/

Байгаа цуглуулгууд:

- ftp.freebsd.org: FreeBSD FTP серверийн бүрэн толин тусгал.

**Америкийн Нэгдсэн Улс**

rsync://ftp-master.FreeBSD.org/

Энэ серверийг FreeBSD-ийн анхдагч толин тусгал сайтууд зөвхөн хэрэглэж болно.

Байгаа цуглуулгууд:

- FreeBSD: FreeBSD FTP серверийн мастер архив.
- acl: FreeBSD-ийн мастер ACL жагсаалт.

  rsync://ftp13.FreeBSD.org/

  Байгаа цуглуулгууд:

- FreeBSD: FreeBSD FTP серверийн бүрэн толин тусгал.

# Appendix B: Ном зүй

Гарын авлагын хуудаснууд нь FreeBSD үйлдлийн системийн хувь хэсгүүдэд зориулсан эцсийн лавлагааг өгдөг боловч тэдгээр хэсгүүдийг хэрхэн нийлүүлж бүхэл бүтэн үйлдлийн системийг тэгш ажиллуулах талаар дүрслэн үзүүлдэггүйгээрээ зартай юм. Ийм учраас UNIX®, системийн удирдлагын талаар бичсэн сайн ном болон хэрэглэгчдийн сайн гарын авлагыг орлохоор зүйлс байхгүй юм.

## B.1. FreeBSD-ийн талаар тусгайлан бичсэн ном & сэтгүүлүүд

*Олон улсад хэвлэгдсэн ном & сэтгүүлүүд:*

- Using FreeBSD (Уламжлалт Хятад хэл дээр), хэвлэсэн Drmaster, 1997. ISBN 9-578-39435-7.
- FreeBSD Unleashed (Хялбаршуулсан Хятад орчуулга), хэвлэсэн China Machine Press. ISBN 7-111-10201-0.
- FreeBSD From Scratch Хоёр дахь хэвлэл (Хялбаршуулсан Хятад хэл дээр), хэвлэсэн China Machine Press. ISBN 7-111-10286-X.
- FreeBSD Handbook Хоёр дахь хэвлэл (Хялбаршуулсан Хятад орчуулга), хэвлэсэн Posts & Telecom Press. ISBN 7-115-10541-3.
- FreeBSD & Windows (Хялбаршуулсан Хятад хэл дээр), хэвлэсэн China Railway Publishing House. ISBN 7-113-03845-X
- FreeBSD Internet Services HOWTO (Хялбаршуулсан Хятад хэл дээр), хэвлэсэн China Railway Publishing House. ISBN 7-113-03423-3
- FreeBSD (Япон хэл дээр), хэвлэсэн CUTT. ISBN 4-906391-22-2 C3055 P2400E.
- Complete Introduction to FreeBSD (Япон хэл дээр), хэвлэсэн Shoeisha Co., Ltd. ISBN 4-88135-473-6 P3600E.
- Personal UNIX Starter Kit FreeBSD (Япон хэл дээр), хэвлэсэн ASCII. ISBN 4-7561-1733-3 P3000E.
- FreeBSD Handbook (Япон орчуулга), хэвлэсэн ASCII. ISBN 4-7561-1580-2 P3800E.
- FreeBSD mit Methode (Герман хэл дээр), хэвлэсэн Computer und Literatur Verlag/Vertrieb Hanser, 1998. ISBN 3-932311-31-0.
- FreeBSD de Luxe (Герман хэл дээр), хэвлэсэн Verlag Modere Industrie, 2003. ISBN 3-8266-1343-0.
- FreeBSD Install and Utilization Manual (Япон хэл дээр), хэвлэсэн Mainichi Communications Inc, 1998. ISBN 4-8399-0112-0..
- Onno W Purbo, Dodi Maryanto, Syahrial Hubbany, Widjil Widodo *Building Internet Server with FreeBSD* (Индонез хэл дээр), published by Elex Media Komputindo.
- Absolute BSD: The Ultimate Guide to FreeBSD (Уламжлалт Хятад орчуулга), хэвлэсэн GrandTech Press, 2003. ISBN 986-7944-92-5.
- The FreeBSD 6.0 Book (Уламжлалт Хятад хэл дээр), хэвлэсэн Drmaster, 2006. ISBN 9-575-27878-X.

*Англи хэл дээрх ном & сэтгүүлүүд:*


- Absolute FreeBSD, 2nd Edition: The Complete Guide to FreeBSD, хэвлэсэн No Starch Press, 2007. ISBN: 978-1-59327-151-0
- The Complete FreeBSD, хэвлэсэн O'Reilly, 2003. ISBN: 0596005164
- The FreeBSD Corporate Networker's Guide, хэвлэсэн Addison-Wesley, 2000. ISBN: 0201704811
- FreeBSD: An Open-Source Operating System for Your Personal Computer, хэвлэсэн The Bit Tree Press, 2001. ISBN: 0971204500
- Teach Yourself FreeBSD in 24 Hours, хэвлэсэн Sams, 2002. ISBN: 0672324245
- FreeBSD 6 Unleashed, хэвлэсэн Sams, 2006. ISBN: 0672328755
- FreeBSD: The Complete Reference, хэвлэсэн McGrawHill, 2003. ISBN: 0072224096
- BSD magazine, хэвлэсэн Software Press Sp. z o.o. SK. ISSN 1898-9144


## B.2. Хэрэглэгчдийн гарын авлагууд

- Охайогийн Их Сургууль UNIX Introductory Course авлагыг бичсэн бөгөөд энэ нь шууд вэбээр үзэх боломжтойгоор HTML болон PostScript хэлбэрээр байдаг.

  Энэ баримтын Итал орчуулга Италийн FreeBSD-ийн Баримтжуулах Төслийн хэсэг болон авах боломжтойгоор байдаг.

- Jpman Төсөл, Японы FreeBSD Хэрэглэгчдийн Бүлэг. FreeBSD User's Reference Manual (Япон орчуулга). Mainichi Communications Inc., 1998. ISBN4-8399-0088-4 P3800E.
- Эдинбургийн Их Сургууль UNIX орчинд шинээр ирж байгаа хүмүүст зориулж Онлайн заавар бичсэн.

## B.3. Администраторууд зориулсан зааврууд

- Jpman Төсөл, Японы FreeBSD Хэрэглэгчдийн Бүлэг. FreeBSD System Administrator's Manual (Япон орчуулга). Mainichi Communications Inc., 1998. ISBN4-8399-0109-0 P3300E.
- Dreyfus, Emmanuel. Cahiers de l'Admin: BSD 2nd Ed. (Франц хэл дээр), Eyrolles, 2004. ISBN 2-212-11463-X

## B.4. Програм зохиогчдод зориулсан зааврууд


- Computer Systems Research Group, UC Berkeley. *4.4BSD Programmer's Reference Manual*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-078-3
- Computer Systems Research Group, UC Berkeley. *4.4BSD Programmer's Supplementary Documents*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-079-1
- Harbison, Samuel P. and Steele, Guy L. Jr. *C: A Reference Manual*. 4th ed. Prentice Hall, 1995. ISBN 0-13-326224-3
- Kernighan, Brian and Dennis M. Ritchie. *The C Programming Language*. 2nd Ed. PTR Prentice Hall, 1988. ISBN 0-13-110362-8

- Lehey, Greg. *Porting UNIX Software*. O'Reilly & Associates, Inc., 1995. ISBN 1-56592-126-7
- Plauger, P. J. *The Standard C Library*. Prentice Hall, 1992. ISBN 0-13-131509-9
- Spinellis, Diomidis. Code Reading: The Open Source Perspective. Addison-Wesley, 2003. ISBN 0-201-79940-5
- Spinellis, Diomidis. Code Quality: The Open Source Perspective. Addison-Wesley, 2006. ISBN 0-321-16607-8
- Stevens, W. Richard and Stephen A. Rago. *Advanced Programming in the UNIX Environment*. 2nd Ed. Reading, Mass. : Addison-Wesley, 2005. ISBN 0-201-43307-9
- Stevens, W. Richard. *UNIX Network Programming*. 2nd Ed, PTR Prentice Hall, 1998. ISBN 0-13-490012-X


## B.5. Үйлдлийн системийн дотоод бүрэлдэхүүнүүдийн талаар


- Andleigh, Prabhat K. *UNIX System Architecture*. Prentice-Hall, Inc., 1990. ISBN 0-13-949843-5
- Jolitz, William. "Porting UNIX to the 386". *Dr. Dobb's Journal*. January 1991-July 1992.
- Leffler, Samuel J., Marshall Kirk McKusick, Michael J Karels and John Quarterman *The Design and Implementation of the 4.3BSD UNIX Operating System*. Reading, Mass. : Addison-Wesley, 1989. ISBN 0-201-06196-1
- Leffler, Samuel J., Marshall Kirk McKusick, *The Design and Implementation of the 4.3BSD UNIX Operating System: Answer Book*. Reading, Mass. : Addison-Wesley, 1991. ISBN 0-201-54629-9
- McKusick, Marshall Kirk, Keith Bostic, Michael J Karels, and John Quarterman. *The Design and Implementation of the 4.4BSD Operating System*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-54979-4

  (Энэ номын 2-р бүлэг онлайнаар FreeBSD баримтжуулах төслийн хүрээнд байгаа.)

- Marshall Kirk McKusick, George V. Neville-Neil *The Design and Implementation of the FreeBSD Operating System*. Boston, Mass. : Addison-Wesley, 2004. ISBN 0-201-70245-2
- Stevens, W. Richard. *TCP/IP Illustrated, Volume 1: The Protocols*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-63346-9
- Schimmel, Curt. *Unix Systems for Modern Architectures*. Reading, Mass. : Addison-Wesley, 1994. ISBN 0-201-63338-8
- Stevens, W. Richard. *TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP and the UNIX Domain Protocols*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-63495-3
- Vahalia, Uresh. *UNIX Internals — The New Frontiers*. Prentice Hall, 1996. ISBN 0-13-101908-2
- Wright, Gary R. and W. Richard Stevens. *TCP/IP Illustrated, Volume 2: The Implementation*. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-63354-X

## B.6. Аюулгүй байдлын тухай номнууд


- Cheswick, William R. and Steven M. Bellovin. *Firewalls and Internet Security: Repelling the Wily Hacker*. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-63357-4
- Garfinkel, Simson. *PGP Pretty Good Privacy* O'Reilly & Associates, Inc., 1995. ISBN 1-56592-098-8


## B.7. Тоног төхөөрөмжийн тухай номнууд


- Anderson, Don and Tom Shanley. *Pentium Processor System Architecture*. 2nd Ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40992-5
- Ferraro, Richard F. *Programmer's Guide to the EGA, VGA, and Super VGA Cards*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-62490-7
- Интел корпораци өөрсдийн CPU-ууд, бичил схемүүд болон стандартуудын тухай баримт бичгүүдийг хөгжүүлэгчийн вэб хуудсанд ихэвчлэн PDF файл хэлбэрээр байрлуулдаг.
- Shanley, Tom. *80486 System Architecture*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40994-1
- Shanley, Tom. *ISA System Architecture*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40996-8
- Shanley, Tom. *PCI System Architecture*. 4th ed. Reading, Mass. : Addison-Wesley, 1999. ISBN 0-201-30974-2
- Van Gilluwe, Frank. *The Undocumented PC*, 2nd Ed. Reading, Mass: Addison-Wesley Pub. Co., 1996. ISBN 0-201-47950-8
- Messmer, Hans-Peter. *The Indispensable PC Hardware Book*, 4th Ed. Reading, Mass: Addison-Wesley Pub. Co., 2002. ISBN 0-201-59616-4


## B.8. UNIX®-ийн түүх


- Lion, John *Lion's Commentary on UNIX, 6th Ed. With Source Code*. ITP Media Group, 1996. ISBN 1573980137
- Raymond, Eric S. *The New Hacker's Dictionary, 3rd edition*. MIT Press, 1996. ISBN 0-262-68092-0. Жаргон Файл гэж бас хэлэгддэг.
- Salus, Peter H. *A quarter century of UNIX*. Addison-Wesley Publishing Company, Inc., 1994. ISBN 0-201-54777-5
- Simon Garfinkel, Daniel Weise, Steven Strassmann. *The UNIX-HATERS Handbook*. IDG Books Worldwide, Inc., 1994. ISBN 1-56884-203-1. Хэвлэгдэхээ больсон, гэхдээ онлайн байгаа.
- Don Libes, Sandy Ressler *Life with UNIX* - special edition. Prentice-Hall, Inc., 1989. ISBN 0-13-536657-7
- *BSD-н үргийн бичиг*. FreeBSD машин дээрх http://www.FreeBSD.org/cgi/cvsweb.cgi/src/shared/misc/bsd-family-tree эсвэл /usr/shared/misc/bsd-family-tree
- *Networked Computer Science Technical Reports Library*. http://www.ncstrl.org/

- *Computer Systems Research group (CSRG) буюу Компьютерийн Системүүдийн Судалгааны Бүлгийн гаргасан хуучин BSD хувилбарууд*. http://www.mckusick.com/csrg/: 4 CD бүхий цуглуулга нь 1BSD-ээс авахуулаад 4.4BSD болон 4.4BSD-Lite2 (харамсалтай нь 2.11BSD-г биш) хүртэлх бүх BSD хувилбаруудыг хамардаг. Сүүлийн диск нь төгсгөлийн эхүүдээс гадна SCCS файлуудыг бас агуулдаг.

## B.9. Сонин, сэтгүүлүүд

- *The C/C++ Users Journal*. R&D Publications Inc. ISSN 1075-2838
- *Sys Admin - The Journal for UNIX System Administrators* Miller Freeman, Inc., ISSN 1061-2688
- *freeX - Das Magazin für Linux - BSD - UNIX* (Герман хэл дээр) Computer- und Literaturverlag GmbH, ISSN 1436-7033

# Appendix C: Интернэт дэх эх үүсвэрүүд

FreeBSD-ийн дэвшлийн хурдан алхаа нь хамгийн сүүлийн хөгжүүлэлтүүдийг дагахаас хамаараад хэвлэн гаргах хэрэгцээгүй болгоход хүргэдэг. Цахим эх үүсвэрүүд нь хамгийн шилдэг нь бөгөөд хамгийн сүүлийн үеийн дэвшилтүүдийн талаар мэдлэгтэй байх бараг цорын ганц арга юм. FreeBSD нь сайн дурын чармайлт болохоор хэрэглэгчийн хүрээнийхэн нь өөрсдөө цахим захидал, вэб хэлэлцүүлгүүд болон тэр хүрээнийхэнд хүрэх хамгийн үр ашигтай арга болох USENET мэдээнүүдийн тусламжтайгаар ерөнхийдөө нэгэн хэлбэрийн "техникийн дэмжлэгийн хэлтэс" маягаар үйлчилдэг.

FreeBSD-ийн хэрэглэгчийн хүрээнийхэнд хандах хамгийн чухал цэгүүдийг доор дурдав. Хэрэв та энд дурдагдсанаас өөр эх үүсвэрүүдийн талаар мэдэж байвал тэдгээрийг бас энэ жагсаалтад оруулахаар FreeBSD баримтжуулах төслийн захидлын жагсаалт уруу илгээнэ үү.

## C.1. Захидлын жагсаалтууд

Захидлын жагсаалтууд нь FreeBSD-ийн хэрэглэгчдэд чиглэсэн асуултуудаа тавих эсвэл техникийн хэлэлцүүлэг нээх хамгийн шулуун зам юм. FreeBSD-ийн төрөл бүрийн сэдвүүдээр олон янзын жагсаалт байдаг. Өөрийн асуултуудаа хамгийн тохиромжтой захидлын жагсаалт руу хаягласнаар илүү хурдан, зөв хариу авах боломжтой болгодог.

