OpenAFS an der HSZ-T

Peter Schmid

Hochschule für Technik Zürich
Studiengang Informatik

2.11.2008

Outline

HOCHSCHULE FÜR TECHNIK ZÜRICH

Wieso OpenAFS an der HSZ-T?

- Echtes Distributed Networked File System (DFS).
- Uniform (Global) Namespace, `/afs/hsz-t.ch/` ist weltweit gültig.
- Sichere Authentifizierug über Kerberos.
- Snapshots (Readonly Copies).
- Clients für UNIX/Linux, Windows und OSX. Server für Linux, Solaris und AIX.
- Open Source. Aktive Weiterentwicklung.
- AFS skaliert gut. Zehntausende Workstations, zehntausende Benutzer und hunderte Dateiserver sind keine Seltenhei
- Hohe Zuverlässigkeit dank redundanten Database Server.

HOCHSCHULE FÜR TECHNIK ZÜRICH

Geschichtlicher Rückblick

- Ursprünglich universitäres Projekt an der Carnegie Mellon Universität. Teil des Andrew Projects (1983).
- Später von Firma Transarc unter Transarc AFS kommerzialisiert (1989).
- Transarc wird von IBM übernommen und Produkt als IBM-AFS vermarktet (1999).
- Ab 2000 unter Open Source Lizenz (IBM Public License).

HOCHSCHULE FÜR TECHNIK ZÜRICH

HSZ-T

- DEC Athena mit kerberisiertem NFS war von 1993 bis 2002 im Einsatz. Basierte auf Kerberos IV.
- DEC Athena war nicht für Linux verfügbar → Wunsch nach Ersatz, musste sicheres DFS sein, Clients für UNIX, Linux und Windows.
- Im 2002 neuer Fileserver mit OpenAFS, NIS für Verzeichnisdienst und kaserver für Authetifizierung.
- LDAP als Verzeichnisdienst (2004), MIT Kerberos V (2006).

HOCHSCHULE FÜR TECHNIK ZÜRICH

Bestandteile eines AFS Systems

- AFS Server, Clients.
- Administratoren, Benutzer (z.B.`fruppen` und Gruppen (z.B. `system:anyuser`)
- Cells z.B. `hsz-t.ch`
- Volumes z.B. `user.pschmid`

Cells

- Unabhängige Verwaltungseinheiten
- Verbund von Database Servers und File Servers bilden Datenraum.
- Zugriff auf Daten anderer Zellen.
- Clients haben Standard Zelle.
- Name leitet sich aus Domain ab, z.B. `hsz-t.ch`.
- Admin gibt Struktur der Zelle vor.

Volumes

- Enthalent Verzeichnisse und Dateien des Dateiservers.
- Volumenname z.B. `public.share`
- Veränderbare max. Größe (Quota).
- Zugriff über MountPoint im AFS Dateibaum `/afs/hsz-t.ch/public/share`.
- Jeder Benutzer erhält eine eigene Volume.
- Volumeinstanzen
  - Read Write (RW)
  - Read Only (RO)
  - Backup
  - Temporäre Clones

HOCHSCHULE FÜR TECHNIK ZÜRICH

ACLs

AFS Zugriffsrechte:

- Werden in ACLs (Access Control Lists) verwaltet.
- Gelten nur für Verzeichnisse.
- Können für einzelne Benutzer oder Gruppen festgelegt werden.
- Rechte werden auch in Unterverzeichnisse übernommen.
- ACL Rechte: Read (`r`), Write (`w`), Lookup (`l`), Insert (`i`), Delete (`d`) , Lock (`k`), Administer (`a`).
- Vereinfachung der Rechte: `read` für `rl`, `write` für `rlidwk`, `all` für `rlidwka`, `none` entfernt alle Rechte.

HOCHSCHULE FÜR TECHNIK ZÜRICH

Tokens

- Zeitlich begrenzte Authentifizierung, z.B. 25 h.
- Client erhält Token von AFS Server.
- Token enthält verschlüsselte Benutzeridentifikation.
- Kann nur von AFS-Server entschlüsselt werden.
- Zellenweiter einheitlicher Schlüssel.

Benutzergruppen

Gruppen sind Benutzern gleichgestellt.
Von AFS vorgegeben Gruppen:

- `system:anyuser`
  Jeder Benutzer, auch ohne gültige AFSBerechtigung.
- `system:authuser`
  Jeder Benuter mit gültiger AFS Berechtigung (Token).
- `system:administrators`
  Administratoren Gruppe.
- Jeder Benutzer kann auch eigene Gruppen definieren.

