OpenAFS, OpenLDAP und Kerberos 5 als Server und Client unter Debian

Stefan Heimers

25. Februar 2011

Inhaltsverzeichnis

1 Nötige Debianpakete

Auf dem Client und dem Server

- libpam-openafs-session
- openafs-client
- openafs-krb5
- openafs-modules-source
- libkrb53
- krb5-clients
- krb5-config
- krb5-doc
- krb5-user
- libpam-krb5
- libnss-ldap
- libldap2
- ldap-utils

libpam-openafs-session openafs-client openafs-krb5 openafs-modules-source libkrb53 krb5-clients krb5-config krb5-doc krb5-user libpam-krb5 libnss-ldap libldap2 ldap-utils

Auf dem Server

- openafs-dbserver
- openafs-fileserver
- krb5-admin-server
- krb5-kdc
- slapd

Auf einigen Rechnern hatten wir Probleme mit den veralteten Openafs-Paketen in Debian Sarge, insbesondere mit Kerneln neuer als 2.6.8. Deshalb gehen wir dazu über, Openafs auch auf Sarge-Systemen mit den Etch-Paketen zu verwenden, die wir auf Sarge neu kompilieren.

Hingenen scheinen neuere Versionen von AFS nicht mit Kernel 2.6.8 zu funktionieren (muss noch genauer geprüft werden).

Editiere /etc/apt/sources.list und füge die folgenden Zeilen ein:

```
# neuere pakete f<FC>r sarge
deb http://www.backports.org/debian/ sarge-backports main contrib non-free
```

Danach:

```
apt-get update
```

Nun kann nach einem apt-get update mit der installation der neueren openafs-debianpakete begonnen werden.

2 OpenAFS Kernelmodul

Um mit OpenAFS zu arbeiten muss das Kernelmodul openafs installiert sein. Das Modul muss nicht in /etc/modules eingetragen werden, es wird vom Init-Skrtipt für Openafs automatisch geladen.

Debian liefert dieses Modul leider nur im Quellcode. Wir haben einige vorkompilierte Debian-Pakete für gewisse Kernel und Debian Sarge erstellt.

```
http://debian.seismo.ethz.ch/dists/sarge/main/binary-i386/net/
```

Mit folgender Zeile in /etc/apt/sources.list können die Pakete per apt-get installiert werden.

```
deb http://debian.seismo.ethz.ch sarge ethz_sed main non-free contrib non-us
```

Bis jetzt verfügbare Pakete, per apt-get install paketname installierbar:

- openafs-modules-2.6.8-2-686 (1.3.81-3+10.00.Custom)

- openafs-modules-2.4.27-2-686 (1.3.81-3+10.00.Custom)

Wer einen anderen Kernel benutzt oder unseren Modulen nicht traut kann wie folgt vorgehen.

Wichtig: Der Kernel 2.6.x enthält ein Modul afs.ko. Dabei handelt es sich nicht um OpenAFS, damit lässt sich kein AFS-Server betreiben.

Der Quellcode für das OpenAFS Kernelmodul befindet sich im Debian-Paket openafs-modules-source. OpenAFS ab 1.3.81 funktioniert auch mit 2.6er Kerneln ab Version 2.6.8. OpenAFS bis Version 1.3.74 hat Probleme mit 2.6er Kerneln.

Der Quellcode wird mit "apt-get install openafs-modules-source" unter /usr/src/openafs.tar.gz abgelegt, aber nicht automatisch kompiliert. Dort muss er entpackt werden.

```
cd /usr/src
tar -zxvf openafs.tar.gz
```

Die Quelldateien werden ins Verzeichnis /usr/src/modules/openafs/ entpackt.