Энэ баримтын сүүл хэсэгт төрөл бүрийн жагсаалтуудад зориулсан дүрмүүд бий. *Аль нэг жагсаалт уруу элсэн орох эсвэл захидал илгээхээсээ өмнө жагсаалтын дүрмийг уншина уу*. Одоо манай захидлын жагсаалтын захиалагчдын ихэнх нь FreeBSD-тэй холбоотой олон зуун захиануудыг өдөр бүр авдаг бөгөөд зөв хэрэглэх тухай дүрэм, журмуудыг гаргаснаар бид жагсаалтуудын шуугиан, дохионы харьцааг (signal-to-noise ratio) өндөр байлгахыг хичээдэг. Бага хийх нь захидлын жагсаалтыг энэ төсөлд зориулсан үр ашигтай холбооны зөөвөрлөгч байхыг эцсийн эцэст болиулж болох юм.

*Хэрэв та FreeBSD-ийн жагсаалтууд руу илгээж чадаж байгаа эсэхээ шалгаж тест хийхийг хүсвэл FreeBSD тест захидлын жагсаалт руу тест захидал илгээнэ үү.* Өөр бусад жагсаалт руу тест захидал битгий илгээгээрэй.

Аль жагсаалт уруу асуултаа илгээхээ мэдэхгүй эргэлзэж байгаа бол FreeBSD-асуултууд захидлын жагсаалтаас хэрхэн хамгийн их ашгийг авч болох вэ хэсгээс үзнэ үү.

Аль нэг жагсаалт уруу илгээхээсээ өмнө байнга давтагддаг хэлэлцүүлгүүдийг үүсгэхгүй байхад хэрхэн туслах зэрэг захидлын жагсаалтыг хэрхэн хамгийн үр ашигтай ашиглах талаар сурах хэрэгтэй. Үүнийг тулд Захидлын жагсаалтын байнга асуудаг асуултууд (FAQ) баримтыг унших нь зүйтэй.

Бүх захидлын жагсаалтуудын хувьд архивууд хадгалагддаг бөгөөд FreeBSD Дэлхий Даяарх Вэб сервер ашиглан хайлт хийж болно. Түлхүүр үгээр хайж болох архив нь байнга асуугддаг асуултуудад зориулсан хариултуудыг олох хамгийн шилдэг аргыг санал болгодог бөгөөд асуулт илгээхээсээ өмнө эдгээрээс лавлах хэрэгтэй юм. Энэ нь бас FreeBSD-ийн захидлын жагсаалт руу илгээсэн захиануудыг үргэлж хадгалж байдаг гэсэн үг юм. Хувийн нууцыг хамгаалах шаардлагатай бол устгах боломжтой хоёр дахь захианы хаяг ашиглаж зөвхөн нийтийн мэдээллийг явуулах хэрэгтэй.

### C.1.1. Жагсаалтын тойм

*Ерөнхий жагсаалтууд:* Дараах нь ерөнхий жагсаалтууд бөгөөд эдгээрт хэн ч гэсэн чөлөөтэй (зоригтойгоор) элсэж болно:

| Жагсаалт | Зорилго |
| --- | --- |
| freebsd-advocacy | FreeBSD Evangelism |
| FreeBSD зарлалын захидлын жагсаалт | Чухал үйл явдлууд болон төслийн гол гол үе шатууд (зохицуулалттай) |
| freebsd-arch | Архитектур болон дизайны хэлэлцүүлгүүд |
| freebsd-bugbusters | FreeBSD-ийн асуудлын тайлангийн мэдээллийн бааз болон түүнтэй холбоотой хэрэгслүүдийн арчилгаатай холбоотой хэлэлцүүлгүүд |
| freebsd-bugs | Алдааны тайлангууд |
| freebsd-chat | FreeBSD хүрээнийхэнтэй холбоотой техникийн бус зүйлс |
| freebsd-chromium | FreeBSD-тэй холбоотой Chromium-ийн хэлэлцүүлэг |
| FreeBSD-CURRENT захидлын жагсаалт | FreeBSD-CURRENT-н хэрэглээтэй хамаатай хэлэлцүүлэг |
| freebsd-isp | FreeBSD ашиглаж байгаа Интернэтийн үйлчилгээ үзүүлэгчдийн асуудлууд |
| freebsd-jobs | FreeBSD-тэй холбоотой ажил болон зөвлөгөөний боломжууд |
| freebsd-questions | Хэрэглэгчийн асуултууд болон техникийн дэмжлэг |
| FreeBSD аюулгүй байдлын мэдэгдлүүд захидлын жагсаалт | Аюулгүй байдлын мэдэгдлүүд (зохицуулалттай) |
| FreeBSD-STABLE; захидлын жагсаалт | FreeBSD-STABLE-н хэрэглээтэй холбоотой хэлэлцүүлэг |
| FreeBSD тест захидлын жагсаалт | Аль нэг жагсаалт уруу тест захиа явуулахын оронд энэ жагсаалт ашиглагдана |

*Техникийн жагсаалтууд:* Дараах жагсаалтууд нь техникийн хэлэлцүүлэгт зориулагдсан. Эдгээр жагсаалтын агуулга болон хэрхэн ашиглах талаар нарийн чанд зааврууд байдаг учраас аль нэг уруу нь элсэх юм уу эсвэл захидал илгээхээсээ өмнө жагсаалт болгоны дүрэм, журмыг та унших хэрэгтэй юм.

| Жагсаалт | Зорилго |
| --- | --- |
| FreeBSD ACPI захидлын жагсаалт | ACPI болон тэжээлийн удирдлагын хөгжүүлэлт |

| Жагсаалт | Зорилго |
| --- | --- |
| freebsd-afs | AFS-г FreeBSD уруу хөрвүүлэх |
| freebsd-aic7xxx | Adaptec® AIC 7xxx-д зориулж драйверууд хөгжүүлэх |
| freebsd-amd64 | FreeBSD-г AMD64 системүүд уруу хөрвүүлэх (зохицуулалттай) |
| freebsd-apache | Apache-тай холбоотой портуудын тухай хэлэлцүүлэг |
| freebsd-arm | FreeBSD-г ARM® процессоруудуруу хөрвүүлэх |
| freebsd-atm | FreeBSD дээр ATM сүлжээг ашиглах |
| freebsd-bluetooth | FreeBSD дээр Bluetooth® технологийг ашиглах |
| freebsd-cluster | FreeBSD-г кластерчлагдсан орчинд ашиглах |
| freebsd-cvsweb | CVSweb арчилгаа |
| freebsd-database | FreeBSD дээрх мэдээллийн бааз болон хөгжүүлэлтийн талаар хэлэлцэх |
| freebsd-desktop | Десктоп дээр FreeBSD-г ашиглах ба сайжруулах |
| freebsd-doc | FreeBSD-тэй холбоотой баримтуудыг үүсгэх |
| freebsd-drivers | FreeBSD-д зориулж төхөөрөмжийн драйверуудыг бичих |
| freebsd-eclipse | Eclipse IDE, хэрэгслүүд, клиентийн баялаг програмууд болон портуудын FreeBSD хэрэглэгчид. |
| freebsd-embedded | FreeBSD-г авсаар (embedded) програмуудад ашиглах |
| freebsd-eol | FreeBSD төслөөр дэмжигдэхээ больсон FreeBSD-тэй холбоотой програм хангамжийн дэмжлэг. |
| freebsd-emulation | Линукс/MS-DOS®/Windows® зэрэг системүүдийн эмуляц |
| freebsd-firewire | FreeBSD FireWire® (iLink, IEEE 1394) техникийн хэлэлцүүлэг |
| freebsd-fs | Файлын системүүд |
| freebsd-gecko | Gecko Rendering Engine-тэй холбоотой асуудлууд |
| freebsd-geom | GEOM-той холбоотой хэлэлцүүлгүүд болон шийдлүүд |

| Жагсаалт | Зорилго |
| --- | --- |
| freebsd-gnome | GNOME болон GNOME програмуудыг хөрвүүлэх |
| freebsd-hackers | Техникийн ерөнхий хэлэлцүүлэг |
| freebsd-hardware | FreeBSD-г тааруулахтай холбоотой тоног төхөөрөмжийн ерөнхий хэлэлцүүлэг |
| freebsd-i18n | FreeBSD Интернационалчлал |
| freebsd-ia32 | IA-32 (Intel® x86) тавцан дээрх FreeBSD |
| freebsd-ia64 | FreeBSD-г Intel®-ийн ирж байгаа IA64 системүүд уруу хөрвүүлэх |
| freebsd-infiniband | FreeBSD дээрх Infiniband |
| freebsd-ipfw | IP галт ханын кодны шинэ дизайны тухай техникийн хэлэлцүүлэг |
| freebsd-isdn | ISDN хөгжүүлэгчид |
| freebsd-jail | jail(8) боломжийн тухай хэлэлцүүлэг |
| freebsd-java | Java™ хөгжүүлэгчид болон JDK™-г FreeBSD уруу хөрвүүлж байгаа хүмүүс |
| freebsd-kde | KDE болон KDE програмуудыг порт хийх |
| freebsd-lfs | LFS-г FreeBSD руу порт хийх |
| freebsd-mips | FreeBSD-г MIPS® рүү порт хийх; |
| freebsd-mobile | Хөдөлгөөнт тооцооллын тухай хэлэлцүүлгүүд |
| freebsd-mono | FreeBSD дээрх Mono болон C# програмууд |
| freebsd-mozilla | Mozilla-г FreeBSD руу порт хийх |
| FreeBSD мультимедиа захидлын жагсаалт | Мультимедиа програмууд |
| freebsd-new-bus | Шугамын архитектурын тухай техникийн хэлэлцүүлгүүд |
| freebsd-net | Сүлжээний хэлэлцүүлэг ба TCP/IP эх код |
| freebsd-numerics | libm функцуудын өндөр чанартай шийдлийн тухай хэлэлцүүлгүүд |
| freebsd-office | FreeBSD дээрх Оффисын програмууд |
| freebsd-performance | Өндөр хурдтай ажиллагаа/ачааллын суулгалтуудад зориулсан ажиллагааг тааруулах асуултууд |
| freebsd-perl | Perl-тэй холбоотой портуудын арчилгаа |
| freebsd-pf | Пакет шүүгч галт ханын системийн тухай хэлэлцүүлэг болон асуултууд |

| Жагсаалт | Зорилго |
| --- | --- |
| freebsd-platforms | Intel® биш архитектур тавцангууд руу порт хийх талаар |
| freebsd-ports | Портын цуглуулгын тухай хэлэлцүүлэг |
| freebsd-ports-announce | Портын цуглуулгын тухай чухал мэдээнүүд болон зааврууд (зохицуулалттай) |
| freebsd-ports-bugs | Портын алдаанууд/PR-уудын тухай хэлэлцүүлэг |
| freebsd-ppc | FreeBSD-г PowerPC® руу порт хийх |
| freebsd-proliant | HP ProLiant сервер тавцангууд дээрх FreeBSD-ийн тухай техникийн хэлэлцүүлэг |
| freebsd-python | FreeBSD-тэй холбоотой Python-ий асуудлууд |
| freebsd-rc | rc.d систем болон түүний хөгжүүлэлттэй холбоотой хэлэлцүүлэг |
| freebsd-realtime | FreeBSD-ийн realtime өргөтгөлүүдийн хөгжүүлэлт |
| freebsd-ruby | FreeBSD-тэй холбоотой Ruby-ийн хэлэлцүүлэг |
| freebsd-scsi | SCSI дэд систем |
| FreeBSD аюулгүй байдлын захидлын жагсаалт | FreeBSD-д нөлөөлөх аюулгүй байдлын асуудлууд |
| freebsd-small | Embedded буюу авсаар програмуудад FreeBSD-г ашиглах (хуучин; үүний оронд freebsd-embedded-г ашигла) |
| freebsd-snapshots | FreeBSD хөгжүүлэлтийн хормын хувилбарын зарууд |
| freebsd-sparc64 | FreeBSD-г SPARC® дээр суурилсан системүүдэд порт хийх |
| freebsd-standards | C99 болон POSIX® стандартуудтай FreeBSD-ийн нийцтэй байдал |
| freebsd-sysinstall | sysinstall(8)-ийн хөгжүүлэлт |
| freebsd-tcltk | FreeBSD-ийн Tcl/Tk хэлэлцүүлэг |
| freebsd-threads | FreeBSD дээрх урсгалжилт |
| freebsd-tilera | Tilera төрлийн CPU-ууд руу FreeBSD-г порт хийх тухай хэлэлцүүлэг |
| freebsd-tokenring | FreeBSD дээрх Token Ring-ийн дэмжлэг |
| freebsd-toolchain | FreeBSD-д багтдаг хэрэгслүүдийн цуглуулгын тухай хэлэлцүүлэг |

| Жагсаалт | Зорилго |
| --- | --- |
| freebsd-usb | USB-д зориулсан FreeBSD-ийн дэмжлэгийн тухай хэлэлцүүлэг |
| freebsd-virtualization | FreeBSD-ийн дэмждэг төрөл бүрийн виртуалчлалтын техникүүдийн талаарх хэлэлцүүлэг |
| freebsd-vuxml | VuXML дэд бүтцийн тухай хэлэлцүүлэг |
| freebsd-x11 | FreeBSD дээрх X11-ийн арчилгаа ба дэмжлэг |
| freebsd-xen | FreeBSD-ийн Xen™ портын хэлэлцүүлэг - шийдэл ба хэрэглээ |
| freebsd-xfce | FreeBSD дээрх XFCE - порт хийх ба арчилгаа |
| freebsd-zope | FreeBSD дээрх Zope - порт хийх ба арчилгаа |

*Хязгаарлагдмал жагсаалтууд:* Дараах жагсаалтууд нь илүү тусгайлсан (болон шаардаж байгаа) олон нийтэд зориулагдсан бөгөөд ерөнхий нийтийн сонирхлыг татахгүй байж болох юм. Эдгээр хязгаарлагдмал жагсаалтуудын аль нэг уруу элсэхээсээ өмнө техникийн жагсаалтуудад орох нь зүйтэй юм. Ингэснээр та харилцааны ёс дүрмийг ойлгох болно.

| Жагсаалт | Зорилго |
| --- | --- |
| freebsd-hubs | Толин тусгал сайтуудыг ажиллуулж байгаа хүмүүс (дэд бүтцийн дэмжлэг) |
| freebsd-user-groups | Хэрэглэгчийн бүлгийн зохицуулалт |
| freebsd-wip-status | FreeBSD-ийн Work-In-Progress буюу хийгдэж байгаа ажлуудын төлөв |
| freebsd-wireless | 802.11 стек, хэрэгслүүд, драйверын хөгжүүлэлтийн талаарх хэлэлцүүлэг |

*Digest буюу товч мэдээллийн жагсаалтууд:* Дээрх бүх жагсаалтууд нь товч хэлбэрээр байдаг. Жагсаалтад бүртгүүлсний дараа та өөрийн бүртгэлийн тохируулгуудын хэсэгт товч мэдээлэл хүлээж авах тохируулгуудыг өөрчилж болно.

