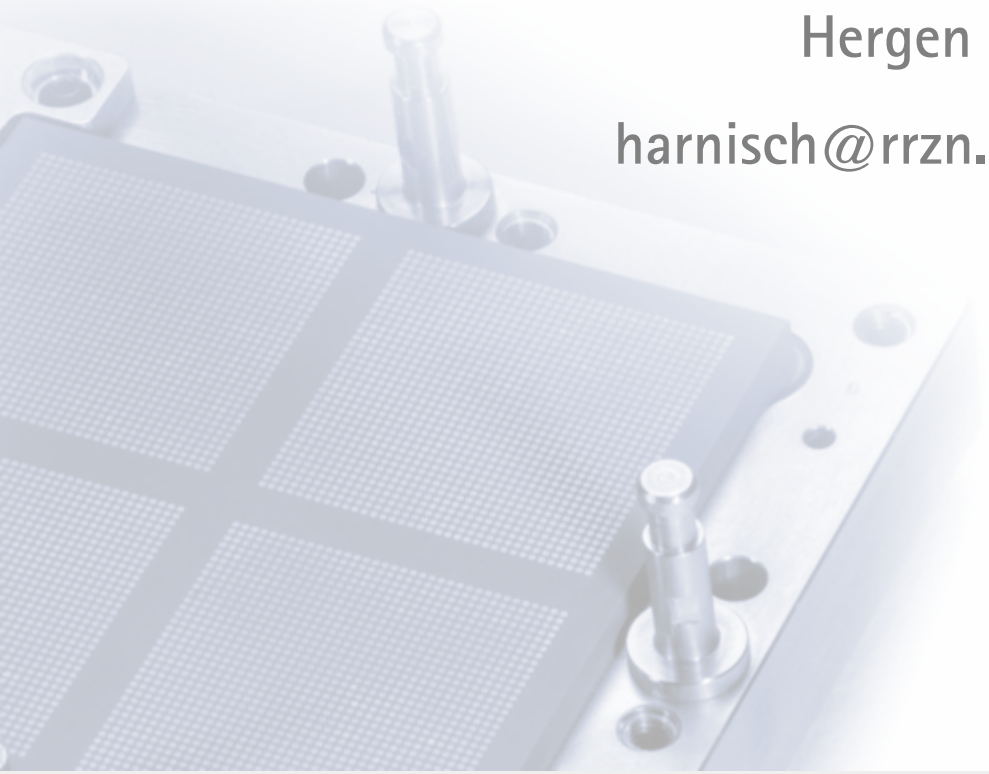
Windows-SMB & Samba



Hergen Harnisch

harnisch@rrzn.uni-hannover.de

Vortragsziele

- Grundlagen Windows-Dateisysteme
  absichern gegen unsichere Altlasten
- Unix als Dateisystem-Server (Samba) und -Client (auch andere)
- Überblick: Samba-Funktionalitäten und -Möglichkeiten

unberücksichtigt bleiben

- Implementations- und Konfigurationsdetails
- Drucken, IPC/RPC, Zusammenspiel Windows-Tools & Samba

1. Windows-Netzwerk
2. Unix als Client
3. Unix als Server:
   - Allgemeines
   - Anwendungsszenarien
     - Workgroup/Standalone
     - Windows-ADS-Domäne
     - Samba-NT-Domäne
   - Sicherheit
4. Schlussbemerkungen:
   - Ausblick
   - Fazit

Netbios, smb, cifs

NetBIOS Network Basic Input Output System
Layer-6 Netzwerkprotokoll

wins Windows Internet Naming Service
zentrales dynamisches Namensverzeichnis, nötig für Netbios-Verbünde über Subnetzgrenzen, verhindert Broadcasts

smb Server Message Block
Layer-7, Netzwerkdateisystem, basiert zwingend auf Netbios

cifs Common Internet File System
Weiterentwicklung von smb, RPC- und NT-Domänenunterstützung, ab Win2k auch ohne Netbios einsetzbar

Netbios über TCP/IP

Ports

UDP/TCP	42	WINS-Replication
UDP	137	Name Service Broadcast oder Wins
UDP	138	Datagram Service Broad- und Unicast (eher unwichtig)
TCP	139	Session Service Unicast, verwendet von smb

Wenn NetBIOS noch nötig sein sollte:

- 1-2 WINS-Server einsetzen, um Broadcast zu vermeiden und Namensauflösung zu gewährleisten
- DNS-Hostname = Netbios-Name, um Namenskonflikte zu vermeiden

ohne NetBIOS

- in Win2k nicht standardmäßig, aber konfigurierbar;
  ab WinXP der Standard
- TCP-445 (Microsoft-DS) statt TCP-139 (Session-Service)
- WINS ersetzt durch (evt. dynamisches) DNS[1]
- SMB2 in Vista/2008: robuster, performanter, Balast abgeworfen (z.B. nur noch Unicode)

Wird von Samba unterstützt und sollte möglichst verwendet werden.
Daher in `smb.conf`:

```
disable netbios = yes
```

Wenn Samba PDC/BDC ist, dann aber `lmhosts`-Datei auf Clients.

```
smb ports = 445
```

aber nur, wenn Samba kein PDC/BDC (genauer: unklar, da client-abhängig).