Gebraucht werden nun noch der Kernelsource des laufenden Kernels sowie die Konfiguration, welche in den kernel-headers-* Paketen enthalten ist. In unserer Beispielinstallation wird der Kernel 2.4.27-2-686 aus dem Debian-Paket kernel-image-2.4.27-2-686 verwendet. Bei anderen Kernelversionen müssen die Pfade und Versionsnummern angepasst werden.

```
apt-get install kernel-image-2.4.27-2-686 \
     kernel-source-2.4.27 \
     kernel-headers-2.4.27-2-686
```

Bei kernel-headers muss die Version genau dem Output von "name -a" entsprechen. Zur Zeit der Erstellung dieses Dokuments war Openafs Version 1.3.74 in Debian Sarge, und funktionierte nicht mit einem 2.6.8er Kernel. Mit zukünftigen OpenAFS- und 2.6er Kernelversionen sollte es wieder gehen.

```
cd /usr/src
tar -jxvf kernel-source-2.4.27.tar.bz2
cp /usr/src/kernel-headers-2.4.27-2-686/.config \
    /usr/src/kernel-source-2.4.27
cd /usr/src/kernel-source-2.4.27
make-kpkg clean
```

```
make prepare
make prepare scripts
make-kpkg --append_to_version -2-686 configure
make-kpkg --append_to_version -2-686 modules_image
```

Eine ausführlichere Beschreibung für die Erstellung dieses Modules liegt unter /usr/src/modules/openafs/debian/README.modules.

Der Befehl make-kpkg erzeugt ein Debian-Paket mit dem kompilierten OpenAFS-Kernelmodul. Dieses muss nun installiert werden:

```
cd /usr/src
dpkg -i openafs-modules-2.4.27-2-686_1.3.74-1+10.00.Custom_i386.deb
```

Das Modul sollte jetzt unter /lib/modules/2.4.27-2-686/fs/openafs.o liegen und sich laden lasen:

```
modprobe openafs
```

Das Modul muss sowohl auf dem Server als auch auf dem Client installiert und geladen werden.

3 Einrichtung des Kerberos-Servers

Die Konfiguration befindet sich unter /etc/krb5kdc/. In /etc/krb5kdc/kdc.conf muss der Abschnitt [realms] angepasst werden, in unserem Beispiel:

```
[kdcdefaults]
        kdc_ports = 750,88

[realms]
SED.ETHZ.CH = {
                database_name = /var/lib/krb5kdc/principal
                admin_keytab = FILE:/etc/krb5kdc/kadm5.keytab
                acl_file = /etc/krb5kdc/kadm5.acl
                key_stash_file = /etc/krb5kdc/stash
                kdc_ports = 750,88
                max_life = 10h 0m 0s
                max_renewable_life = 7d 0h 0m 0s
```

```
                master_key_type = des3-hmac-sha1
                supported_enctypes = des3-hmac-sha1:normal des-cbc-crc:normal de
                default_principal_flags = +preauth
        }
```

Nach der Installation der Kerberos-Debianpakete muss die Principal-Datenbank mit folgendem Befehl initialisiert werden:

```
kdb5_util create -s
```

4 Einrichtung des Kerberos-Clients

Die Clientkonfiguration befindet sich in /etc/krb5.conf. Im Abschnitt libdefaults muss default_realm gesetzt werden, zum Beispiel:

```
[libdefaults]
        default_realm = SED.ETHZ.CH


[realms]
SED.ETHZ.CH = {
         kdc = newseismo.ethz.ch
         admin_server = newseismo.ethz.ch
}


[domain_realm]
  .ethz.ch = SED.ETHZ.CH
  ethz.ch  = SED.ETHZ.CH
```

SED.ETHZ.CH durch den eigenen Domainnamen ersetzen, aber in Grossbuchstaben.

Weiter unten in /etc/krb5.conf ist ein Abschnitt [realms], in dem das eigene Realm (meist gleich dem Domainnamen in Grossbuchstaben) eingetragen werden muss. Ebenfalls muss der Kerberos-Server angegeben werden. WICHTIG: Beim Kerberosserver (kdc=...) muss derjenige Name angegeben werden, der auf dem Server mit /bin/hostname ausgegeben wird, und nicht ein Alias.