*SVN жагсаалтууд:* Дараах жагсаалтууд нь эх модны төрөл бүрийн талбарууд дахь өөрчлөлтүүдийн тухай бүртгэлийн мэдэгдлүүдийг харахыг хүссэн хүмүүст зориулагдсан байдаг. Тэдгээр нь *Зөвхөн Уншигдах* жагсаалтууд бөгөөд тэдгээр рүү захидал илгээх ёсгүй юм.

| Жагсаалт | Эх талбар | Талбарын тайлбар (зориулагдсан эх) |
| --- | --- | --- |
| svn-doc-all | /usr/doc | doc Subversion репозиторын бүх өөрчлөлтүүд ( user, projects ба translations бусад) |
| svn-doc-head | /usr/doc | doc Subversion репозиторын "head" салбарын бүх өөрчлөлтүүд |

| Жагсаалт | Эх талбар | Талбарын тайлбар (зориулагдсан эх) |
| --- | --- | --- |
| svn-doc-projects | /usr/doc/projects | doc Subversion репозиторын projects талбарын бүх өөрчлөлтүүд |
| svn-doc-svnadmin | /usr/doc | doc Subversion репозиторын удирдлагын скриптүүд, hook, болон бусад тохиргооны өгөгдлийн бүх өөрчлөлтүүд |
| svn-ports-all | /usr/ports | Портын Subversion репозиторын бүх өөрчлөлтүүд |
| svn-ports-head | /usr/ports | Портын Subversion репозиторын "head" салбарын бүх өөрчлөлтүүд |
| svn-ports-svnadmin | /usr/ports | Портын Subversion репозиторын удирдлагын скриптүүд, hook, болон бусад тохиргооны өгөгдлийн бүх өөрчлөлтүүд |
| Бүх src модны SVN нийлүүлэлтийн зурвасууд("user" ба "projects"-с бусад) | /usr/src | src Subversion репозиторын бүх өөрчлөлтүүд (user болон projects-с бусад) |
| head/-current src модны SVN нийлүүлэлтийн зурвасууд | /usr/src | src Subversion репозиторын "head" салбарын бүх өөрчлөлтүүд (FreeBSD-CURRENT салбар) |
| svn-src-projects | /usr/projects | src Subversion репозиторын projects талбарын бүх өөрчлөлтүүд |
| svn-src-release | /usr/src | src Subversion репозиторын releases талбарын бүх өөрчлөлтүүд |
| svn-src-releng | /usr/src | src Subversion репозиторын releng салбаруудын бүх өөрчлөлтүүд (аюулгүй байдал / хувилбар инженерчлэлийн салбарууд) |
| svn-src-stable | /usr/src | src Subversion репозиторын бүх тогтвортой салбаруудын бүх өөрчлөлтүүд |

| Жагсаалт | Эх талбар | Талбарын тайлбар (зориулагдсан эх) |
|---|---|---|
| svn-src-stable-6 | /usr/src | src Subversion репозиторын stable/6 салбарын бүх өөрчлөлтүүд |
| svn-src-stable-7 | /usr/src | src Subversion репозиторын stable/7 салбарын бүх өөрчлөлтүүд |
| svn-src-stable-8 | /usr/src | src Subversion репозиторын stable/8 салбарын бүх өөрчлөлтүүд |
| Зөвхөн 9-stable src модны SVN нийлүүлэлтийн зурвасууд | /usr/src | src Subversion репозиторын stable/9 салбарын бүх өөрчлөлтүүд |
| svn-src-stable-other | /usr/src | src Subversion репозиторын хуучин stable салбаруудын бүх өөрчлөлтүүд |
| svn-src-svnadmin | /usr/src | src Subversion репозиторын удирдлагын скриптүүд, хуукууд, болон бусад тохиргооны өгөгдлийн бүх өөрчлөлтүүд |
| svn-src-user | /usr/src | src Subversion репозиторын туршилтын user талбарын бүх өөрчлөлтүүд |
| svn-src-vendor | /usr/src | src Subversion репозиторын үйлдвэрлэгчийн ажлын талбарын бүх өөрчлөлтүүд |

### C.1.2. Хэрхэн бүртгүүлэх вэ

Жагсаалтанд бүртгүүлэхийн тулд дээр байгаа жагсаалтын нэрэн дээр дарах юм уу эсвэл https://lists.freebsd.org руу очоод сонирхож байгаа жагсаалт дээрээ дарах хэрэгтэй. Жагсаалтын хуудас бүртгүүлэхэд хэрэгтэй бүх зааврууд агуулсан байх ёстой.

Өгөгдсөн жагсаалт уруу илгээхийн тулд ердөө л жагсаалтын_нэр@FreeBSD.org хаяг уруу захидал илгээнэ. Дараа нь захидлын жагсаалтын дэлхий даяарх гишүүдэд тараагдах болно.

Жагсаалтаас гарахын тулд жагсаалтаас хүлээн авсан цахим захидал бүрийн доор байрлах URL дээр дарах хэрэгтэй. Өөрийгөө бүртгэлээс хасахын тулд бас цахим захидал жагсаалтын_нэр-unsubscribe@FreeBSD.org руу илгээж болно.

Дахин хэлэхэд бид техникийн захидлын жагсаалтуудад техникийн талын зүйлс хэлэлцэхийг танаас хүсэж байна. Хэрэв та зөвхөн чухал зарлалуудыг сонирхож байгаа бол та FreeBSD зарлалын захидлын жагсаалт-д бүртгүүлэх нь зүйтэй юм. Энэ жагсаалт нь зөвхөн

мэдээллийн сийрэг урсгалд зориулагдсан юм.

### C.1.3. Жагсаалтын дүрэм, журмууд

FreeBSD-ийн *бүх* захидлын жагсаалтууд нь тодорхой нэг үндсэн дүрмүүдтэй байдаг. Тэдгээрийг ашиглаж байгаа болгон дагаж мөрдөх ёстой. Эдгээр зааврууды дагахгүй байх тохиолдолд FreeBSD Postmaster postmaster@FreeBSD.org-с хоёр (2) анхааруулга бичгээр ирэх бөгөөд үүний дараа гурав дахь удаагаа зөрчсөн тохиолдолд илгээгчийг FreeBSD-ийн бүх захидлын жагсаалтаас хасаж тэдгээр рүү дахиж илгээхийг нь шүүх болно. Бид ийм дүрмүүд, арга хэмжээнүүд байгаад харамсаж байгаа боловч өнөөгийн Интернэт нь маш танхай балмад орчин юм шиг санагддаг бөгөөд олон хүмүүс эдгээрийн арга замууд ямар эмзэг болохыг хүндэтгэдэггүй билээ.

Дагаж мөрдөх дүрэм, журмууд:

- Аль ч илгээлтийн сэдэв нь илгээгдсэн жагсаалтынхаа үндсэн дүрэм, журмыг дагаж мөрдөж байх ёстой, өөрөөр хэлбэл хэрэв жагсаалт техникийн асуудлуудын талаар бол таны илгээж байгаа захидал техникийн хэлэлцүүлэг агуулсан байх ёстой. Хамааралгүй үргэлжлэх чалчаа эсвэл уурлаж шатсан байдал нь захидлын жагсаалтын утгыг түүн дээр байгаа бүхний хувьд алдагдуулж тэвчихгүй байдалд хүргэнэ. Ямар ч тодорхой сэдэвгүй чөлөөт хэлбэрийн хэлэлцүүлэгт зориулсан FreeBSD chat захидлын жагсаалт хэлэлцүүлэг чөлөөтэй байдаг бөгөөд үүнийг ашиглах ёстой юм.
- 2-оос илүү захидлын жагсаалтад зэрэг илгээж болохгүй бөгөөд зөвхөн 2 хүртэлхийг хэрэв тодорхой, илт хэрэгцээ байгаа тохиолдолд хоёр жагсаалт уруу илгээхийг зөвшөөрдөг. Ихэнх жагсаалтуудын хувьд аль хэдийн маш их хэмжээний бүртгэгдсэн хүмүүс давхацдаг бөгөөд ихэнх тусгай зориулагдсан холилдлуудыг ("-stable & -scsi" гэх мэт) тооцохгүй юм бол нэгэн зэрэг нэг жагсаалтаас олон жагсаалт уруу илгээх шалтгаан жинхэнэдээ байхгүй юм. Хэрэв мэдээ ийм замаар `Cc` мөрөн дээр олон захидлын жагсаалттайгаар тань руу илгээгдсэн бол дахин илгээхээсээ өмнө `Cc` мөрийг бас авч хаях ёстой юм. _ Анхлан эхлэгч нь хэн байгаагаас үл хамааран та олон жагсаалт уруу илгээх өөрийн илгээлтүүдийнхээ төлөө хариуцлага бас хүлээх ёстой юм._
- Хувь хүн рүү чиглэсэн халдлагууд болон гутаан доромжлол нь (маргаан) зөвшөөрөгддөггүй бөгөөд энэ нь хэрэглэгчид болон хөгжүүлэгчдийг адилхан хамардаг. Зөвшөөрөөгүй байхад эсвэл зөвшөөрөл өгөгдөхөөргүй байхад хувийн захиаг нээх эсвэл дахин илгээх зэрэг сүлжээн дэх харилцааны бүдүүлэг үйлдлүүд сайшаагддаггүй боловч хүчээр таслан зогсоогддоггүй. *Гэхдээ* жагсаалтын дүрэм, журмын хүрээнд ийм агуулга таарах зарим цөөн тохиолдол бас байдаг бөгөөд ийм тохиолдолд анхааруулга явуулах (эсвэл хөөх) арга хэмжээнүүдийг тухай бүрт нь авч болох юм.
- FreeBSD-тэй холбоогүй бүтээгдэхүүнүүд эсвэл үйлчилгээнүүдийг сурталчлахыг хатуугаар хориглодог бөгөөд хэрэв зөрчигч спамаар сурталчилж байгаа нь илт байвал түүнийг нэн даруй хөөхөд хүргэнэ.

*Жагсаалт бүрийн дүрэм, журмууд:*

**FreeBSD ACPI захидлын жагсаалт**

*ACPI болон тэжээлийн удирдлагын хөгжүүлэлт*

### freebsd-afs

*Andrew File System* буюу Эндрю файлын систем

Энэ жагсаалт нь AFS-г CMU/Transarc-аас порт хийж ашиглах тухай хэлэлцүүлэгт зориулагдсан

### FreeBSD зарлалын захидлын жагсаалт

*Чухал үйл явдлууд / түүхийн гол үеүүд*

Энэ нь FreeBSD-ийн чухал үйл явдлуудын тухай тохиолдлын зарлалуудыг сонирхсон хүмүүст зориулагдсан захидлын жагсаалт юм. Үүнд хормын хувилбарууд болон бусад хувилбаруудын тухай зарлалууд багтдаг. Энэ нь бас FreeBSD-ийн шинэ боломжуудын зарлалуудыг агуулдаг. Сайн дурынхныг уриалах гэх мэтийг бас агуулж болох юм. Энэ нь бага ачаалалтай, чанд хянагдсан захидлын жагсаалт юм.

### freebsd-arch

*Архитектур болон дизайны хэлэлцүүлгүүд*

Энэ жагсаалт нь FreeBSD-ийн архитектурын хэлэлцүүлэгт зориулагдсан. Захианууд нь ихэвчлэн маш техникийн шинж чанартай байх болно. Тохиромжтой сэдвүүдийн жишээг дурдвал:

- Нэгэн зэрэг хэд хэдэн өөрчлөн тохируулсан бүтээлтүүдийг ажиллуулахын тулд бүтээх системийг хэрхэн засаж сэлбэх талаар.
- Heidemann буюу Хайдеман давхаргуудыг ажиллуулахын тулд VFS-д юуг засварлах ёстой талаар.
- Ижил драйверуудыг олон шугамууд болон архитектурууд дээр цэвэрхэн ашиглаж чаддаг байхын тулд төхөөрөмжийн драйверийн интерфэйсийг хэрхэн өөрчлөх талаар.
- Сүлжээний драйверийг хэрхэн бичих талаар.

### freebsd-bluetooth

*FreeBSD дээрх Bluetooth®*

Энэ нь FreeBSD-ийн Bluetooth® хэрэглэгчид цугладаг хэлэлцүүлэг юм. Дизайны асуудлууд, шийдлийн тухай дэлгэрэнгүй зүйлс, засварууд, алдааны тайлангууд, төлвийн тайлангууд, боломжийн хүсэлтүүд, итгэмжлэн оруулсан зүйлсийн бүртгэлүүд болон Bluetooth®-тай холбоотой бусад бүх зүйлс энэ жагсаалтаар хэлэлцэгддэг.

### freebsd-bugbusters

*Асуудал Тайлагналттай (Problem Report) ажиллах хүчин чармайлтын зохицуулалт*

Энэ жагсаалтын зорилго нь Алдааны маэстро (Bugmeister), түүний Цох нядлагчид (Bugbusters) болон PR мэдээллийн баазад үнэн сэтгэлээсээ сонирхолтой бусад талуудад зориулсан зохицуулалт болон хэлэлцүүлэг болон үйлчлэх явдал юм. Энэ жагсаалт нь тусгайлсан алдаанууд, засварууд юм уу эсвэл PR-уудын тухай хэлэлцүүлгүүдэд зориулагдаагүй.

### freebsd-bugs

*Алдааны тайлангууд*

Энэ нь FreeBSD дээрх алдаануудыг тайлагнахад зориулагдсан захидлын жагсаалт юм. Алдаануудыг аль болох send-pr(1) тушаалыг ашиглах юм уу эсвэл түүний ВЭБ интерфэйсийг ашиглан илгээх ёстой.

### freebsd-chat

*FreeBSD-ийн хүрээнийхэнд хамаагүй техникийн бус зүйлс*

Энэ жагсаалт нь бусад жагсаалтаас хальсан техникийн бус, нийгмийн мэдээллийг агуулдаг. Жордан хүүхэлдэйн киноны өмхий хүрэнтэй адил уу эсвэл үгүй юу, томоор бичих үү эсвэл үгүй юу, хэн хэтэрхий их кофе ууж байна, хамгийн шилдэг пиво хаана исгэж байна, хэн подвалдаа пиво исгэж байна гэх мэт хэлэлцүүлгүүд үүнд багтдаг. Чухал үйл явдлуудын (болох үдэшлэгүүд, хуримнууд, төрсөн өдрүүд, шинэ ажлууд гэх мэт) тохиолдлын зарлал техникийн жагсаалтууд уруу хийгдэж болох боловч дараа дараагийн захианууд энэ -chat жагсаалт уруу заагдах ёстой байдаг.

### freebsd-chromium

*FreeBSD-тэй холбоотой Chromium-ийн хэлэлцүүлэг*

Энэ жагсаалт нь FreeBSD-д зориулсан Chromium-ийн дэмжлэгийг хэлэлцэхэд зориулагдсан. Энэ нь Chromium-ийн хөгжүүлэлт болон суулгалтыг хэлэлцэх техникийн жагсаалт юм.

### Core Team

*FreeBSD гол баг*

Энэ нь гол гишүүдийн ашиглахад зориулсан дотоод захидлын жагсаалт юм. FreeBSD-тэй холбоотой чухал зүйлийг шүүх эсвэл өндөр түвшний нарийвчилсан нягтлал хийх шаардлагатай үед захидлууд энэ жагсаалт уруу илгээгдэж болно.

### FreeBSD-CURRENT захидлын жагсаалт

*FreeBSD-CURRENT-г ашиглах тухай хэлэлцүүлэг*

Энэ нь FreeBSD-CURRENT-ийн хэрэглэгчдэд зориулсан захидлын жагсаалт юм. Хэрэглэгчдэд нөлөөлөх -CURRENT-с гарсан шинэ боломжуудын тухай анхааруулгууд болон -CURRENT байхын тулд хийх алхмуудын зааврууд энэ жагсаалт агуулдаг. "CURRENT"-г ашиглаж байгаа хэн бүхэн энэ жагсаалтад бүртгүүлэх ёстой. Энэ нь техникийн чанартай агуулгыг хатуу чанд хүлээж байдаг техникийн захидлын жагсаалт юм.

### freebsd-cvsweb

*FreeBSD CVSweb төсөл*

FreeBSD-CVSweb-ийн хэрэглээ, хөгжүүлэлт болон арчилгааны тухай техникийн хэлэлцүүлгүүд.