[1] kann Namensraum unabhängig vom „offiziellen DNS" sein, z.B `*.intern`.

spezielle Shares

`C$` und analog für jedes Laufwerk
verwendet für Administration und Backup

`ADMIN$` %SYSTEMROOT%, also `C:\WINDOWS` o.Ä.
verwendet für Remote-Administration

`IPC$` Inter-Process Communications (anonymer Zugriff)
RPC über Named Pipes
auch unter Samba dafür nötig

`NETLOGON` verwendet vom Logon-Service
Speicherort von Polices/GPO & NT-Logonscripts, Default-Profile
wenn Samba als PDC/BDC

`SYSVOL` seit W2k Speicherort Logonscripts
da Samba nur NT-Funktionalität: in Samba uninteressant

`PRINT$` Drucker-Treiber
beim Drucken über Samba (vgl. SiTa WS06/07)

Security

Logon auf einen Server für einen Share geschieht

- mit 1-2 Passwörtern (ro/rw) je Share (share level security)
  veraltet, passt zu Windows 9x mit FAT-Dateisystem
  in samba `smb.conf: security = share`
- mit Username/Passwort (user level security)
  neuer, sinnvoll bei Dateirechten wie bei NTFS
  in samba `smb.conf: security = user`

Samba

- Share-Level höchstens noch sinnvoll bei reinen Gast-Share-Servern (z.B. Read-Only-Shares mit Software zur Installation)
- Gast-Shares auch im User-Level möglich, flexibler
- zudem `security=ads`, entspricht `=user` in AD-Domäne

Authentifizierungs-Methode

Vier Methoden der Passwort-Prüfung

- Klartext-Passwörter ⟵ indiskutabel
- Challenge-Response mit Hash
  - LM ⟵ völlig unsicher
  - NTLM
  - NTLMv2 ⟵ Wörterbuchattacken schwer, vorzuziehen
- Kerberos ⟵ geht nur in AD-Domäne

Kritik/Bewertung

- Hash hat Passwort-Charakter, wenn Challenge-Response
- LM case-insensitiv, zerlegt Geheimnis in Teile vor Hash ⇝ unsicher

sichere Authentifizierung

Problem

Bei Challenge-Response werden aus Kompatibilitätsgründen standardmäßig mehrere Hashes übermittelt, auch LM!

`LMCompatibilityLevel`: „LAN Manager authentication level“
in Gruppenrichtlinien setzen auf 4 oder 5, d.h.

- Client sendet nur NTLMv2 (und LMv2)
- Server akzeptiert NTLMv2 (und LMv2), bei 4 auch NTLM

Bei Samba in `smb.conf` (für Server, Client-Einstellungen ähnlich)

`lanman auth = no` LM-Hash nicht zulassen (d.h. kein Win95/98, Dos)

`ntlm auth = no` NTLM-Hash nicht zulassen (d.h. kein Win-NT4 mit SP<4)

Unix als Client

Linux

- Kernel-Modul smbfs: veraltete Implementierung,
- Kernel-Modul cifs(-vfs): Unterstützung bis NTLMv2 und Kerberos unterstützt CIFS-Unix-Erweiterungen von Samba-Servern
- samba-Utilities: im Userspace, auch grafische Erweiterungen unter Verwendung von Samba-Bibliotheken, „Share-Browser“

Mac-OS-X

- smbfs-Implementierung aus BSD übernommen
- Signing vor Leopard nicht unterstützt; GP Änderung auf Windows-Seite: „Microsoft-Netzwerk(Server): Kommunikation digital signieren“

Linux im AD-Verbund

- Anbindung der Nutzerverwaltung ans AD möglich,
  ähnlich wie bei nis oder ldap über nsswitch
- Schwierigkeit ist Ergänzung um Unix-spezifische Daten,
  z.B. `uid`, Homeverzeichnis
- Offline-Nutzerdaten möglich wie für Windows-Clients (Notebooks)

Tool dafür ist winbind (Samba-Tool, s.u. bei ADS-Setup),
einzubinden in nsswitch und pam