Bei Problemen kann auch noch eine logging-Section angelegt werden:

```
[logging]
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmin.log
default = FILE:/var/log/krb5lib.log

[realms]
        SED.ETHZ.CH = {
                kdc = newseismo.ethz.ch
                admin_server = newseismo.ethz.ch
                default_domain = sed.ethz.ch
        }
```

5 Einrichtung des AFS-Servers

Die Einrichtung wird an einem Beispiel gezeigt. In unserem Beispiel heisst der Server newseismo.ethz.ch und die AFS-Cell sed.ethz.ch.

Die AFS-Cell muss eingestellt werden (im Allgemeinen die Domain, in der die Server stehen, kann aber beliebig gewählt werden)

Zuerst müssen einige Zugriffsreche für die frisch angelegten Konfigurationsverzeichnisse korrigiert werden, sonst will der AFS-Server nicht starten:

```
chmod  755 /etc/openafs/server
chmod  770 /etc/openafs/server-local
```

In /etc/openafs/ThisCell steht nur der Name der Zelle:

```
sed.ethz.ch
```

In /etc/openafs/CellServDB steht eine Liste mit Zellen, die für den Client gebraucht werden sollen. An erster Stelle tragen wir unsere neue Zelle und den Serverein: (Die Einträge sind ev. schon von der Debian-Konfiguration erstellt worden, bitte zuerst prüfen, ob sie schon existieren.)

```
>sed.ethz.ch
129.132.17.20            # newseismo.ethz.ch
>grand.central.org      #GCO Public CellServDB 10 Oct 2004
18.7.14.88                      #grand-opening.mit.edu
128.2.191.224                   #penn.central.org
```

```
>wu-wien.ac.at          #University of Economics, Vienna, Austria
137.208.3.33                    #afsdb1.wu-wien.ac.at
137.208.7.4                     #afsdb2.wu-wien.ac.at
137.208.7.7                     #afsdb3.wu-wien.ac.at
# und so weiter...
```

Für den Server gibt es ebenfalls eine Datei CellServDB unter /etc/openafs/server/CellServDB. Dort muss unsere Zelle ebenfalls eingetragen werden:

```
>sed.ethz.ch    #Cell name
129.132.17.20    #newseismo.ethz.ch
```

AFS legt seine Daten in Unterverzeichnissen mit Namen /vicepa, /vicepb,... an. Eines soll nun erzeugt werden:

```
mkdir /vicepa
```

Es wird empfohlen, dafür eine eigene Partition anzulegen.

In Kerberos einen Principal[1] mit Namen afs anlegen

```
kadmin.local -q "ank -randkey afs"
```

Principal afs zur Keytabelle krb5.keytab.afs hinzufügen. Die Keytabelle wird benötigt, damit afs auf den Kerberos Server zugreifen kann ohne, dass jedesmal ein Passwort angegeben werden muss.

```
kadmin.local -q "ktadd -e des-cbc-crc:afs3 -k /etc/krb5.keytab.afs afs"
```

Asetkey ersetzt den AFS "bos setkey" Befehl. Die Zahl nach "add" ist die Zahl, die ktadd nach "Key: vno" ausgibt. Wurde ktadd schon früher ausgeführt und die Zahl ist nicht bekannt hilft getprinc:

```
kadmin.local -q "getprinc afs"
asetkey add 4 /etc/krb5.keytab.afs afs
```

Asetkey hat den Schlüssel jetzt in /etc/openafs/server/KeyFile abgelegt.

Nun kann der Basic OverSeer (BOS) Server gestartet werden. Er muss auf jedem Fileserver laufen, überwacht die AFS Serverprozesse, startet diese neu wenn sie abstürzen, verarbeitet bos-Befehle (Status abfragen, Prozesse starten und stoppen).

[1] Ein Principal entspricht einem Login- oder Usernamen bei Kerberos

```
bosserver -noauth
```

Im Normalbetrieb (Start über /etc/init.d/) braucht der bosserver eine Authentifizierung für die folgenden Dienste. Falls Zugriffsfehler auftreten, den bosserver killen und mit bosserver -noauth wieder starten.