### freebsd-desktop

*Десктоп дээр FreeBSD-г ашиглах бас сайжруулах*

Энэ хэлэлцүүлэг нь десктоп дээрх FreeBSD-н талаар юм. Десктоп руу хөрвүүлэгчид болон хэрэглэгчдийн хувьд FreeBSD-н десктопын дэмжлэгийг сайжруулах болон түүнтэй холбоотой асуудлуудыг хэлэлцэхэд зориулсан гол талбар юм.

### freebsd-doc

*Бариmтжуулах төсөл*

Энэ захидлын жагсаалт нь FreeBSD-д зориулсан баримтжуулалтыг үүсгэхтэй холбоотой төслүүд болон асуудлуудын хэлэлцүүлэгт зориулсан билээ. Энэ захидлын жагсаалтын гишүүдийг бүхэлд нь "FreeBSD баримтжуулах төсөл" гэдэг. Энэ нь нээлттэй жагсаалт юм; чөлөөтэй нэгдэж хувь нэмрээ оруулаарай!

### freebsd-drivers

*FreeBSD-д зориулж төхөөрөмжийн драйверууд бичих*

Энэ нь FreeBSD дээрх төхөөрөмжийн драйверуудтай холбоотой техникийн хэлэлцүүлгүүдэд зориулсан форум юм. FreeBSD цөм дэх API-уудыг ашиглан төхөөрөмжийн драйверуудыг хэрхэн бичих талаар төхөөрөмжийн драйвер бичигчдийн асуух тэр талбар юм.

### freebsd-eclipse

*Eclipse IDE, хэрэгслүүд, клиентийн баялаг програмууд болон портуудын FreeBSD хэрэглэгчид.*

Энэ жагсаалтын зорилго нь FreeBSD тавцан дээр Eclipse IDE, хэрэгслүүд, клиентийн баялаг програмуудыг сонгох, суулгах, ашиглах, хөгжүүлэх болон арчлах, Eclipse IDE болон залгаасуудыг FreeBSD орчин уруу порт хийхэд туслахтай холбоотой бүх зүйлсэд зориулж хамтын дэмжлэгийг хангах явдал юм.

Eclipse хүрээнийхэн болон FreeBSD хүрээнийхэнд хоёуланд нь ашигтай байхаар тэдгээрийн хооронд мэдээлэл солилцохыг хөнгөвчлөх бас зорилготой юм.

Энэ жагсаалт нь Eclipse хэрэглэгчдийн хэрэглээн дээр үндсэндээ анхаарал тавьдаг боловч бас Eclipse тогтолцоог ашиглан FreeBSD програмуудыг хөгжүүлэхийг хүссэн хүмүүст зориулсан форум болдог.

### freebsd-embedded

*FreeBSD-г суулгагдсан авсаар програмуудад ашиглах*

Энэ жагсаалт нь FreeBSD-г суулгагдсан програмуудад ашиглахтай холбоотой сэдвүүдийг хэлэлцдэг. Энэ нь зөвхөн техникийн агуулга хүлээж байдаг техникийн захидлын жагсаалт юм. Энэ жагсаалтын зорилгын хувьд бид суулгагдсан системүүдийг тодорхойлдог бөгөөд эдгээр системүүд нь ширээний системүүд биш бөгөөд тооцоолох ерөнхий орчнуудад зориулагдаагүй тодорхой нэг зорилгоор ашиглагддаг тооцоолох төхөөрөмжүүд юм. Жишээнүүдэд бүх төрлийн утасны харилцуурууд, чиглүүлэгчүүд, шилжүүлэгчүүд болон PBX-үүд, алсын хэмжих төхөөрөмж, PDA-ууд, Point of Sale буюу POS системүүд гэх зэрэг орох бөгөөд зөвхөн энд дурдагдсанаар хязгаарлагдахгүй юм.

### freebsd-emulation

*Линукс/MS-DOS®/Windows® зэрэг бусад системүүдийн эмуляц*

Бусад үйлдлийн системүүдэд зориулж бичигдсэн программуудыг FreeBSD дээр ажиллуулахтай холбоотой техникийн хэлэлцүүлгүүдэд зориулсан форум юм.

### freebsd-eol

*FreeBSD төслийн дэмжихээ больсон FreeBSD-тэй холбоотой програм хангамжуудын тэгш дэмжлэг.*

Энэ жагсаалт нь FreeBSD төслийн албан ёсны (өөрөөр хэлбэл аюулгүй байдлын зөвлөгөөнүүд болон засваруудын хэлбэрээр) дэмжлэг үзүүлэхээ больсон FreeBSD-тэй холбоотой програм хангамжийн дэмжлэг үзүүлэх эсвэл ашиглахыг хүссэн хүмүүст зориулагдсан юм.

### freebsd-firewire

*FireWire® (iLink, IEEE 1394)*

Энэ нь FreeBSD дээр FireWire® (IEEE 1394, бас iLink гэгддэг) дэд системийн дизайн болон шийдлийн хэлэлцүүлэгт зориулсан захидлын жагсаалт юм. Хамааралтай сэдвүүдэд ялангуяа стандартууд, шугамын төхөөрөмжүүд болон тэдгээрийн протоколууд, адаптер хавтангууд/картууд/бичил схемүүд, тэдгээрийг зөв дэмжих кодын архитектур болон шийдэл ордог.

### freebsd-fs

*Файлын системүүд*

FreeBSD файлын системтэй холбоотой хэлэлцүүлгүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-gecko

*Gecko Rendering Engine*

Энэ нь FreeBSD дээр Gecko програмуудыг ашиглах талаар хэлэлцэх хэлэлцүүлэг юм.

FreeBSD дээрх Gecko портын програмуудыг тойрсон хэлэлцүүлэг, тэдгээрийн суулгалт, хөгжүүлэлт болон дэмжлэг.

### freebsd-geom

*GEOM*

GEOM болон түүнтэй холбоотой шийдлүүдтэй холбоотой хэлэлцүүлгүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-gnome

*GNOME*

FreeBSD системүүдэд зориулсан GNOME Ширээний орчны тухай хэлэлцүүлгүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-infiniband

*FreeBSD дээрх Infiniband*

FreeBSD дээрх Infiniband, OFED ба OpenSM-ийн талаар хэлэлцдэг техникийн захидлын жагсаалт.

### freebsd-ipfw

*IP Firewall буюу галт хана*

Энэ нь FreeBSD дээрх IP галт ханын кодыг дахин дизайн хийхтэй холбоотой техникийн хэлэлцүүлгүүдэд зориулсан форум юм. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-ia64

*FreeBSD-г IA64 руу порт хийх*

Энэ нь FreeBSD-г Intel®-ийн IA-64 тавцан руу порт хийхээр идэвхтэйгээр ажиллаж байгаа хувь хүмүүст зориулсан, асуудлууд гаргаж тавих юм уу эсвэл өөр шийдлүүдийн талаар хэлэлцэх техникийн захидлын жагсаалт юм. Техникийн хэлэлцүүлгийг дагах сонирхолтой хувь хүмүүс бас нэгдэж болох юм.

### freebsd-isdn

*ISDN холбоонууд*

Энэ нь FreeBSD-д зориулсан ISDN-ий дэмжлэгийн хөгжүүлэлтийг хэлэлцэхэд зориулсан захидлын жагсаалт юм.

### freebsd-java

*Java™ хөгжүүлэлт*

Энэ нь FreeBSD дээр чухал Java™ програмуудыг хөгжүүлэх болон JDK™-г порт хийх болон арчлах талаар хэлэлцэхэд зориулсан захидлын жагсаалт юм.

### freebsd-jobs

*Санал болгож байгаа болон хайж байгаа ажлууд*

Энэ нь FreeBSD-тэй ялангуяа холбоотой ажлын мэдэгдлүүд болон CV/resume-үүдийг илгээх зориулалттай форум юм. Өөрөөр хэлбэл хэрэв та FreeBSD-тэй холбоотой ажил хайж байгаа эсвэл танд FreeBSD оролцсон зарлах ажил байгаа бол энэ нь зөв газар юм. Энэ нь ерөнхий ажилтай холбоотой асуудлуудад зориулсан захидлын жагсаалт *биш* юм. Учир нь түүнд зориулсан тохирсон форумууд хаа нэгтээ аль хэдийн байдаг.

Энэ жагсаалт нь бусад `FreeBSD.org` захидлын жагсаалтуудын адил дэлхий даяар тардаг. Тиймээс та байрлал болон аялах эсвэл шинэ газар нүүхтэй холбоотой туслалцаа байгаа эсэх тэр хир хэмжээг ойлгосон байх хэрэгтэй.

Цахим захидлууд нь нээлттэй хэлбэршилтүүдийг ашиглах ёстой - цэвэр текстийг аль болох эрхэмлэсэн байх ёстой боловч Portable Document Format (PDF), HTML, болон бусад цөөн хэдэн хэлбэршилт ихэнх уншигчдын хувьд хүлээж авах боломжтой байдаг. Microsoft® Word (.doc) зэрэг хаалттай хэлбэршилтүүдээс захидлын жагсаалтын сервер

татгалзах болно.

**freebsd-kde**

*KDE*

FreeBSD системүүд дээрх KDE-тэй холбоотой хэлэлцүүлгүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

**freebsd-hackers**

*Техникийн хэлэлцүүлгүүд*

Энэ нь жагсаалт нь FreeBSD-тэй холбоотой техникийн хэлэлцүүлгүүдэд зориулагдсан. Энэ нь техникийн анхдагч захидлын жагсаалт юм. Энэ нь FreeBSD дээр идэвхтэйгээр ажиллаж байгаа хувь хүмүүст асуудлууд гаргаж тавих эсвэл өөр шийдлүүдийг хэлэлцэхэд нь зориулагджээ. Техникийн хэлэлцүүлгийг дагах сонирхолтой хувь хүмүүс нэгдэхэд дуртайяа байх болно. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

**freebsd-hardware**

*FreeBSD тоног төхөөрөмжийн ерөнхий хэлэлцүүлэг*

FreeBSD-ийн ажиллаж байгаа тоног төхөөрөмжийн төрлүүд, төрөл бүрийн асуудлууд болон юу худалдаж авах эсвэл авахгүй байхтай холбоотой саналуудын тухай ерөнхий хэлэлцүүлэг.

**freebsd-hubs**

*Толин тусгал сайтууд*

FreeBSD толин тусгал сайтуудыг ажиллуулж байгаа хүмүүст зориулсан зарлалууд ба хэлэлцүүлэг.

**freebsd-isp**

*Интернэтийн үйлчилгээ үзүүлэгчдэд зориулсан асуудлууд*

Энэ захидлын жагсаалт нь FreeBSD ашиглаж байгаа Интернэтийн үйлчилгээ үзүүлэгчидтэй (ISP-ууд) холбоотой сэдвүүдийг хэлэлцэхэд зориулагдсан. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

**freebsd-mono**

*FreeBSD дээрх Mono болон C# програмууд*

Энэ нь FreeBSD дээрх Mono хөгжүүлэлтийн тогтолцоотой хамааралтай хэлэлцүүлгүүдэд зориулагдсан жагсаалт юм. Энэ нь техникийн захидлын жагсаалт юм. Энэ нь FreeBSD рүү Mono болон C# програмуудыг хөрвүүлэхэхээр идэвхтэйгээр ажиллаж байгаа хүмүүс асуудлууд тавих эсвэл өөр боломжит шийдлүүдийг хэлэлцэхэд зориулагдсан юм. Техникийн хэлэлцүүлгүүдийг сонирхож дагах гэсэн хүмүүс бас оролцох боломжтой юм.

**freebsd-office**

*FreeBSD дээрх Оффисын програмууд*

Хэлэлцүүлэг FreeBSD дээрх оффисын програмууд, тэдгээрийг суулгах, хөгжүүлэх болон дэмжлэгийн талаар голчлон хэлэлцдэг.

### freebsd-ops-announce

*Төслийн дэд бүтцийн талаарх мэдээлэл*

Энэ захидлын жагсаалт нь FreeBSD.org төслийн дэд бүтэцтэй холбоотой асуудлууд болон өөрчлөлтүүдийг сонирхсон хүмүүст зориулагдсан.

Энэ зохицуулагдсан жагсаалт зөвхөн мэдээлэл, зард зориулагдсан бөгөөд хариулт, хүсэлт, хэлэлцүүлэг болон санал бодолд зориулагдаагүй.

### freebsd-performance

*FreeBSD-г тааруулах эсвэл хурдлуулах тухай хэлэлцүүлгүүд*

Энэ захидлын жагсаалт нь хакерууд, администраторууд ба/эсвэл оролцож байгаа талууд FreeBSD-тэй хамаатай ажиллагаатай холбоотой сэдвүүдийг хэлэлцэхэд зориулсан талбар юм. Хүлээж авах боломжтой сэдвүүдэд их ачаалалтай, ажиллагааны асуудлуудтай учирч байгаа эсвэл FreeBSD-ийн хязгааруудад тулж байгаа FreeBSD суулгацуудын тухай хэлэлцүүлэг ордог. FreeBSD-ийн ажиллагааг сайжруулах талаар ажиллах сонирхолтой талуудыг энэ жагсаалтад бүртгүүлэхэд туйлаас дэмждэг. Энэ нь FreeBSD-г хурдан, хүчирхэг, өргөжих боломжтой болгох сонирхолтой, туршлагатай FreeBSD хэрэглэгчид, хакерууд эсвэл администраторуудад туйлын таарсан техникийн жагсаалт юм. Энэ нь баримтыг уншсанаар орлуулж болох асуулт хариултын жагсаалт биш боловч хувь нэмэр оруулах юм уу эсвэл хариулагдаагүй бөгөөд ажиллагаатай холбоотой сэдвүүдийн талаар лавлаж болох талбар юм.

### freebsd-pf

*Пакет шүүгч галт ханын системийн тухай хэлэлцүүлэг ба асуултууд*

Пакет шүүгч (pf) галт ханын системийн тухай FreeBSD-тэй холбоотой хэлэлцүүлэг. Техникийн хэлэлцүүлэг болон хэрэглэгчийн асуултууд аль алийг хэлэлцэх боломжтой. Энэ жагсаалт нь ALTQ QoS тогтолцоог хэлэлцэх бас талбар юм.

### freebsd-pkg

*Бинар багцын удирдлага болон багцын хэрэгслүүдийн хэлэлцүүлэг*

FreeBSD систем дээр програм суулгахдаа бинар багцын хэрэгслүүд, хэлбэрүүд, тэдгээрийн хөгжүүлэлт ба FreeBSD дээрх багцын репозиторын удирдлага дахь дэмжлэг болон гуравдагч талуудын багцууд зэрэг бинар багцуудыг ашиглахтай хоблоотой бүхий л төрлийн хэлэлцүүлэг.

Багц үүсгэхдээ зөв үүсгэдэггүй портуудын талаарх хэлэлцүүлэг нь портын асуудалд хамаатай бөгөөд энэ жагсаалтад хамаарахгүй.

### freebsd-platforms

*Intel® биш тавцангууд руу порт хийх*

Тавцан хоорондох FreeBSD-ийн асуудлууд, ерөнхий хэлэлцүүлэг ба Intel® биш FreeBSD

портуудад зориулагдсан төслүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-ports

*"Портуудын" хэлэлцүүлэг*

FreeBSD-ийн "портын цуглуулга" (/usr/ports), портын дэд бүтэц болон портын ерөнхий зохицуулалтын чармайлтуудын тухай хэлэлцүүлгүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-ports-announce

*FreeBSD "Портын цуглуулгын" тухай чухал мэдээнүүд болон зааврууд*

"Портын цуглуулгын" (/usr/ports) хөгжүүлэгчид, порт хийгчид болон хэрэглэгчдэд зориулсан архитектур/дэд бүтцийн өөрчлөлтүүд, шинэ боломжууд, маш чухал шинэчлэх зааврууд болон хувилбар инженерчлэлийн мэдээлэл зэрэг чухал мэдээнүүд. Энэ нь бага ачаалалтай захидлын жагсаалт бөгөөд зарлалд зориулагдсан.

