redhat.

Samba –
a group policy engine

Günther Deschner
`gd@samba.org`

(Red Hat / Samba Team)

Agenda

- What is Group Policy ?
- Why should we bother with Group Policy ?
- Samba GP engine components
- Samba GP consumers
- Discussion

What is Group Policy?

- CIFS, LDAP, Registry
- GPO (LDAP)
- GPT (CIFS)
- Engine
- Client side extensions (CSE)
- mmc snapins
- Tools

Samba & Group Policy

- Primary motivation: `winbind` enables AD integration but has no support for Group Policy
- There are many Group Policy engines for Unix around – just none of them is open source
- Samba / `winbind` has all technology on board to do that itself
- Experiment: provide a group policy framework based on Samba with an open architecture
- Main obstacle: there is no – established - centralized configuration framework on Unix

Samba & Group Policy

- Architecture & Design:
- What from Group Policy do we need, what can we use ?
  - We can't say yet
- Where do we “store” Group Policy ?
  - For now: like Windows, in the local registry
- Who will be the consumers of Group Policy ?
  - For now: tools from the Samba suite
- Basic approach: observe Windows registry modifications to learn inner workings of group policy

Samba Group Policy components

- `libgpo/`
  - `ads_get_gpo_list()`
  - `check_refresh_gpo_list()`
  - `gpo_process_gpo_list()`
- `net ads gpo` commands
- `winbind` async group policy child (with `smbcontrol` signaling, `winbind` queries)
  - `winbind` is processing Group Policy for the machine account
  - `winbind` is also processing user-based Group Policy when using `pam_winbind`
- Client side extensions modular API: `gpext`
- Group Policy backends: `gpdb`

Client side extensions: gpext

- Stored below `/usr/lib{64}/samba/gpext`
- Will autoregister by GUID in local Samba registry below:

  `HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\GPExtensions`

- CSE module configuration stored in registry itself:
  - DllName, ProcessGroupPolicy, NoMachinePolicy, NoUserPolicy, NoSlowLink, NoBackgroundPolicy, NoGPOListChanges, PerUserLocalSettings, RequiresSuccessfulRegistry, EnableAsynchronousProcessing, ExtensionDebugLevel

Client side extensions: Existing modules

- `registry.so {35378EAC-683F-11D2-A89A-00C04FBBCFA2}`
  - parses PReg files (`Registry.pol`)
  - applies them to local Samba registry
- `security.so {827D319E-6EAC-11D2-A4EA-00C04F79F83A}`
  - parses ini files (`GptTmpl.inf`)
  - applies them to the local Samba registry (under `HKLM\SOFTWARE\Samba\Group Policy`)
- `scripts.so {42B5FAAE-6536-11D2-AE5A-0000F87571E3}`
  - parses ini files (`scripts.ini`)
  - applies the script info to the local Samba registry

Group Policy Backends: gpdb

- `registry.so` (default)
  difficult to access (outside of samba)
- `elektra.so`
  nice framework but unfortunately totally unused
- `gconf.so`
  complex setup (schema), has no “system” specific part

Samba Group Policy consumers

- `smbd`
  - Queries registry to retrieve privileges (merged with local privileges from the tdb)
- `pam_winbind`
  - During interactive logon, winbindd retrieves warning attributes (days before pwd expiry warning) from registry. pam_winbind displays them.
- Future consumers:
  - Samba3:
    Restricted groups?
    Kerberos Policy (for `/var/lib/samba/smb_krb5/krb5.conf.$DOMAIN`)
  - Wine as registry consumer ?

Samba Group Policy future

- Samba4
- Use open architecture of Administrative Templates for Unix settings

Thank you for your attention!