Testen, ob bosserver funktioniert und der Server in der CellServDB gefunden wird:

```
bos listhosts newseismo.ethz.ch -noauth
```

Als nächstes wird der Protection Server (ptserver) gestartet.

```
bos create -server newseismo.ethz.ch -instance ptserver -type simple \
  -cmd /usr/lib/openafs/ptserver -cell sed.ethz.ch -noauth
```

AFS Benutzer admin anlegen

```
bos adduser newseismo.ethz.ch admin -cell sed.ethz.ch -noauth
```

Zeige die Serververschlüsselungskeys aus /etc/openafs/server/KeyFile

```
bos listkeys newseismo.ethz.ch -cell sed.ethz.ch -noauth
```

Mit pts wird der Protection Server administriert. Es wird ein Benutzer admin erzeugt und in die Gruppe system:administrators eingetragen, und dann geprüft, ob der Eintrag funktioniert hat.

```
kadmin.local
Authenticating as principal root/admin@SED.ETHZ.CH with password.
kadmin.local:  addprinc admin
WARNING: no policy specified for admin@SED.ETHZ.CH; defaulting to no policy
Enter password for principal "admin@SED.ETHZ.CH":
Re-enter password for principal "admin@SED.ETHZ.CH":
Principal "admin@SED.ETHZ.CH" created.
kadmin.local:  quit
pts createuser -name admin -cell sed.ethz.ch -noauth
pts adduser admin system:administrators -cell sed.ethz.ch -noauth
pts membership admin -cell sed.ethz.ch -noauth
```

Nun soll der BOS Server alle Dienste neu starten

```
bos restart newseismo.ethz.ch -all -cell sed.ethz.ch -noauth
```

Nun soll der eigentliche Fileserver gestartet werden:

```
bos create -server newseismo.ethz.ch -instance fs -type fs \
  -cmd /usr/lib/openafs/fileserver \
  -cmd /usr/lib/openafs/volserver \
  -cmd /usr/lib/openafs/salvager \
  -cmd /usr/lib/openafs/vlserver  -cell sed.ethz.ch -noauth
```

Und der Test:

```
bos status newseismo.ethz.ch fs -long -noauth
```

Die Partition /vicepa wird exportiert:

```
vos create -server newseismo.ethz.ch -partition /vicepa \
  -name root.afs -cell sed.ethz.ch -noauth
```

```
bos shutdown newseismo.ethz.ch -wait -noauth
```

```
pkill bosserver
```

Jetzt wird der Openafs Fileserver “normal” gestartet

```
/etc/init.d/openafs-fileserver start
/etc/init.d/openafs-client start
```

Den Benutzer “admin” über den Kerberosserver authentifizieren lassen. Es wird ein Passwort abgefragt.

```
kinit admin
```

Kerberos-Informationen anzeigen

```
klist
```

Am AFS-Server anmelden. Aklog funktioniert nur, wenn man sich vorher mit kinit authentifiziert hat. Es wird deshalb auch nicht mehr nach einem Passwort gefragt.

```
aklog
```

```
tokens
```

```
fs checkvolumes
```

```
vos create newseismo.ethz.ch /vicepa root.cell
```

```
fs mkmount /afs/sed.ethz.ch root.cell
```

Leserechte für alle auf /afs/sed.ethz.ch setzen

```
fs setacl /afs/sed.ethz.ch system:anyuser rl
```

```
fs mkmount /afs/.sed.ethz.ch root.cell -rw
```

```
pts creategroup sed
```

```
mkdir /afs/sed.ethz.ch/home
```

6 OpenLDAP

OpenLDAP soll verwendet um die Benutzerdaten zentral zu verwalten. Dazu gehören unter anderem User-ID, Gruppen-ID, Adresse, Telefonnummer, Login-Name, Home-Verzeichnis,...

Die Passwörter werden bei uns nicht mit LDAP verwaltet, obwohl das mölich wäre, sondern mit Kerberos weil AFS auf Kerberos ausgerichtet ist.