### freebsd-ports-bugs

*"Портуудын" алдаануудын хэлэлцүүлэг*

FreeBSD-ийн "портын цуглуулга"д (/usr/ports) зориулсан асуудлын тайлангууд, санал болгогдож байгаа портууд эсвэл портуудад хийгдэх өөрчлөлтүүдтэй холбоотой хэлэлцүүлгүүд. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-proliant

*HP Proliant сервер тавцангууд дээрх FreeBSD-ийн техникийн хэлэлцүүлэг*

Энэ захидлын жагсаалт нь Proliant-тай холбоотой драйверууд, удирдлагын програм хангамж, тохиргооны хэрэгслүүд ба BIOS-ийн шинэчлэлтүүдийн хэлэлцүүлэг зэрэг HP Proliant серверүүд дээрх FreeBSD-ийн хэрэглээний техникийн хэлэлцүүлэгт зориулагдсан талбар юм. Тиймээс энэ нь hpasmd, hpasmcli, болон hpacucli модулиудын талаар хэлэлцэх үндсэн газар юм.

### freebsd-python

*FreeBSD дээрх Python*

Энэ нь FreeBSD дээр Python дэмжлэгийг сайжруулахтай холбоотой хэлэлцүүлгүүдэд зориулсан жагсаалт юм. Энэ нь техникийн захидлын жагсаалт юм. Энэ нь Python, түүний гуравдагч талуудын модулиуд болон Zope-ийн юмсыг FreeBSD руу порт хийхээр ажиллаж байгаа хувь хүмүүст зориулагдсан юм. Техникийн хэлэлцүүлгийг дагах сонирхолтой хувь хүмүүс нэгдэхэд бас дуртайяа байх болно.

### freebsd-questions

*Хэрэглэгчийн асуултууд*

Энэ нь FreeBSD-ийн тухай асуултуудад зориулагдсан захидлын жагсаалт юм. Та асуултаа нэлээн техникийн гэж үзэж байгаагаас бусад тохиолдолд "how to" буюу "хэрхэн яаж" гэсэн

асуултуудыг техникийн жагсаалтууд руу илгээх ёсгүй юм.

### freebsd-ruby

*FreeBSD-тэй холбоотой Ruby-ийн хэлэлцүүлэг*

Энэ нь FreeBSD дээрх Ruby-ийн дэмжлэгтэй холбоотой хэлэлцүүлэгт зориулсан жагсаалт юм. Энэ нь техникийн захидлын жагсаалт юм. Ruby портууд, гуравдагч сангууд болон тогтолцоонууд дээр ажиллаж байгаа хувь хүмүүст зориулсан юм.

Техникийн хэлэлцүүлгийг дагах сонирхолтой хувь хүмүүс нэгдэхэд бас дуртайяа байх болно.

### freebsd-scsi

*SCSI дэд систем*

Энэ нь FreeBSD-д зориулагдсан SCSI дэд систем дээр ажиллаж байгаа хүмүүст зориулсан захидлын жагсаалт юм. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### FreeBSD аюулгүй байдлын захидлын жагсаалт

*Аюулгүй байдлын асуудлууд*

FreeBSD компьютерийн аюулгүй байдлын асуудлууд (DES, Kerberos, мэдэгдэж байгаа аюулгүй байдлын цоорхойнууд болон засварууд гэх мэт). Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг. Энэ нь асуулт, хариултын жагсаалт биш гэдгийг санах хэрэгтэй, гэхдээ FAQ-д хувь нэмэр оруулахыг (асуулт, хариултын аль нь ч байсан гэсэн) дэмждэг.

### FreeBSD аюулгүй байдлын мэдэгдлүүд захидлын жагсаалт

*Аюулгүй байдлын мэдэгдлүүд*

FreeBSD-ийн аюулгүй байдлын асуудлууд ба засваруудын мэдэгдлүүд. Энэ нь хэлэлцүүлгийн жагсаалт биш юм. Хэлэлцүүлгийн жагсаалт нь FreeBSD-security юм.

### freebsd-small

*FreeBSD-г суулгагдсан програмууд дээр ашиглах*

Энэ жагсаалт нь ердийн биш жижиг ба суулгагдсан авсаар FreeBSD суулгацуудтай холбоотой сэдвүүдийг хэлэлцдэг. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

Энэ жагсаалт нь freebsd-embedded-р хуучирч ашиглагдахаа больсон.

### freebsd-snapshots

*FreeBSD хөгжүүлэлтийн хормын хувилбарын зарууд*

Энэ жагсаалт нь FreeBSD-н хөгжүүлэлтийн head/ ба stable/ салбарын шинэ хормын хувилбарууд гарсан талаар мэдээлнэ.

### FreeBSD-STABLE; захидлын жагсаалт

*FreeBSD-STABLE-ийн хэрэглээний тухай хэлэлцүүлэг*

Энэ нь FreeBSD-STABLE-ийн хэрэглэгчдэд зориулсан захидлын жагсаалт юм. Хэрэглэгчдэд нөлөөлөх -STABLE-с гарсан шинэ боломжуудын тухай анхааруулгууд болон -STABLE байхын тулд хийх алхмуудын зааврууд энэ жагсаалт агуулдаг. "STABLE"-г ашиглаж байгаа хэн бүхэн энэ жагсаалтад бүртгүүлэх хэрэгтэй. Энэ жагсаалт нь техникийн захидлын жагсаалт бөгөөд зөвхөн техникийн агуулгыг хүлээж байдаг.

### freebsd-standards

*C99 & POSIX нийцтэй байдал*

Энэ нь C99 болон POSIX стандартуудтай FreeBSD хэрхэн нийцэж байгаатай холбоотой техникийн хэлэлцүүлгүүдэд зориулагдсан форум юм.

### freebsd-toolchain

*FreeBSD-д багтдаг хэрэгслүүдийн цуглуулгын үйлчилгээ*

Энэ захидлын жагсаалт нь FreeBSD-тэй цуг ирдэг хэрэгслүүдийн цуглуулгын үйлчилгээтэй холбоотой хэлэлцүүлэгт зориулагдсан. Энэ сэдэвт Clang болон GCC-ийн төлөв бас асемблер, линкер болон дибаг хийгч зэрэг програм хангамжууд багтаж болно.

### freebsd-usb

*USB-д зориулсан FreeBSD-ийн дэмжлэгийг хэлэлцэх*

Энэ нь USB-д зориулсан FreeBSD-ийн дэмжлэгтэй холбоотой техникийн хэлэлцүүлгүүдэд зориулсан захидлын жагсаалт юм.

### freebsd-user-groups

*Хэрэглэгчийн бүлгийн зохицуулалтын жагсаалт*

Энэ нь бүс нутгийн Хэрэглэгчдийн бүлэг бүрийн зохицуулагчид нэг нэгэнтэйгээ болон гол багийн хариуцсан хүнтэй юмс хэлэлцэхэд зориулагдсан захидлын жагсаалт юм. Энэ захидлын жагсаалт нь ерөнхий агуулгууд болон хэрэглэгчийн бүлгүүдэд тархсан төслүүдийн зохицуулалтад тааран хязгаарлагдсан байх ёстой.

### freebsd-virtualization

*FreeBSD-ийн дэмждэг төрөл бүрийн виртуалчлалын техникүүдийн талаарх хэлэлцүүлэг*

FreeBSD-ийн дэмждэг төрөл бүрийн виртуалчлалын техникүүдийн талаар хэлэлцэх жагсаалт. Нэг талаасаа үндсэн ажиллагааны шийдэл болон шинэ техникүүд нэмэх тал дээр төвлөрөх болно. Нөгөө талаасаа хэрэглэгчдэд ямар нэг асуудал гарсан үед тусламж гуйх эсвэл өөрсдийнхөө хэрэглээний тохиолдлуудыг хэлэлцэх хэлэлцүүлэг байх болно.

### freebsd-wip-status

*FreeBSD-ийн Work-In-Progress буюу хийгдэж байгаа ажлуудын төлөв*

Энэхүү захидлын жагсаалтыг FreeBSD-тэй холбоотой өөрийн эхлүүлж байгаа ажил болон түүний явцыг зарлахад ашиглаж болох юм. Захидлууд нь хянагддаг. Илүү сэдэвт ойр

FreeBSD-ийн жагсаалт руу "To:" дээр хаягийг бичин, харин энэ жагсаалт руу "BCC:" хийж захидлыг илгээх нь зүйтэй юм. Энэ жагсаалтад хэлэлцүүлэг зөвшөөрөгддөггүй учир ингэснээр таны WIP бас тухайн сэдэвтэй холбоотой жагсаалтад хэлэлцэгдэх боломжтой юм.

Тохиромжтой захидлуудын жишээг архиваас үзнэ үү.

Энэ жагсаалтад явуулсан захидлуудын редакторын тоймыг FreeBSD-ийн вэб хуудсанд хэдэн сар тутам Төлвийн Тайлангийн хэсэг болгон тавьж болох юм. Тэндээс илүү олон жишээнүүд болон хуучин тайлангуудыг та бас олж болох юм.

**freebsd-wireless**

*802.11 стек, хэрэгслүүд, драйверын хөгжүүлэлтийн талаарх хэлэлцүүлэг*

FreeBSD-wireless жагсаалт нь 802.11 стек (sys/net80211), төхөөрөмжийн драйвер болон хэрэгслүүдийг хөгжүүлэхтэй холбоотой асуудлыг хэлэлцэхэд зориулагдсан. Үүнд цоорхой, шинэ боломжууд болон арчлахтай холбоотой асуудлууд багтана.

**freebsd-xen**

*FreeBSD-ийн Xen™ портын хэлэлцүүлэг - шийдэл ба хэрэглээ*

Энэ жагсаалт нь FreeBSD-ийн Xen™ портын талаар хэлэлцдэг. Захидлын урсгалын түвшин бага болохоор дизайны нарийн зүйлсүүд ба шийдлийн техникийн хэлэлцүүлэг болон хэрэгжүүлэхтэй холбоотой удирдлагын асуудлуудыг хэлэлцэхээр зориулагдсан.

**freebsd-xfce**

*XFCE*

Энэхүү хэлэлцүүлэг нь XFCE орчинг FreeBSD-д оруулахтай хамааралтай юм. Энэ нь техникийн захидлын жагсаалт юм. XFCE-г FreeBSD рүү порт хийхээр идэвхтэй ажиллаж байгаа хүмүүст асуудлыг тавих эсвэл өөр шийдэл ярилцахад нь зориулагдсан юм. Техникийн хэлэлцүүлгийг дагаж унших сонирхолтой хүмүүс бас бүртгүүлж болно.

**freebsd-zope**

*Zope*

Энэхүү хэлэлцүүлэг нь Zope орчинг FreeBSD-д оруулахтай хамааралтай юм. Энэ нь техникийн захидлын жагсаалт юм. Zope-г FreeBSD рүү порт хийхээр идэвхтэй ажиллаж байгаа хүмүүст асуудлыг тавих эсвэл өөр шийдэл ярилцахад нь зориулагдсан юм. Техникийн хэлэлцүүлгийг дагаж унших сонирхолтой хүмүүс бас бүртгүүлж болно.

### C.1.4. Захидлын жагсаалтууд дээрх шүүлт

FreeBSD-ийн захидлын жагсаалтууд нь спам, вирус, болон бусад хүсээгүй цахим захидлуудаас сэргийлэхийн тулд олон аргаар шүүгддэг. Энэ хэсэгт тайлбарласан шүүлт хийх үйлдлүүд нь захидлын жагсаалтуудыг хамгаалахад хэрэглэгддэг бүх зүйлсийг хамрахгүй.

Захидлын жагсаалтад зөвхөн тодорхой төрлийн хавсралтууд зөвшөөрөгддөг. Доорх жагсаалтад дурдагдаагүй MIME агуулгын төрөл бүхий бүх хавсралтууд захидлын жагсаалтад цахим захидал түгээгдэхээс өмнө авагддаг.

- application/octet-stream
- application/pdf
- application/pgp-signature
- application/x-pkcs7-signature
- message/rfc822
- multipart/alternative
- multipart/related
- multipart/signed
- text/html
- text/plain
- text/x-diff
- text/x-patch

Зарим захидлын жагсаалтууд бусад MIME агуулгын төрлийн хавсралтуудыг зөвшөөрч болох боловч дээрх жагсаалт нь захидлын жагсаалтуудын ихэнх хэсэгт хамаарах болно.

Хэрэв цахим захидал HTML болон цэвэр текст хувилбарыг хоёуланг агуулж байвал HTML хувилбарыг устгах болно. Хэрэв цахим захидал зөвхөн HTML хувилбарыг агуулж байвал энэ нь цэвэр текст рүү хөрвүүлэгдэх болно.

# C.2. Usenet Newsgroups буюу мэдээний бүлгүүд

FreeBSD-ийн хоёр тусгай мэдээний бүлгээс гадна FreeBSD-н талаар хэлэлцдэг юм уу эсвэл FreeBSD хэрэглэгчидтэй холбоотой бусад олон бүлгүүд байдаг.

## C.2.1. BSD-тэй холбоотой мэдээний бүлгүүд

- comp.unix.bsd.freebsd.announce
- comp.unix.bsd.freebsd.misc
- de.comp.os.unix.bsd (German)
- fr.comp.os.bsd (French)
- it.comp.os.freebsd (Italian)

## C.2.2. Бусад сонирхол татахуйц UNIX®-ийн мэдээний бүлгүүд

- comp.unix
- comp.unix.questions
- comp.unix.admin
- comp.unix.programmer

- comp.unix.shell
- comp.unix.user-friendly
- comp.security.unix
- comp.sources.unix
- comp.unix.advocacy
- comp.unix.misc
- comp.unix.bsd

### C.2.3. X Цонхот Систем

- comp.windows.x.i386unix
- comp.windows.x
- comp.windows.x.apps
- comp.windows.x.announce
- comp.windows.x.intrinsics
- comp.windows.x.motif
- comp.windows.x.pex
- comp.emulators.ms-windows.wine

## C.3. Word Wide Web серверүүд

### C.3.1. Форум, блог, болон нийгмийн сүлжээнүүд

- FreeBSD-ийн форумууд нь FreeBSD-ийн асуултууд болон техникийн хэлэлцүүлэгт зориулсан вэб дээр суурилсан хэлэлцүүлгийн форумууд юм.
- FreeBSD-ийн ертөнц нь FreeBSD-ийн хөгжүүлэгчдийн бичсэн хэдэн арван блог дээрх мэдээллийг цуглуулж харуулах боломжийг бүрдүүлдэг. Олон хөгжүүлэгч юун дээр ажиллаж байгаагаа болон шинэ нөхөөсүүд ба бусад хийгдэж байгаа ажлуудынхаа талаар илгээхдээ үүнийг ашигладаг.
- YouTube суваг дээрх BSD-ийн хурлууд нь дэлхий даяар болсон BSD хурлуудын өндөр чанартай видеонуудын цуглуулгыг агуулдаг. Энэ нь гол хөгжүүлэгчдийн FreeBSD дээрх өөрсдийн шинэ ажлуудынхаа талаар илтгэл тавихыг нь үзэх шилдэг арга юм.

