science + computing

| an atos company

NFS oder die Unmöglichkeit zusammenzuarbeiten

Daniel Kobras
science + computing ag
IT-Dienstleistungen und Software für anspruchsvolle Rechnernetze
Tübingen | München | Berlin | Düsseldorf

Gameplay

science + computing
| an atos company

- Gruppe von Anwendern
- arbeitet in verschiedenen, teils überlappenden Projektteams
- Zentrale NFS-Server mit getrennten Freigaben für jedes Projekt
- Berechtigungsstruktur anhand der Projektteams

→ Wie werden Berechtigungen abgebildet?
→ Wie werden passende Berechtigungen vorgegeben?

Bonuspunkte

science + computing
| an atos company

- Limits:
    - Beliebig viele Projekte
    - Beliebig viele Nutzer
    - Mehrstufige Berechtigungen (r/o, r/w, Projektleiter, ...)
- Usability:
    - Rechtestruktur bedienbar/nachvollziehbar für Anwender
    - Robust gegen Nutzer-Interaktion
- Zukunftssicher:
    - standardisiert
    - implementierungsunabhängig
- Pflegeaufwand:
    - Mit Bordmitteln umsetzbar
    - Kein Wartungsbedarf für Zusatzinfrastruktur

Level 0: Plain old NFS

science + computing
| an atos company

- Lösungsweg:
  - NFS-Projektfreigaben
  - Unix-Gruppen für jedes Projektteam
  - Gruppenschreib/leserechte, sgid/sticky-Bit auf Projektverzeichnis

Level 0: Plain old NFS

science + computing
| an atos company

- Wertung:
  - Falsche Rechte durch unpassende umask oder aus anderen Bereichen verschobene Dateien
  - Unterverzeichnisse erhalten sgid-Bit nicht automatisch
  - Nur eine Berechtigungsgruppe möglich (kein proj1-rw, proj1-ro)
  - Probleme für Accounts in mehr als 16 Gruppen
  - Bonuspunkte: Standard-Bordmittellösung

→ bestenfalls als simples Austauschverzeichnis (Netzwerk-/tmp) tauglich

Level 1: Cronjobs FTW!

science + computing
| an atos company

- Lösungsweg:
  - Skript bringt Berechtigungen regelmäßig auf Soll-Zustand
- Wertung:
  - „Selbstheilende Probleme“, für Anwender nur bedingt nachvollziehbar
  - Zeitverzögerung bis Cronjob-Lauf
  - ineffizient in großen Verzeichnisbäumen
  - Pflegeaufwand für Skript
  - Nur eine Berechtigungsgruppe pro Unterverzeichnis möglich
  - 16-Gruppen-Limit
  - Bonuspunkte: Robust, simpel, Bordmittel

→ Holzhammermethode, nicht schön, nicht flexibel, aber effektiv

Level 2: Taking the V out of VCS

science + computing
| an atos company

- Lösungsweg:
  - Gruppenmitgliedschaft wird nicht notwendig fest vergeben
  - Dynamische Gruppenzuweisung über suid/sgid-Wrapper
  - Diverse Varianten:
    - Dateieigentümer beibehalten oder auf einheitlichen Projektaccount setzen
    - Lesezugriff über gewöhnliche Gruppenmitgliedschaft oder nur über Wrapper
    - Vom einfachen **`newgrp`**-Wrapper bis zum umfassenden Data-Management-System

Level 2: Taking the V out of VCS

science + computing
| an atos company

- Wertung:
  - Jederzeit korrekte Zugriffsrechte
  - Effizient, keine Dateisystem-Scans nötig
  - Je nach Variante:
    - rw/ro-Gruppen möglich
    - 16-Gruppen-Limit wird umgangen
    - Interaktion erheblich erschwert
    - Standardtools funktionieren nicht oder nur eingeschränkt
  - Hoher Pflegeaufwand für Insellösung
  - Bonuspunkte: wenig praktische Limitierungen, robust und nachvollziehbar

→ Gefahr der Datenhaltung außerhalb Projektverzeichnis
→ Umstieg auf Versionskontrollsystem?

Level 3: Power to the Server

science + computing
| an atos company

- Lösungsweg:
  - NFS-Server wertet Gruppenmitgliedschaften selbst aus
  - Varianten:
    - verwirft Gruppen (bis auf primäre) in RPC (Linux, NetApp)
    - fügt serverseitige Gruppen hinzu (Solaris)

Level 3: Power to the Server

science + computing
| an atos company

- Wertung:
  - Implementierungsabhängig
  - Linux-Variante (**`--manage-gids`**):
    - verletzt NFSv3-Standard
    - clientseitige Gruppenmitgliedschaften (bis auf primäre) unwirksam
    - Accounts müssen auf NFS-Server bekannt sein
  - Kein Schutz gegen falsch gesetzte Rechte
  - Nur eine Berechtigungsgruppe möglich
  - Bonuspunkte: wenig praktische Limitierungen, geringer Wartungsaufwand

→ pragmatische Lösung, speziell Linux-Variante aber anfällig für subtile Kompatibilitätsprobleme

Level 4: Is Posix afraid of Poseven?

science + computing
| an atos company

- Lösungsweg:
  - Rechtevergabe über Posix-ACLs
  - auf Nutzer- und/oder Gruppenbasis
  - Vorgabe von Rechten über Default-ACLs
  - Ausgewertet und durchgesetzt auf NFS-Server
  - Als Erweiterung sichtbar auf NFS-Client
  - Unterstützung in meisten Betriebs- und Dateisystemen
  - Unterstützung in coreutils, rsync, Filemanager-GUIs