6.1 Einrichtung des LDAP-Servers

Der OpenLDAP Server befindet sich im Debian-Paket slapd.

```
apt-get install slapd
```

In /etc/ldap/slapd.conf müssen folgende Zeilen eingefügt oder angepasst werden:

```
# The base of your directory in database #1
suffix          "dc=sed,dc=ethz.ch,dc=ch"
rootdn          "cn=Manager,dc=sed,dc=ethz,dc=ch"
rootpw          "geheimnis"
```

Der Eintrag rootpw ist das "Hauptpasswort" für den LDAP-Server. Anstatt dem Klartextpasswort kann auch ein verschlüsseltes verwendet werden. Dies ist in der OpenLDAP-Dokumentation beschrieben und wird hier der Einfachheit halber weggelassen.

Nun muss die Datenbank für die Posix-Accounts vorbereitet werden. Eine Datei mit dem Namen posix_init.ldif und folgendem Inhalt erstellen:

```
dn: ou=People,dc=sed,dc=ethz,dc=ch
ou: People
objectClass: top
objectClass: organizationalUnit
description: Parent object of all UNIX accounts

dn: ou=Groups,dc=sed,dc=ethz,dc=ch
ou: Groups
objectClass: top
objectClass: organizationalUnit
description: Parent object of all UNIX groups
```

Diese wird nun mit folgendem Befehl in die LDAP-Datenbank eingetragen:

```
ldapadd -x -D "cn=Manager,dc=sed,dc=ethz,dc=ch" -W -f posix_init.ldif
```

6.2 Einrichtung von LDAP auf den Clients

Zuerst muss ldap so konfiguriert werden, dass es Anfragen an den richtigen Server stellt. Dies geschieht in der Datei /etc/ldap/ldap.conf die wie folgt aussehen soll:

```
# ldap.conf
BASE    dc=sed,dc=ethz,dc=ch
URI     ldap://ldap1.seismo.ethz.ch ldap://ldap2.seismo.ethz.ch ldap://ldap3.sei
```

Danach muss das System so eingerichtet werden, dass es Benutzerdaten für Benutzer die in /etc/passwd nicht eingetragen sind auf dem LDAP-Server sucht. Dazu dienen die Dateien /etc/libnss-ldap.conf und /etc/nsswitch.conf die wie folgt aussehen:

```
# /etc/libnss-ldap.conf
host ldap1.seismo.ethz.ch ldap2.seismo.ethz.ch ldap3.seismo.ethz.ch
base dc=sed,dc=ethz,dc=ch
# ... (den Rest wie  in der Beispieldatei lassen)

# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the ‘glibc-doc’ and ‘info’ packages installed, try:
# ‘info libc "Name Service Switch"’ for information about this file.

passwd:         compat ldap
group:          compat ldap
shadow:         compat ldap

hosts:          files ldap dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

netgroup:       ldap [NOTFOUND=return] files
```

7 Benutzerverwaltung

Die Benutzer (in diesem Beispiel Username newusername) müssen drei mal eingetragen werden:

7.1 In Kerberos (zur Authentifizierung)

```
kadmin.local -q "ank newusername"
```

Der Benutzer kann später sein Passwort selbst ändern:

```
kpasswd newusername
```

7.1.1 Lebensdauer der Tickets

Mit dem folgenden Befehl wird die maximale Lebensdauer der Service-Tickets für alle Benutzer gesetzt (muss nur einmal ausgeführt werden):

```
kadmin.local -q "modprinc -maxlife \"7 days\" krbtgt/SED.ETHZ.CH@SED.ETHZ.CH"
kadmin.local -q "modprinc -maxlife \"7 days\" afs@SED.ETHZ.CH"
```

Der folgende Befehl setzt die maximale Lebensdauer für den neuen Benutzer (muss für jeden Benutzer zusätzlich zu den beiden obigen ausgeführt werden):

```
kadmin.local -q "modprinc -maxlife \"7 days\" newusername"
```

Dies ist nur wirksam, wenn die entsprechenden Grenzen auch in /etc/krb5kdc/kdc.conf (Option max_life und max_renewable_life) genug hoch gesetzt sind und der Benutzer kinit mit der Option -l benutzt, z.B.:

```
kinit -l 2d # get a ticket with two days lifetime
aklog       # aklog gets a token with the same lifetime as the ticket
klist       # optional, check the kerberos tickets
tokens      # optional, check the afs tokens
```

7.2 In AFS

```
pts createuser -name newusername -id 1023
```

Die Angabe von -id ¡nummer¿ ist freiwillig, es ist aber zu empfehlen, die Unix-ID (UID), die auch in LDAP oder /etc/passwd verwendet wird anzugeben.