### C.3.2. Албан ёсны толин хуудаснууд

## C.4. Цахим захидлын хаягууд

Дараах хэрэглэгчийн бүлгүүд нь FreeBSD-тэй холбоотой цахим захидлын хаягуудыг өөрсдийн гишүүддээ хангадаг. Жагсаагдсан администратор нь хаягийг буруугаар ашигласан тохиолдолд цуцлах эрхийг эдэлдэг.

| Домэйн | Боломжууд | Хэрэглэгчийн бүлэг | Администратор |
| --- | --- | --- | --- |
| ukug.uk.FreeBSD.org | Зөвхөн дамжуулах | ukfreebsd@uk.FreeBSD.org | Ли Жонстон lee@uk.FreeBSD.org |

# Appendix D: PGP түлхүүрүүд

The OpenPGP keys of the FreeBSD.org officers are shown here. These keys can be used to verify a signature or send encrypted email to one of the officers. A full list of FreeBSD OpenPGP keys is available in the PGP Keys article. The complete keyring can be downloaded at pgpkeyring.txt.

## D.1. Officers

### D.1.1. Аюулгүй байдлын Түшмэдийн Баг <security-officer@FreeBSD.org>

```
pub   rsa4096/D9AD2A18057474CB 2022-12-11 [C] [expires: 2026-01-24]
      Key fingerprint = 0BE3 3275 D74C 953C 79F8  1107 D9AD 2A18 0574 74CB
uid                           FreeBSD Security Officer <security-officer@freebsd.org>
sub   rsa4096/6E58DE901F001AEF 2022-12-11 [S] [expires: 2026-01-15]
sub   rsa4096/46DB26D62F6039B7 2022-12-11 [E] [expires: 2026-01-15]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGOVdeUBEADHF5VGg1iPbACB+7lomX6aDytUf0k2k2Yc/Kp6lfYv7JKU+1nr
TcNF7Gt1YkajPSeWRKNZw/X94g4w5TEOHbJ6QQWx9g+N7RjEq75actQ/r2N5zY4S
ujfFTepbvgR55mLTxlxGKFBmNrfNbpHRyh4GwFRgPlxf5Jy9SB+0m54yFS4QlSd0
pIzO0CLkjHUFy/8S93oSK2zUkgok5gLWruBXom+8VC3OtBElkWswPkE1pKZvMQCv
VyM+7BS+MCFXSdZczDZZoEzpQJGhUYFsdg0KqlLv6z1rP+HsgUYKTkRpcrumDQV0
MMuCE4ECU6nFDDTnbR8Wn3LF5oTt0GtwS0nWf+nZ1SFTDURcSPR4Lp/PKjuDAkOS
P8BaruCNx1ItHSwcnXw0gS4+h8FjtWNZpsawtzjjgApcl+m9KP6dkBcbN+i1DHm6
NG6YQVtVWyN8aOKmoC/FEm1CWh1bv+ri9XOkF2EqT/ktbjbT1hFoFGBkS9/35y1G
3KKyWtwKcyF4OXcArl6sQwGgiYnZEG3sUMaGrwQovRtMf7le3cAYsMkXyiAnEufa
deuabYLD8qp9L/eNo+9aZmhJqQg4EQb+ePH7bGPNDZ+M5oGUwReX857FoWaPhs4L
dAKQ1YwASxdKKh8wnaamjIeZSGP5TCjurH7pADAIaB3/D+ZNl2a7od+C1wARAQAB
tDdGcmVlQlNEIFNlY3VyaXR5IE9mZmljZXIgPHNlY3VyaXR5LW9mZmljZXJAZnJl
ZWJzZC5vcmc+iQJSBBMBCgA8AhsBBAsJCAcEFQoJCAUWAgMBAAIeBQIXgBYhBAvj
MnXXTJU8efgRB9mtKhgFdHTLBQJjlXeQBQkF3u+rAAoJENmtKhgFdHTLOVoQALS3
cj7rqYkHiV4zDYrgPEp9O1kAyGI8VdfGAMkDVTqr+wP4v/o7LIUrgwZl5qxesVFB
VknFr0Wp5g9h0iAjasoI5sDd6tH2SmumhBHXFVdftzDQhrugxH6fWRhHs0SaFYCk
Qt5nFbcpUfWgtQ35XTbsL8iENdYpjKXsSFQrJneGSwxIjWYTFn6ps/AI3gwR8+Bn
OffEFdYugJ049O6Vu6YBFJHrnMO7NbF4v95dVYuLtpMIaXWM+V9KITmhaBzFz5fM
Q7UOzcLlbxOYKNIWcp8QQk429mayKW5VUeUExUD1ZzBHn+P6ZG7QTMDu/RmBqiHo
ewCMVz4n9uXT5BiOngE4CvS0WQwHzK+k9MLpG2u/Bo9+LT0Ceh9Ou1rfU5+0tRwl
GyOFFjf3INS7I7gkcAwxQ7dzDItN/UQPZpg8y9mABU2x4enz0AvTnb61d/1dnTEr
tdNgU433he0ZnD1HurZCjBEWC656wv6iMdWcD8gjhMbmEpPmjvXcYlTO6zhEygSM
DiwdQCWK2W4++YJerA6ULBi3niNWBpofOFH8XylV56ruhjtHCo7+/3carcMoPOJv
lVZ1zCKxLro3TRBT15JTFBGqblRyTopFK3PuxW//GTnZOtpQEOV6yL4RAXcWeC1d
1hb5k/YxUmRF6XsDNEH4b08T8ZO8dV3dAV43Wh1oiQEzBBABCAAdFiEEuyjUCzYO
7pNq7RVv5fe8y6O93fgFAmObXVYACgkQ5fe8y6O93fiBlwf/W8y1XXJIx1ZA3n6u
f7aS70rbP9KFPr4U0dixwKE/gbtIQ9ckeNXrDDWz0v0NCz4qS+33IPiJg1WcY3vR
W90e7QgAueCo5TdZPImPbCs42vadpa5byMXS4Pw+xyT+d/yp2oLKYbj3En4bg1GM
w71DezIjvV+e01UR++u1t9yZ8LOWM5Kumz1zyQLZDZ8qIKt1bBfpa+E0cEqtNQWu
```

```
iGhQE3AHI8eWV+jBkg5y2zHRIevbWb1UPsj43lgkFtAGHk9rrM8Rmgr4AXr531iD
srBwauKZ/MElcF3MINuLH+gkPPaFHw/YIpLRLaZXZVsw3Xi1RNXI2n2ea29dvs/C
Lcf1vYkCMwQQAQgAHRYhBPwOh4rlr+eIAo1jVdOXkvSep+XCBQJjm14FAAoJENOX
kvSep+XC0DcP/1ZB7k9p1T+9QbbZZE1PJiHby3815ccH3XKexbNmmakHIn3L6Cet
F891Kqt9ssbhFRMNtyZ/k/8y8Hv5bKxVep5/HMyK+8aqfDFN0WMrqZh0/CiR6DJh
gnAmPNw/hAVHMHaYGII9kCrFfPFJ02FKoc81g9F08odb7TV+UlvRjkErhRxF+dGS
wQoO0RCbf0Z1cs7nd0Vb2z4IJh4XMxBjWc/uQ2Q9dH/0uRzwpAnR4YX+MG5YrX7Z
zBvDyR0r76iQwRSDKgioNgkr6R3rq1NZGdaj+8b0LzdOqtzKJ/eupDe3+H67e/EN
qymtreGjrubpiU9bKvYArisUqhE5KtguryvR6Qz9bj87nPg33DT3WWGVrwFRxBox
dbWzjQFv0wug8m4GAwVF7fPR5/eW7IHw8zvgn0vSPcZz7MZ4e6Y5jN4kA5/xWJYZ
Sps54qQWB+FA30unIXN68KqdIzONIbtaY3W4/JjJUCm4T+wEjKaH+wJX8w1DMjlg
mkTmGh/UrTyC1vXbPgk9Sy3cRTICR1T9z7W8UlmTtnKrUklrjlFR7SXzrEXzLGOX
Fm+NEHpHNXqzcm6c3QfzY/yQ9HSAQ/t7SUQ9caRePbDz3/msyPxtGFor9roQv6VN
wRXCyRgkH4Y5tPhJAQ8G/FxX+VXFb93QL0lfelb23/BBu6cUwW63SRn5iHUEExYI
AB0WIQQeB2Johg/5/ikUnJwDU9SVF1S13AUCZISO3wAKCRADU9SVF1S13NnqAP95
LA1Om9XSAKl76VtV+L3JPDdAwIdbNa0OsRT4Wm7U3wD/YoFrdHXVHHQFKwYeUUhj
XZcxnZLe9Ixo0/JP+RVFVw65Ag0EY5V2yQEQAOqjzPpMUCGu8eElXnAd2PruC6hi
+lc/yC90KqizxIuW6qLQBaAkTCWq7suYpDqoygn7YM3rL50S285WAECAXrcst/cV
Aqr0UH/e6p4iJCUIiXcfjd/wq20RnN/+VuvLhjpCFLY5czfVS31D7Uh9MbC+zUTz
8nVTiNCsAao0qSdfJDIzB4nSO+9xIsme/dLsI5QlU5PdxOBV6HdEhCUXOoratJCb
KAOlLxtPwyMKxmv4oZ7Mqlt10peKjhpBb97qcIzJhHxujQZD0OmzIA6xoQ2eSCGd
xCEDsZ09kr3Esw1AwKnQ51xmWpFWNFk6627M1bo8+hzOz81CrTZhYrgE+1JXv6V1
L2A9lMsimdE1BHNycDS+dBOpIB9qxXCwAab4ykfvNxox/ZPDUrTy7v7mDI5uDNTN
CYYsKCj1UidycOKzSziB9Oa2uvmMJ5XstgNBf7Z8Cky1dtVd4ol6bU9L5nos9tbY
eSXFi4bmcWB7AJiVCMq6N+LBbUKWGLglB4TU1qhttpqv31X9V6ges5gARY/RuRTK
sVyhwsn7SDcqmNKRy0im2AYakwEp7hT07ulahOSLxjP+5hCf+nSJlwbxJ8ozwjjb
zeN2yLlJSI00klkIFBNUDt3wzFRW/n6qlf+/lepgzekfNrYMtfPB8AT07Z2A3U4x
lgiV346dZymbY/EjABEBAAGJBHIEGAEKACYWIQQL4zJ110yVPHn4EQfZrSoYBXR0
ywUCY5V2yQIbAgUJAgIpAAJACRDZrSoYBXR0y8F0IAQZAQoAHRYhBLYVJ36BCH33
XIGDO25Y3pAfABrvBQJjlXbJAAoJEG5Y3pAfABrvuBsQAOlQFPXhx6whO4yw5Ziz
IS02YHhSVMVYKS2T9jPIKi1qxnEiEw9eKH0bW0Oj0TEhZPyM2NJID7DRWK5r8+Ks
Mu8jwm1fUmIrefAx6fCVfCWRECT1MlbL3jhh6AcX/nK2e3Bn8vgExhzczO3JlvD6
wPCc0FkpiY7yDB9ihu1+gbE5Hg6dvftttRXDrbEdAifbNp9KYxDigxdlOb0S14hj
CBysLWH5Su/khcIlkeuqZcI8TmDldnUb2OqTCVpFhaNwsPSrHBzmb0s2sXo4FL03
pLsOdwhi31W6kjk4KvW5FKrOpoEwUMKVNMf50DHdvonUoUHRSIc/cV5NqUWHwvc0
T5031qk0CCRRa/+/iij/p2RG7c1mx7ZECj+jZfmvjSqT+WHJ1BFlNJMWyK4fdVRZ
WyCaoAecdbukwzDwUCUqHJFIWeFtbut7SOPxcwg7sbnKNApAKdi491dvH75s/U/O
wRYO/2P+ymHlqtyix2jq0ReSVYcQPXswQ8i2ifX41F+xTSl4RWCBBeXB1Nxk3+Hs
V4Jnnp1zAJZ0KlKW/oJxbNFdI1TImkpr2p8ioFf+aiePLvDkgeaG8vABgjoihPXW
HVAMR8Z+GvBY/A6OdexpibkTvC/zDr0/Exs4lsylZKDwvvFbctcpHVXBeCBQLX5v
fLrsTkaCLWF/SV9OdMykvYKU7ZAP+gKEwhp+HPFuOHZbOBhqFUdkfeCkdzX/QGdz
Tuz349roRhgz2vRfN7MtbtuzA6NWWhEWt5DcUgX/Y5I3Q4Z2bt3JiXQ6WJMgMMOX
Ar+XxtxyRRykc1HV3DQ/cq8OWYubNnIbgebPNIFr2OIWKsR9yDaucZzpmLfzaMZU
Au5hWmU9fIw5SIKGnQABBnNMhilfD+CkETp6baTvjTK4rpaobjJdeCTrsWgfXRNC
8x3hDvcrjPD70MylOGVQdx8GYChWJnCKXsLTGX7KwdfxkjclTyzWvdcCemp0eLha
mLGb9y1dtWdNIDcVCvZJy0lipHVUdFYYxb4iLZJANL631tlPM6AA8sO1/L4mqEGn
AIHVrUQd+2QkSiOl9mKlpgaR/fJz683BR5Qen9ywX0JPtBupqPW3t9VbO/uNxUql
HCeAhPi9NLOpujpYLfgW5QAfS3u0nkp5nrbkCoQUua2q0Oj7J0mFmtWtcE1c9+TH
mFJVb8j2G9yQw3ADe3Qp9ALazP5nVDVri8NZBhHK1/KuBmRYZtcyfqXUnKoiiWAl
m5rHaRiztW7e3wqm2oJu/RkEAagybutEuBWh2Ej2+gDxjEKKtIKGu54lif4kqTww
jKTcN1ekGihwwgCMUkBSBeNXk1ClkzLFHwESJCcFwdEgpVYQTKFsu0emYISyco3I
```

```
pUajGzfUiQRyBBgBCgAmAhsCFiEEC+MydddMlTx5+BEH2a0qGAV0dMsFAmWFy28F
CQPyfaYCQMF0IAQZAQoAHRYhBLYVJ36BCH33XIGDO25Y3pAfABrvBQJjlXbJAAoJ
EG5Y3pAfABrvuBsQAOlQFPXhx6whO4yw5ZizIS02YHhSVMVYKS2T9jPIKi1qxnEi
Ew9eKH0bW0Oj0TEhZPyM2NJID7DRWK5r8+KsMu8jwm1fUmIrefAx6fCVfCWRECT1
MlbL3jhh6AcX/nK2e3Bn8vgExhzczO3JlvD6wPCc0FkpiY7yDB9ihu1+gbE5Hg6d
vftttRXDrbEdAifbNp9KYxDigxdlOb0S14hjCBysLWH5Su/khcIlkeuqZcI8TmDl
dnUb2OqTCVpFhaNwsPSrHBzmb0s2sXo4FL03pLsOdwhi31W6kjk4KvW5FKrOpoEw
UMKVNMf50DHdvonUoUHRSIc/cV5NqUWHwvc0T5031qk0CCRRa/+/iij/p2RG7c1m
x7ZECj+jZfmvjSqT+WHJ1BFlNJMWyK4fdVRZWyCaoAecdbukwzDwUCUqHJFIWeFt
but7SOPxcwg7sbnKNApAKdi491dvH75s/U/OwRYO/2P+ymHlqtyix2jq0ReSVYcQ
PXswQ8i2ifX41F+xTSl4RWCBBeXB1Nxk3+HsV4Jnnp1zAJZ0KlKW/oJxbNFdI1TI
mkpr2p8ioFf+aiePLvDkgeaG8vABgjoihPXWHVAMR8Z+GvBY/A6OdexpibkTvC/z
Dr0/Exs4lsylZKDwvvFbctcpHVXBeCBQLX5vfLrsTkaCLWF/SV9OdMykvYKUCRDZ
rSoYBXR0yyOqEACitDvbkbfjaton6izr4T8QU2yvhJHkf4B6KeVDbKY1J4784OxX
p2bJgPeF53SYBe8gm3YHjp8ULh4A/19U4hswyE8ymcm5nIs8OLyBdxkuBZJGEnzx
H3woiyYqWH7991kzhEjUkuMgKLuTI1HiO0oLMuPQNhUHOnWafSVPC0XO/tIL12Om
oUuc7ligY9Z9AceFjTZOuHamixHAAc6hpxdIW+yhC/qTpc2VK0niWeuQfq3453iR
Tf9MnR5Beztl3ZYRWcx7UiFuKGwZwBibNnNmUs6GyQcJ5UTa1oeJcLqHi0Lf/r0j
Xo3wgJq7EZjjVyU+GI2ZVoDOaS6c4/OvLm62XoeSlnn/dQxUcjUki+x8lb69IxSF
1xAgsC/oNtFZYd5rHdlnqIBUYK0lLtSCXBkzVeivSiQa0hL5on8LDu1nw2bXyW61
yt/YxVb4FanMxAqdYVBhOfU0RaPNifH01rbb4TwC9bTZN1LQ1KI/Swb/SruUE0Ry
T28fhYRtsReS2PnUODghJSFDJbwFbBZf6RKI16q1xqKRRvxIWPm+lMOi1NLOKR9P
+OKy9HmChMw0UJUcVl1cJ2xtRl3wi5t6AA6HoNv/TrLeYVgMR9wYmKlpvjTQ5jTd
rbHD1XP5jGsp8QsJMGja1m/7cryReCpcVxvImeReaOdgz+zDmQqq3O5zuIkEcgQY
AQoAJgIbAhYhBAvjMnXXTJU8efgRB9mtKhgFdHTLBQJnhEEJBQkF0/JAAkDBdCAE
GQEKAB0WIQS2FSd+gQh991yBgztuWN6QHwAa7wUCY5V2yQAKCRBuWN6QHwAa77gb
EADpUBT14cesITuMsOWYsyEtNmB4UlTFWCktk/YzyCotasZxIhMPXih9G1tDo9Ex
IWT8jNjSSA+w0Viua/PirDLvI8JtX1JiK3nwMenwlXwlkRAk9TJWy944YegHF/5y
tntwZ/L4BMYc3MztyZbw+sDwnNBZKYmO8gwfYobtfoGxOR4Onb37bbUVw62xHQIn
2zafSmMQ4oMXZTm9EteIYwgcrC1h+Urv5IXCJZHrqmXCPE5g5XZ1G9jqkwlaRYWj
cLD0qxwc5m9LNrF6OBS9N6S7DncIYt9VupI5OCr1uRSqzqaBMFDClTTH+dAx3b6J
1KFB0UiHP3FeTalFh8L3NE+dN9apNAgkUWv/v4oo/6dkRu3NZse2RAo/o2X5r40q
k/lhydQRZTSTFsiuH3VUWVsgmqAHnHW7pMMw8FAlKhyRSFnhbW7re0jj8XMIO7G5
yjQKQCnYuPdXbx++bP1PzsEWDv9j/sph5arcosdo6tEXklWHED17MEPIton1+NRf
sU0peEVggQXlwdTcZN/h7FeCZ56dcwCWdCpSlv6CcWzRXSNUyJpKa9qfIqBX/mon
jy7w5IHmhvLwAYI6IoT11h1QDEfGfhrwWPwOjnXsaYm5E7wv8w69PxMbOJbMpWSg
8L7xW3LXKR1VwXggUC1+b3y67E5Ggi1hf0lfTnTMpL2ClAkQ2a0qGAV0dMsNxRAA
suW1aLh+hgydW+iH6DmdQRMESsB1kE02kO1462TAQaziIAvNoxw5h48xvyEnrDA8
d+9IDMyxdrLmAbndUlSveMa9+EPiGHwr6VTyFL8nA5F7DcFi4mjEyGKe18JcaAlY
UtvHgWH6EjiX2iSXpsrJFEhtfFNolZ5sp9LFI6hOBihSJxZK4sbMR7Q6IkDuAVpT
FLiejBRlsXpFvTGL6040CtxbL5cqkVMYP38rFMTuc3pGGJA4wb5EC1dGjUi6XjbY
H7kuCAFyXqV9eQQP61x7K9W8qnXW+weCIMKfSX7AcCtH1jXBAM6lqpPrh6amc+/r
bg2eNA7DmgJnEY4apIcDB/b4khRMga2ozeGWWyIvOaVvR2R7ALQ+Rgut85cM+4+V
l2PHmOzW/yYdHVb5REQItFR5COb/mGUqYhkCtiV3nXo/K0uOQKu5SBbNzLuNvuwd
n+Eimxjl18VnrGG7sjtUa0MLmtr62GiEVrhrDqa/biHp8LdWkAQjLZ4aTRh2XZig
gaVFZHmkw3ILPyKKM21UXdM0YRk3TGVK8ODQy58ebPS4v9yYT9gUA9UDkDYeGcF2
qjoDPVNvcG6H8jCSsPRl1KZwtqqITCOSAIAPI4Nu97kO6nbOyQpYlwjd1MhvVXnP
66mHSvmqaxbNGX1mF9B/yERkBkooNZrKuJSvBTC2J1q5Ag0EY5V3BwEQAMpFVczZ
o9ZPNsgW791UW5o6wnrnd1nIO+S4rc37q2TEz8KGHCuxo5NwffZ2t6Ln04BI54pb
apg17b7a0hPka37HFkL28n4VyMdx0CsAm3QEfUsdK6xwKV2SucYeVcrV1upcN4Pd
XD7su1I7/A4CWXFJG047zJ0Z89lJZiQEiAq7ghvEoinC0sm+0a6ao/ocqCgWCKM1
```

```
yCPOyzJXleRrv29SRnYziMR+q2U0x9xg9Xl6GMwUmFwbJc9nORVvLH7fbU6/du8E
goAYrglFOFZG/TSolSGWRSMiavz0JSD/i+rEN4aIT4WfBe+L9Wy1AmrNxiAO+zKm
zHQu3JSxDncr+y+hcd+W0gqw10FoI9jWLcL7kR+6a0iOjuJSXSopq2l3DafiPxtC
Fmr4CGQhzBHM6e4/v/NNd3F0XpVbJ6RQph7lkfvfz8q2lvUlHhezJ0p1xXmhff9C
HjdVMhmAmz5+imBAXk2mottNfKb0pFEen1xY3K/UPA4g+oPsSj495MsvIg9eIMCc
C3/z0SEUMWH/styyJzPqfpyfGwZeTcIj9vg2o+RnGvmcLVYA/EGToPk905kv/cK7
3oy8bZyOB0zMg7T9PaWgLUO0sqjqo0Mw3knFySg3oRXlcilPQvfPdX0JvwLpc9DW
lr1+1GkCXJ08lWugJc96CJQupKRb1IbC0oUXABEBAAGJAjwEGAEKACYWIQQL4zJ1
10yVPHn4EQfZrSoYBXR0ywUCY5V3BwIbDAUJAgIpAAAKCRDZrSoYBXR0ywwtD/wI
DmEcHdFlyFRTomUBjbeK2uzcZIhkkgL58lc63UPle5iJ2FBvmYS+0rQS53sVEscc
n5KfkOwTryKllvWbl0IzuiqfawxALcfWpfZJHzTMSnDHfgXvOOyFMQruqRDAHAr7
PNC0CnbT0sEF2ZFzad8M9fLqtkXUx4mgECNGJ4CVqg75KY8uUzv/BmRwEf587FT5
/iAIed5MjFB2VFDX9GABcvTTbHxCZIxnxl3cs15SxT0lAofZ2ueU6kWYWZSXFeaE
M/4ymPJws2mmV0AkbJghLXCn9Mx3nX6NTZZ9Harbru+RzW3/Hg3DZd0J9vko8Paf
P0l1NWtgyX74CqvTgjzTxXTnqrRXzcczK7fhcC2u4i0prPtXXcyyi7SwpoLikaZC
LFFhUmOx+mS5TjtgFyFZBNxnO7iAwkzfcTcC9sPoWaFmiQf6q5EIYzG+WQpncj80
mxl3HWOP6oFj/hZJRYseKeMkvJzLTo87rFdM6CsMrLwETR6e+aWM0btPFil1rXVA
CNOjsy0bxTV8OJEfyxnYmyjvnBvB0kdiaVEDdVhxgSqzLAX4mgXa49/V6M/uzMr+
n3/A1Jdk4V6fVm8S5cFIXxoUat3cB4xGaT9OWD3o1NPr6eS9Vo0EsJlRl81SG68f
S+Qtk2fX27T68YG4Aa3zMfZxUsVuFLtTuQbRC+fJpIkCPAQYAQoAJgIbDBYhBAvj
MnXXTJU8efgRB9mtKhgFdHTLBQJlhcuqBQkD8n1oAAoJENmtKhgFdHTLoO0QAJsT
E9fkleb7YzPEuP9GJ3jx8PGdWm7n+8UNdr24kS6gOXVUfPZrWa5So21hcIwZb4PZ
DqHSVSQnRciKhSnG7gplYPNGZ4+FWbLr/mBRYarjkVFLUuCPexSIjxV1KSGJnWs9
YTVAKZAz75GpCML6jD6biCOQCQ86wqOdWvZIZR8YvurrxR64ABB0rjbsaG8cNOUX
1cwAfdLwthf64dS+2m3lqNGDHkP5eNL0RIxC5gXYEp0lvmlMH3ZuO5WrfH73PTDg
89bxXeuhrFmSEwf4xWm603oi8/2qQvR9/7jb0o+t71NQuWrWIFONZWWgZBUGso+u
yT3XgY4YqKGR3z2QzKHYnJ6M7SvSYpqS7RtcxcCXF0HGNfES8cAgtKVpFtbtSwXX
p8O8oLyjmVIO/NjUpbLOGdFIsarsezLFV9f2fqZ63J34hyUSg8LrYVV1fA5DJUpe
bbX4hLpdk0MMtgG43BwKIGlJTpL5RkQ/uQU3YW2kairy7o+1imDD0TRzQxtdjVOI
5vnlTNcfJZIIfLx4drABA12OvpX3dfPV62R+8BAlJFT430CG6AISJIBqJRFvuikm
nZGUvEHmOUs/FLbbaXTPkKc7tR2WIwljRvMV+Qk84cWcX6YchMslMuiDM1mtlQZi
g34WHGSE+zCWnXAslIHlSwox7qfdO0Kz2XncSbIAiQI8BBgBCgAmAhsMFiEEC+My
dddMlTx5+BEH2a0qGAV0dMsFAmeEQS4FCQXT8gIACgkQ2a0qGAV0dMsm1Q//V09x
OutSbWU44KRurdnGKsk56DFlqXtjGYJqDPrODpX3M8IDf2MuTIN2yfPMv984bAbO
A9RL7EaGVlQUW9QWPURMsZKEFQljhfxRJO9JoGDYI7uRDnSEi6WjVvgUk5OiIh0K
EI4jEcaLCzveIEcswrVDSAn+7nGvewP8Rrx7qMUNvLAltxiMyfGXneavRs3sfusz
db9LTTY8lCU0xrslaXrrvfCkaRbskFi3S31I+1ZB/ewuAhHqfc13eRBjPwQOanJF
epAzP4GF41fVQN9GtssATCD+dV6OFhYjfWJbOIcPv277wCvGIFucM9XRjbkCIYFQ
E5W+1O/act3Obj1sB90C+cVOgCng37YqfObYLF19RE4+a7NUAh/GxHj+8TxUyvvr
aWWyfNqTMDjHMShNDjG0qSzFX7vfyUpmfAfz+6ad78aks9LMf+86iGkvBhXFs7cz
Vv4PWWYV1+WShNU2Y9yMaH3zpWUaREdB07HKbLva4Y1icqWVx+z6xs3PvsqbTei/
moXiZk7ohpbBm0htJlki22ARYrGXSK6w5RQtCZoBW0DEj5JNBjkK6XbAW3VFuAA1
J5wLS2z0eIR5adP+/SxQUbTq+ZFiOGdBP1g/783e7zEdyA0YfA2KU9OdLzupUix/
x+JYcxmrZXndSMObd0IiWOhwXlgarbJJMReOORg=
=cYaK
-----END PGP PUBLIC KEY BLOCK-----
```