Client-Caching

- Entlastet Dateiserver → bessere Performance bei WANs.
- Aufteilen in Chunks.
- Cache ist persistent.
- Unter Windows Memory Mapping möglich, einzige Datei.
- Unter Unix viele Dateien in einem Verzeichnis.

Client Software

- Üblicherweise ist jeder Client einer Heimatzelle zugeordnet.
- Stellt Anfrage an AFS Server.
- Hält lokalen Cache.
- Synthetisiert AFS Dateibaum.
- Verwaltet Tokens der Benutzer.
- Speichert Liste mit AFS Servern.

Servertypen

Servertypen in einer AFS Zelle:

- File Server
  - Volume Server
  - File Server
- Database Server:
  - Kerberos Authentifizierungs Server
  - Protection Server
  - Volume Location Server
  - Backup Server

File Server

- Enthält alle Dateien einer AFS Zelle.
- Aufteilung der Plattenkapazität.
- Hält Dateipartitionen mit Instanzen von Volumes z.B. `/vicea`, `/viceb` usw.
- Files und Directories sind in Volumes organisiert.
- Können mehrere IP Adressen haben.

File Server, cont.

Prozesse des OpenAFS Fileservers:

- Fileserver: Bedient Anfragen von AFS Clients.
- Volserver: Stellt Funktionen für Admins bereit. (Volume clonen, Volume an- und abschalten ...)
- Salvager: Testet und repariert Verwaltungsstrukturen (ähnlich `fsck`.

Database Server

- DB Server sind untereinander vernetzt.
- Einstiegspunkt für Clients.
- Verwaltung der DB Server in `CellServDB` oder per DNS beim Client.
- Tauschen Infos über UBIK Protokoll aus.
- Absoluter Gleichlauf der internen Server Uhren erforderlich.

Database Server, cont.

Database Server verwalten folgende Datenbanken:

- Protection Database
  - Verwaltet Benutzer der Zelle und Gruppen.
- Volume Database
  - Führt Buch über Volumes auf Dateiservern.
  - Speichert Liste mit IPs des Dateiservers.
- Backup Database
  - verwaltet das AFS Backup System (Tapes).
- Kerberos Database (veraltet)
  - Arbeitet mit KerberosIV DB.
  - Verwaltet Namen und Passwörter der Benutzer einer AFS Zelle.

Setup

- Das Aufsetzen einer AFS Zelle ist deutlich schwerer, als z.B. das Anlegen einer SMB-Share oder NFS Exports.
- Kryptografische Absicherung mittels Kerberos erfordert hohen Aufwand.
- Nach Installation ist Admin Aufwand verhältnismäßig gering.

Benötigte Dienste

- Network Time Protocol (xntp)
- Domain Name System DNS.
- Kerberos V z.B. von MIT (eigener Kerberos IV Server `kaserver` ist eingebaut, sollte aber nicht mehr benutzt werden).
- Directory Service z.B. LDAP, NIS usw.

AFS Zelle hsz-t.ch

- Etwa 1500 Benutzer-Accounts.
- Jeder Account hat ein eigenes Volume, Quota 1 GB.
- 2 File Server mit insgesamt 1 TB Diskkapazität.
  - Sun Solaris, Volumes unter ZFS, Sun V880.
  - Linux OpenSUSE auf IBM Server.
- 300 Clients: Solaris (Sun Ray Thinclients), Linux und Windows Clients.
- LDAP (Sun Java System Directory server), MIT Kerberos
- TCL-Scripts für Benutzerverwaltung (AFS, LDAP, Kerberos, Samba)
- Passwordsynchronisation über Webapplikation (Samba NT Domain und Kerberos Passwörter).

HOCHSCHULE FÜR TECHNIK ZÜRICH

Dunkle Ecken

- ACLs nur für Verzeichnisse. Auch sonst ist das Konzept veraltet.
- Neuere SSH Server erhalten keinen gültigen Token (Linux).
- RW Volumes können nicht repliziert werden.
- Gruppen können keine Gruppen beinhalten.
- Samba und Apache können nur bedingt mit AFS umgehen (Tokens).

HOCHSCHULE FÜR TECHNIK ZÜRICH