Dateirechte

basieren auf Unix-Dateirechten:

- Share bzw. Logon-User wird auf Unix-User abgebildet
  d.h. Windows SID $\longleftrightarrow$ Unix uid/gid
- Unix-Attribute werden auf Windows-Attribute abgebildet
- Posix-ACLs auf Unix eröffnen mehr „Windows-Möglichkeiten“

daher viele Parameter in `smb.conf` zu

- user/group: mapping, Beschränkungen, erzwingen
- Rechte: Masken, Default-Werte

Nutzerdatenbank

Windows-Nutzer, Mapping zu Unix-Nutzern und Privilegien werden in einer Datenbank verwaltet. Samba unterstützt

tdb lokale Dateien (`tdbsam`; andere DB-Typen gelten als veraltet)
tdbsam ist inzwischen cluster-fähig
(für Backup mit laufendem Samba: `tdbbackup` verwenden!)

LDAP Verzeichnisdienst (`ldapsam`)
LDAP ist aufwändiger, sinnvoll bei übergreifenden Lösungen (Unix- und Windows-Nutzer) oder mehreren Samba-Domäncontrollern (PDC/BDC).

Privilegien

Sind Rechte eines Windows-Nutzers unabhängig von Dateien,
z.B. Erlaubnis für die Aufnahme eines Clients in eine Domäne

Probleme

- komplizierte Setups, u.A. wg. verschiedener Ebenen & Sichtweisen, teilweise Skripting nötig oder fertige Skripte einzubinden
- Kenntnisse in Unix und Windows notwendig
- Datei- und Nutzernamen
  - Case-(in-)sensitivity
  - Codepages
- Nutzerverwaltung
  - nach LDAP verlagern oder „doppelt“ (Unix & Samba)
  - Passwörter sind einmal neu zu setzen, evt. synchron zu halten

Konfiguration

- alles in einer Textdatei `/etc/samba/smb.conf`
- dort Abschnitte für
  - globale Einstellungen (G)
  - Einstellungen je Share (S)
- Konfigurationsfrontends (swat, webmin) verfügbar

Konfigurationsprobleme

- erschlagende Flut von Optionen / Möglichkeiten (≈350 Parameter)
- viele Optionen haben Seiteneffekt auf andere, teilweise werden andere dadurch ignoriert
- verschiedene Bezeichnungen für gleiche Option (z.B. `guest ok` gleichbedeutend mit `public`)

Workgroup/Standalone

hauptsächlich zwei verschiedene (bedingt mischbare) Typen,
`security =`

`share` Gastfreigaben ohne Benutzername/Passwort
sinnvoll für Druckertreiber, Software-Repositories

`user` Arbeitsgruppen-Server mit Benutzerberechtigungen
Unix-Rechte greifen, zusätzliche Einschränkungen durch Samba
(z.B. nur bestimmte Gruppe darf Share mounten,
gewisse Nutzer nur schreiben)

Windows-ADS-Domäne

Samba-Server kann als Fileserver Mitglied in einer ADS-Domäne werden und Benutzer über ADS prüfen (`security=ads`).[2]

- Samba sollte DC als DNS-Server nutzen
- gemeinsamer Zeitserver `time1.rrzn.uni-h...` oder DC als Zeitserver
- Nutzer-/Client-Authentifizierung nutzt Kerberos (Parameter `realm=`). Anpassungen in `krb5.conf` nicht mehr nötig, geht automatisch per SRV-RRs im DNS.
- Maschinenkonto für Samba-Server im AD nötig (`net ads join`)
- `winbindd` stellt wiederverwendbare DC-Verbindung bereit.

[2] nicht mit security=domain und Beitritt zu einer NT4-Domäne verwechseln!

Windows-ADS-Domäne: ID-Mapping

Problem

Zuordnung Windows-Nutzer zu Unix-Nutzer schwierig, da Nutzer i.Allg. nicht einzeln auf Unix-System einrichtbar (und analog Gruppen).
Muss aber wegen Dateiberechtigungen sein.