### D.1.2. Гол багийн нарийн бичиг <core-secretary@FreeBSD.org>

```
pub   rsa4096/4D632518C3546B05 2024-02-17 [SC] [expires: 2028-02-08]
      Key fingerprint = 1A23 6A92 528D 00DD 7965  76FE 4D63 2518 C354 6B05
uid                           FreeBSD Core Team Secretary <core-
secretary@FreeBSD.org>
sub   rsa4096/CABFDE12CA516ED2 2024-02-17 [E] [expires: 2028-02-08]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGXQ1o8BEAC+Rcg8cmVxuP17Vu+q5KgCx/XiulQuqKXAqqBlYCH2jqk6DINP
yFrREGBhzd/qNmlAYEahQ4Zgl0bUZNTrZVDyzicOvPP0jH+KSTQwRs7NOawEdlVO
cyHrwDCPEqf5ZzD4NhfTriEOw+j0pEH/onitUGvoQRtx15xWyaJQxDEBMTYMLewE
86D1bltwnTNczE3UZb7oQLJXkAX5hcLtou70XJGgZITvJkK+kp/xot2eFjnqRz/u
WeXnKhYAmC07EKwZ1uw047eHKwMMRBYqzApLwoQtfE430Kxf2q8de64x8zDbi6YM
1J4r8OAxOtHVyfJ0j7Q23DEZz0VVb4b1Tx5OG2Re/KSNvqI0awJO4TcRmOR88OyY
dzyXgnX6Sa7GVQY1FXvn7vtFuDAt7egZOzeomSHL9bdX07LTQ4UtM88EV9wm3q4q
smoatV9jsvPQ1zxCU3aQD/5eWTJH2/kz1LIuBL/Qi5XQpJn91lBtUWJrCgkHWPGu
f//rnnXmsG7DAChW+yZ7cFO8lfNa8sFhPqSxCYphWmJTrvadyQtDngB8JakWdnmK
pfGS6y5lel+181vw38ZZKt04AKM+nDY8051lBM7Q9Q6kTLI33UZeImndx5xYukVD
kV6aQ31HYfEark15c7iEz+OAcwFnM2ntXMt7kKGd40CqzusiPcQkPqPbAQARAQAB
tDhGcmVlQlNEIENvcmUgVGVhbSBTZWNyZXRhcnkgPGNvcmUtc2VjcmV0YXJ5QEZy
ZWVCU0Qub3JnPokCVwQTAQoAQQIbAwgLCQ0IDAcLAwUVCgkICwUWAwIBAAIeBQIX
gBYhBBojapJSjQDdeWV2/k1jJRjDVGsFBQJnp8PiBQkHeofTAAoJEE1jJRjDVGsF
GMIQALhj+mNpH8OmTFeihQ6t9P8un3llz6Wmqe/Q+ULWeqJvV/uC1J5T9fnoGhwF
MgECuguXYJtoYQ16KXnsSOs1tcqMOK9GtEtFJTGe2DtfLiBednwUvu9j3HmTlwLN
M+7rqkiC0HCg2qSjcMjxbVbA5BSgNkgfyTSO2YdjfaZ+ceiHwo/qa5xWE6i2dMR1
PLGMMHTeELdtdSH6VR9/3h0qt3qzwdMBRCVAQHbim7CqwjUH9jg+IOySXl81jNoB
xuVZR3pKshyY4Oxo1dK+W86Uiff/+c4jCAxokGWIR7C4MkZZWUQqV7920gkZFC/5
qzpT1A1/sFUg8HfFTr0vCoPSVFWn2+Tto3vr78DICVaAf02aYAFlyKK18BMCoHkS
hDDO+/JQZmvHOIgEYK+T2WN1c9gm9IDJzGZuH0X2C/vkREnJKkccJfB4pXuN10wt
fqyP9fn8h6+/t+sv3qNLm4d8fkmLXofulG3WB4i/F+Hip2rjPvvBCF7zl4xy6cJ2
xVY5HUOBTqmIwVhYwUpXaqNoWa/qJBLTuot3z7ciKmKX6Lq+Dze5dzhrPNl/CalC
HBf3miHRK3TZbYLooG3bcEWgxU2BnBi3vl5NpCoUOKkPYRlALXi2TyHmPxO7oT4e
mIzS6BgnXOqO+AXvbKKfayuSePdBqkNMK/SMC4Dylkf6Xj25iQIzBBABCgAdFiEE
EBpxaxYrAOVb7eoFrbv4YQo3ibcFAmbu7x4ACgkQrbv4YQo3ibcr5A//TIcbD/EW
YJz0zrUQdc9xG3UNfU6uHmQzAuUy5ginevyqv0TSso5qvSkOHvdxbi41rfMiB2RJ
V3q0n0PSvHFld89fOTZUMZXaPvozCiBYWScrt+KA/2pq3K8mUumHS+IFtpHLL2Tu
+gI8XCHUFxO9HUThM/rFlIyEdzoctgmqQ7IG4uZG+o2J3w09llhDAUe0vraJK1On
p9yFACnRrhqvl41JeUWcv9MH9JcwHUqtUo9WLTCIb+hkByTOyRfHBYpYw//bdXdf
6rkCwKVWpyMDbk61zq2VsS1kqbP9IH/A8CsBnA6mg+zPq2i7HIFw8Swj4OGJcIvG
a9ubUYJRjDhX/vBpNrtncANZ88FQmA+Maq0vu0LS5IIGyIKkvd1fKIsyvBDDa9kE
nfCW0XMkJA0Gf+kxdB+elXQHBwKO2sr5BiKnJT5lJq3Yu9fxxGBnf93yiN4E9bmF
gG7cZxpyb1Bp76TJhLcANyybOTjCtiNRrgqeaSx9/6hSPfPigGXIne0H2lmJO6oq
jUrsYmFiWvU9sc7AcPVw/eHG8FgW35TuwKX71z8w994iaahUPNcSVyXOUD+QNROv
HhGUXrc81t613rivh22N0NZpNubVatq43KV7+/bnPyWBIi1Awh3vIFsNNSQSrYxF
lUuQaAHQXTeZMZ/7npE4t86seMt0T7BGn765Ag0EZdDWjwEQAL3VwFifpnRCYzQ4
VZ1dAjp8w542iRprqeA+C3tvNbk2OvKpN3DIc49l2bgNZ/VI7/T58lEKrFsgLK4w
AWtv18OV7xuh0AuOblmq5MvcPrUelkPj5HA7M6Ng/rAubuHfwdP/IjIrzzY6+XDh
fP9N5KIv9VRJYT23BbvLPeit4J4tQEB/NpxL3L6zI75qq4R3T/EkHP6Y9Buup9Lr
isJlcxkSK+CyORCgTpEz6fWsXiTDgS7cTaQ969XCygBpj4wRQzwkBdokxo1wsiFt
```

```
4zLt07/PrPYjeHltTDkrF9XDNhLNJ5GliHnd01oHg1j5/n+9Osvh1maoFwuBxXTN
nTZ2P+7RKLBaVQVSkUa5KJbXoM3v7bMbXM5aLs2XjglrAzrZOV+Y7zOu5UYQdpXd
7x8XAEtEozRJzt7dosQIhKx9h4LlltFFulDUpf5VCvcUEoAzMbiX0aju9n5Rwsql
DHatU9Mm3W8OdFXj1foIXZD+VX2Jp9DgxPLoAJ0CWhPXJ8f+WFSJZCjWoPJEJKWY
0EOxiXyAuvAyniAZAC/eKZkfGckXmu7edRgYbRTTwPmZ/axa/k9aHsHLwmbxuZo/
xxQXzqU70EleHZ31A3mOqsC1epjN6dn4AFKgvVesP/K/fbvSsfsfiABS2A/68ne4
zJG73Tnk4L6J79vrXc2iMJbmdg3ZABEBAAGJAjwEGAEKACYCGwwWIQQaI2qSUo0A
3Xlldv5NYyUYw1RrBQUCZ6fEFQUJB3qIBgAKCRBNYyUYw1RrBQd9EACMVckxzy4w
aUGlERguJ+kslS8MkMjNqnfDPLRDVxbUxNvbbhw7/u9blM65DCGCeNLc2n4oiu5C
E3I095AKmvq/0dOa81mEEdZkC1CVc64bXWbEyz5AtSHUpgdxRso6C+YopndSiz1T
WcIagQRXfWaw3FBWPooA87gmibmSmCegCtqx+uyc5QxX2eFI8mRK7vlfnGpYKHs0
D1/yUSGQOwoNRJ5FYm+ynfDE3FzHEQL7lv1vpv1k3xNKfBziMMg4IMEBKNHV4VKN
qJpa8UCodeTGSWQdNnCeCWPsxz5oQjCcVH9Z3e8sMpWLHhRcBZzSwXUOws2GbRMH
xHwrfrpHJcrBhe64pgfGOvZLUJ9BDs+8egTHsqRFacipbtTR+hhVhuJEHDaQQWuM
8IRHj9HIuTAczET8JTDHTMIoo8DZdOtiW/YgqCDwYghkI77d12oNQqYoeJ2HiqbK
cGzwCpsR0A+p/iOAxJG13tsxqZV8TQ8iTokWG6ACtZ7sfeWEhxqMbUKUMgogZn0I
3n1kV+tUZC6BQrIYI7TiKg95wLZsIydeoIsQoNZwvyKAXfVmQ62YjIX8njZwN+07
8/ipUPJxCYa8zL8BZyDmoFJqa3y9z+11+vtiZ9t+aTwGvjpHDwyeCJco7go9cU3m
GRFZYciqIoG4n3tl8Pob15vFlVqk47rRqg==
=8TzT
-----END PGP PUBLIC KEY BLOCK-----
```