Level 4: Is Posix afraid of Poseven?

science + computing
| an atos company

- Wertung:
  - Nur Mode-Bits (rwx)
  - ACL-Größe beschränkt
  - Sichtbarkeit auf Clientseite implementierungsabhängig
  - Bonuspunkte: Nachvollziehbar, robust, breit verfügbar, Bordmittellösung

→ Praxistaugliche Lösung, Kombination mit **`--manage-gids`** sinnvoll, aber wegen fehlender Standard nur bedingt zukunftssicher

Level 5: Newer is better

- Lösungsweg:
  - NFSv4 (AUTH_SYS) als Drop-In-Replacement für NFSv3
  - Standardvorgabe in aktuellen Betriebssystemversionen
- Bewertung:
  - AUTH_SYS auf RPC-Ebene wie NFSv3 (16 Gruppen)
  - Keine **`--manage-gids`**-Option verfügbar
  - Idmapping auf NFS-Ebene erforderlich, umgehbar über RFC3530bis-Erweiterung
  - Linux: Mapping-Einträge beschränkt durch Kernel-Keyring
  - Bonuspunkte: Standard, Bordmittel

→ Wurde irgend etwas wirklich besser? Was ist mit ACLs?
→ Bekommen eigenen Level.

Level 6: One version to rule them

- Lösungsweg:
  - Eigentlich kein Lösungsweg, sondern Zwang mit NFSv4
  - Rechtevergabe mit NFSv4-ACLs
  - auf Nutzer- und Gruppenbasis
  - vererbbar, feingranular
  - Native Unterstützung:
    - NFS-Server
    - Linux: nfs4-acl-tools (CLI)
    - Samba VFS-Modul
    - ZFS, GPFS
  - Andere Dateisysteme:
    - Mapping von NFSv4-ACLs → Posix-ACLs (verlustbehaftet)
    - Irgendwann: Rich-ACLs

Level 6: One version to rule them

- Bewertung:
  - Für Anwender kaum nachvollziehbar:
    - Keine Unterstützung durch Filemanager-GUIs
    - Keine Unterstützung in coreutils (cp, mv, ls)
      → ACLs gehen verloren beim Bewegen von Daten zwischen NFSv4-Freigabe und lokalem Dateisystem
      → ACLs werden bestenfalls als opaque xattr beim Kopieren innerhalb NFSv4-Freigabe übernommen, durch anschließendes chmod eventuell verfälscht
  - Verhalten anpassbar über **`/etc/xattr.conf`**:

```
system.nfs4_acl permissions   # ACLs ignorieren
system.nfs4_acl skip          # ACLs ignorieren
```

  - Abbildung Mode-Bits (rwx) auf NFSv4-ACLs möglich, aber unklar: soll **`chmod()`** ACLs ersetzen oder ergänzen? Reihenfolge der Einträge?

Level 6: One version to rule them

science + computing
| an atos company

- Bonuspunkte: Standard, irgendwie, Bordmittel

→ Rückschritt gegenüber NFSv3 wegen mangelhafter Unterstützung in Tools und vielen Dateisystemen
→ Mit NFSv3 und Posix-ACLs funktionierende Szenarien deshalb nicht gleichwertig mit NFSv4 umsetzbar
→ Nach wie vor Limitierung auf 16 Gruppen

Level 7: NFS from hell

science + computing
| an atos company

- Lösungsweg:
  - Umstellung auf NFSv4 mit AUTH_GSS (Kerberos)
  - Gruppenzugehörigkeit wird voll auf Serverseite ermittelt
- Bewertung:
  - Kein 16-Gruppen-Limit
  - Keine clientseitige Gruppenzuordnung möglich (weder primär, noch sekundär → keine sgid-Skripte)
  - Komplexe Infrastruktur
  - Beschränkte Ticketlaufzeiten schaffen Probleme für
    - Nicht-interaktive Prozesse (Cronjobs, Batch-Jobs, Daemon-Prozesse)
    - Lang laufende Prozesse in interaktiven Sessions
  - Übertragung von Privilegien problematisch

Level 7: NFS from hell

- Bonuspunkte: keine praktischen Limitierungen, Standard, Bordmittel

→ Nach reiner Lehre das Beste vom Besten. Extrem aufwändig umzusetzen, ohne endgültige Lösung zu bieten.

Abspann

science + computing
| an atos company

- Simple Anforderung, alltäglicher Use-Case
- „Vorwärtsgewandte Lösung“: NFSv4 mit NFSv4-ACLs und AUTH_GSS/Kerberos
    - Hoher Aufwand
    - Schafft erhebliche neue Probleme
    - Für Anwender kaum nachzuvollziehen
- Momentan beste Alternative
    - veraltete Protokollversion
    - implementierungsabhängige, nicht standardisierte Option auf Serverseite (**`--manage-gids`**/look-aside groups) und
    - implementierungsabhängige, nicht standardisierte Option auf Clientseite (Posix-ACLs)

science + computing

an atos company

Vielen Dank für Ihre Aufmerksamkeit.

Daniel Kobras

science + computing ag

www.science-computing.de

Telefon 07071 9457-0

info@science-computing.de