Lösung

- Nutzer- & Gruppen-Namen dynamisch per `nsswitch` von DC (analog wie User-Lookup in einem LDAP)
- Windows-Namen und -SID werden lokal nicht verwendeten `uid` und `gid` zugeordnet (idmap).

für diese Dinge ist der `winbindd` zwingend notwendig

Windows-ADS-Domäne: ID-Mapping

- Modul `nss_winbind` in `nsswitch.conf` eintragen
- in `smb.conf` das Moduls bzw. `winbindd` konfigurieren.
- Mapping Windows-Name zu SID durch Abfrage DC einfach,
- Mapping SID zu uid/gid (`idmap`) schwierig:

  SID besteht aus 96-Bit Domän-ID und 32-BIT Relativ-ID (RID)

  uid/gid traditionell 16 Bits, heute unter Linux 31 Bits
- mehrere Möglichkeiten konfigurierbar:
  - Wahl aus reservierter Range und dauerhafte Fixierung (tdb,ldap)
  - Verwendung der RID + fester Offset (> lokal verwendete uid/gid)
  - Aus dem AD auslesen, wenn Services-For-Unix auf DC installiert sind (eher nicht empfehlenswert)
  - Umgehung aller Mapping-Versuche, sondern direkt (Windows-Name=Unix-Name)

R|R|Z|N

Samba-NT-Domäne

(Haupt)-Konfigurationsparameter in `smb.conf`: `domain logons=yes`

- stellt eine NT4.0-Domäne dar
- Samba kann zwar für die SAM-Datenbank auch ldap verwenden, Nutzerstruktur bleibt aus Windows-Sicht aber flach
- Nutzer- & Computer-Accounts benötigen entsprechenden Unix-Nutzer
- Mischung mit Windows-PDC/-BDC nicht sinnvoll (lieber 2xSamba)
- veraltete DC-Technik ist ein Problem:
  - moderne Clients im Kompatibilitätsmodus (Frage: Port 139 doch nötig?)
  - keine Gruppenrichtlinien
  - Roaming-Profiles möglich, aber z.B. Registry-Änderungen für Vista
- AD-Fähigkeiten für Samba-4 angekündigt, kann noch dauern

Samba-NT-Domäne: Netlogon & Policies

Es gibt gewissen, aber nicht gleichwertigen Ersatz für GPOs:

- Logon-Skripte können über den Netlogon-Share verteilt werden, eigentlich nur ein Login-Skript für alle Rechner
- NT4-Policies z.T. ähnliche Aufgaben wie GPOs, deutlich eingeschränkt im Umfang, zudem Registry-Tainting
- Administrative Vorlagen von GPOs z.T. in NT4-Policies wandelbar

Zusatzmöglichkeit gegenüber Windows-Server:
Samba unterstützt *substitutions*, bei denen Datei- oder Pfadnamen serverseitig umgeschrieben werden, z.B.

- `%U` Username der Client-Sitzung
- `%M` Hostname des Client-Rechners

IP-Beschränkungen

- in `smb.conf` möglich mit `hosts deny` und `hosts allow`
- dieses auch pro Share, ggf. verschieden
- Achtung: Share-Einstellung wird ignoriert bei Global-Einstellung
- wie immer zusätzlich IPTables-/Firewallbeschränkungen

Nutzerbeschränkungen

- neben Berechtigungen im Unix-Dateisystem
- auch Berechtigungen im Zugriff auf Shares

Samba 4

- schon länger in Entwicklung, bisher nur „alpha-Releases“
- bessere Integration als Client in AD
- als AD-Domaincontroller inkl.
  - DNS-Server
  - LDAP
  - Kerberos
- smb2-Support

… aber wohl eher noch weit hin,
gewisser Fortschritt an Integration mit Samba-3-Sourcen erkennbar

Schlussbemerkungen: Fazit

DC vom Einsatz als Domain-Controller ist eher abzuraten, zumindest

- jetzt nicht mehr neu einführen
- nur in kleinen Arbeitsgruppen, wo GPOs nicht notwendig sind, oder in Umfeld mit überwiegend Unix-Desktops
- mit Client-Updates auf Vista oder Windows-7 immer umständlicher

Lieber einen Win2k3/2k8-basierten DC, dazu ggf. Samba als Fileserver

FS gute Lösung als File-Server in einer Workgroup-Umgebung oder als File-Server für Software-Verteilung etc.

Links/Literatur

- Samba-Tutorial-Reihe der iX (2008):
  - I Samba als Standalone-Fileserver (iX 3/2008)
  - II Samba als AD-Mitglied (iX 4/2008)
  - III Samba als Domain-Controller (iX 5/2008)
- Richtlinien in einer NT4-Domäne vgl. Sicherheitstage WS 2005:
  http://www.rrzn.uni-hannover.de/fileadmin/it_sicherheit/pdf/SiTaWS05-Policies.pdf
- Printing/CUPS vgl. Sicherheitstage WS 2006:
  http://www.rrzn.uni-hannover.de/fileadmin/it_sicherheit/pdf/SiTaWS06-CUPS.pdf
- Samba Dokumentation umfassen: Manpage smb.conf und Samba-Howto
  http://de.samba.org/samba/docs/man/Samba-HOWTO-Collection/