### D.1.3. Портын Удирдлагын Багийн Нарийн бичиг <portmgr-secretary@FreeBSD.org>

```
pub   ed25519/E3C401F60D709D59 2023-03-06 [SC] [expires: 2027-03-05]
      Key fingerprint = BED4 A1D3 6555 B681 2E9F  ABDA E3C4 01F6 0D70 9D59
uid                           FreeBSD Ports Management Team Secretary <portmgr-
secretary@FreeBSD.org>
sub   cv25519/2C92B55E27A641C3 2023-03-06 [E] [expires: 2027-03-05]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mDMEZAXJvxYJKwYBBAHaRw8BAQdASFAC20WL3R1T6uNyGMZbfJCxDkcP4C5vi3Op
tcZ2fbq0R0ZyZWVCU0QgUG9ydHMgTWFuYWdlbWVudCBUZWFtIFNlY3JldGFyeSA8
cG9ydG1nci1zZWNyZXRhcnlARnJlZUJTRC5vcmc+iJYEExYKAD4WIQS+1KHTZVW2
gS6fq9rjxAH2DXCdWQUCZAXJvwIbAwUJB4TOAAULCQgHAwUVCgkICwUWAwIBAAIe
BQIXgAAKCRDjxAH2DXCdWYN1AP43TjyfZtZ3DLYT++g0+SuPsoO/3yWVybA+UmFL
zb8MngEA+LLNUfvEwCuXS/soh+ww5bpfmi3UUmeGiQEAXug3iA+JATMEEAEKAB0W
IQT7N0XIbxXo7ayBMvzYKU7Du8TX1QUCZAXLkwAKCRDYKU7Du8TX1XHMB/9R1MX4
6zMgpKqPPt76GOI+eGEdBK6bY8aJZjQGdqTh9f6VtXVoTGIG7cvhc9X8tDBoB0PT
2KZWheF51AV1+NHU4HwLAQ1BMebrFvWSfkw4xg4fBGwDhz9/GN85No+Js772V5ey
8lRiL6meRVWxMlLyWcxGd8JjcC5yX/iAUQ3SBGCLqW7unWjjg7CTd+AMBwcqPGrv
ax8q6eFVguJcHJAjMnKf6HAy4cpK3s+uMoUBCGnszSN12B3ysKfyC4pNO/pix5tA
Q5v8aRqTeFPh5zmNhWo0KGPzplTPqRQSHDl7GDQC8Ru3MhzFkeWzHsexjZVwS6W2
DPcYpuuAsA0XOZIZiQIzBBABCgAdFiEEEBpxaxYrAOVb7eoFrbv4YQo3ibcFAmQF
0u0ACgkQrbv4YQo3ibccwg/9F2Xuic3nhKxRbB3mJeDo6SYQETa/Gh1qQ34+8zlt
8UMazOx67gnYQfy+pXjro6eQ2up0a4eUYezcNOudqAQD21nRz3HA6EQVNcE/TzEA
```

```
xl5CJntTaLOt7S+EDXFW5BuQIvhhoMGgm8+WNVgA0EJ7tfL0OcYBSvr19fqwChEn
9c14cSk6mgHSsleP5NvskYN053pxHwy0LTSb8YBBv52th37t/CRFC1363rS5q+D7
JixFopd1O5pKpA5ipvE4gGgRjPtwjx0SjjepwK/3fuhEJQQyKzTIKlMfu2Dj/iR2
Li1Sfccau5LQXOj9fUITU3u1YG7yrm8VGzT7ao4d+KRwgMLjd2pLqiGIbbJwGBiP
FRmtilWQoeIlmSlFX4obAA517DOK0pW1mH8+eEn4EJd3SekT3yzFyKTASv0J48Z8
3F928xg+eZvHxVC0t1J+J5IG0gt3EEncuWKIPQGR7PiQbti6R3FQVTz6WfMWOebP
Qi0E9F/Aqakr6Vj2sKGrDq+ebpaF5G8Yw1YrUl2IDiPzkCegp3ZbI0wh11Xvzhi8
LXPQGK4jBQas4G8cegfitzmtdGRHYrbMv0R9I4mvaL+WlOuD2AvyVG28lguqVhnN
AZP+ohdquYyX2CNCVvbKWAtXo6Ur0vWG8BL8m6defAtEkIwVBALaOHQOSI3aNUz4
lwy4OARkBcm/EgorBgEEAZdVAQUBAQdAsefmSfxEOdOr02+K/6noYCuJ1FeAWVz6
jFYQ+9w6jggDAQgHiH4EGBYKACYWIQS+1KHTZVW2gS6fq9rjxAH2DXCdWQUCZAXJ
vwIbDAUJB4TOAAAKCRDjxAH2DXCdWRl4AP9h5ot212BK29S6ZcMBhHvmtF5PG1oD
c7LnZycSRmbFiwEAndCMpAGOhDW8iVgDd0wLQq/ZMPe+xccfG1b3zFH2EgE=
=iiAT
-----END PGP PUBLIC KEY BLOCK-----
```

### D.1.4. <doceng-secretary@FreeBSD.org>

```
pub   rsa2048/E1C03580AEB45E58 2019-10-31 [SC] [expires: 2022-10-30]
      Key fingerprint = F24D 7B32 B864 625E 5541  A0E4 E1C0 3580 AEB4 5E58
uid                           FreeBSD Doceng Team Secretary <doceng-
secretary@freebsd.org>
sub   rsa2048/9EA8D713509472FC 2019-10-31 [E] [expires: 2022-10-30]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQENBF27FFcBCADeoSsIgyQUY8vREwkTikwFFlNg31MVy5s/Nq1cNK1PRfRMnprS
yfB62KqbYuz16bmQKaA9zHN4FGfiTvR6tl66LVHm1s/5HPiLv8sP14GsruLro9zN
v72dO7a9i68bMw+jarPOnu9dGiDFEI0dACOkdCGEYKEUapQeNpmWRrQ46BeXyFwF
JcNx76bJJUkwk6fWC0W63D762e6lCEX6ndoaPjjLBnFvtx13heNGUc8RukBwe2mA
U5pSGHj47J05bdWiRSwZaXa8PcW+20zTWaP755w7zWe4h60GANY7OsT9nuOqsioJ
QonxTrJuZweKRV8fNQ1EfDws3HZr7/7iXvO3ABEBAAG0PEZyZWVCU0QgRG9jZW5n
IFRlYW0gU2VjcmV0YXJ5IDxkb2Nlbmctc2VjcmV0YXJ5QGZyZWVic2Qub3JnPokB
VAQTAQoAPhYhBPJNezK4ZGJeVUGg5OHANYCutF5YBQJduxRXAhsDBQkFo5qABQsJ
CAcDBRUKCQgLBRYDAgEAAh4BAheAAAoJEOHANYCutF5YB2IIALw+EPYmOz9qlqIn
oTFmk/5MrcdzC5iLEfxubbF6TopDWsWPiOh5mAuvfEmROSGf6ctvdYe9UtQV3VNY
KeeyskeFrIBOFo2KG/dFqKPAWef6IfhbW3HWDWo5uOBg01jHzQ/pB1n6SMKiXfsM
idL9wN+UQKxF3Y7S/bVrZTV0isRUolO9+8kQeSYT/NMojVM0H2fWrTP/TaNEW4fY
JBDAl5hsktzdl8sdbNqdC0GiX3xb4GvgVzGGQELagsxjfuXk6PfOyn6Wx2d+yRcI
FrKojmhihBp5VGFQkntBIXQkaW0xhW+WBGxwXdaAl0drQlZ3W+edgdOl705x73kf
Uw3Fh2a5AQ0EXbsUVwEIANEPAsltM4vFj2pi5xEuHEcZIrIX/ZJhoaBtZkqvkB+H
4pu3/eQHK5hg0Dw12ugffPMz8mi57iGNI9TXd8ZYMJxAdvEZSDHCKZTX9G+FcxWa
/AzKNiG25uSISzz7rMB/lV1gofCdGtpHFRFTiNxFcoacugTdlYDiscgJZMJSg/hC
GXBdEKXR5WRAgAGandcL8llCToOt1lZEOkd5vJM861w6evgDhAZ2HGhRuG8/NDxG
r4UtlnYGUCFof/Q4oPNbDJzmZXF+8OQyTNcEpVD3leEOWG1Uv5XWS2XKVHcHZZ++
ISo/B5Q6Oi3SJFCVV9f+g09YF+PgfP/mVMBgif2fT20AEQEAAYkBPAQYAQoAJhYh
BPJNezK4ZGJeVUGg5OHANYCutF5YBQJduxRXAhsMBQkFo5qAAAoJEOHANYCutF5Y
kecIAMTh2VHQqjXHTszQMsy3NjiTVVITI3z+pzY0u2EYmLytXQ2pZMzLHMcklmub
```

```
5po0X4EvL6bZiJcLMI2mSrOs0Gp8P3hyMI40IkqoLMp7VA2LFlPgIJ7K5W4oVwf8
khY6lw7qg2l69APm/MM3xAyiL4p6MU8tpvWg5AncZ6lxyy27rxVflzEtCrKQuG/a
oVaOlMjH3uxvOK6IIxlhvWD0nKs/e2h2HIAZ+ILE6ytS5ZEg2GXuigoQZdEnv71L
xyvE9JANwGZLkDxnS5pgN2ikfkQYlFpJEkrNTQleCOHIIIp8vgJngEaP51xOIbQM
CiG/y3cmKQ/ZfH7BBvlZVtZKQsI=
=MQKT
-----END PGP PUBLIC KEY BLOCK-----
```