Danach muss noch das Home-Verzeichnis des Benutzers erstellt und die Zugriffsrechte gesetzt werden:

```
kinit admin
aklog
mkdir /afs/sed.ethz.ch/home/newusername
fs setacl /afs/sed.ethz.ch/home/newusername newusername rwlidk
```

Mögliche Zugriffsrechte:

- r: Read (Lesen)
- l: Lookup
- i: Insert (Neue Dateien im Verzeichnis erstellen)
- d: Delete
- w: Write (Schreiben)
- k: Lock
- a: Administer

Benutzer aus ACLs löschen: Bei den Permissions 'none' setzen. Beispiel:

```
fs setacl /afs/sed.ethz.ch/gse/IDCproducts/REB sarah none
```

Die Rechte können auch für Gruppen vergeben werden. Drei Gruppen sind vom System vorgegeben:

- system:administrators Administratoren
- system:authuser An der lokalen AFS-Zelle angemeldete Benutzer

- system:anyuser Alle

Andere Gruppen können selbst angelegt werden.

```
pts creategroup -name <group name> [-owner <owner of the group>]
```

Benutzer zu einer AFS-Gruppe hinzufügen:

```
newseismo:/afs/sed.ethz.ch# pts adduser -user conti -group sed
```

```
newseismo:/afs/sed.ethz.ch# pts adduser -user heimers  -group sed
```

Bestehende Gruppe anzeigen:

```
newseismo:/afs/sed.ethz.ch# pts examine sed
Name: sed, id: -206, owner: admin, creator: admin,
  membership: 2, flags: S-M--, group quota: 0.
newseismo:/afs/sed.ethz.ch# pts membership sed
Members of sed (id: -206) are:
  heimers
  conti
```

Für Details siehe

http://www.openafs.org/pages/doc/UserGuide/auusg008.htm#HDRWQ60

7.3 In LDAP

Für den neuen Benutzer eine Datei nach folgendem Muster anlegen:

```
dn: uid=newusername,ou=People,dc=sed,dc=ethz,dc=ch
cn: Newusername at the SED
sn: Newusername
uid: newusername
uidNumber: 1326
gidNumber: 1021
homeDirectory: /afs/sed.ethz.ch/home/newusername
loginShell: /bin/bash
roomNumber: HPP P9
mobile: +41 79 12 34 56
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
```

Die Datei speichern unter newusername.ldif und folgenden Befehl ausführen:

```
ldapadd -x -D "cn=Manager,dc=sed,dc=ethz,dc=ch" -W -f newusername.ldif
```

Bei Änderungen an der Datei muss ldapmodify statt ldapadd verwendet werden:

```
ldapmodify  -x -D "cn=Manager,dc=sed,dc=ethz,dc=ch" -W -f newusername.ldif
```

Alle LDAP-Daten anzeigen:

```
ldapsearch -x
```

8 AFS Benutzer anmelden

Situation: Man will Zugriff auf die Daten eines bestimmten AFS-Benutzers. Man ist ohne AFS oder mit AFS als anderer Benutzer angemeldet.

```
kinit benutzername
aklog
```

Kinit authentifiziert den Benutzer gegenüber dem Kerberos-Server. Aklog verschafft ihm nachher Zugriff auf die Daten auf dem AFS Server. Aklog funktioniert nur nach einem erfolgreichen kinit.

8.1 Lebenszeit des Tickets verändern

Das folgende Beispiel holt ein Ticket mit drei Tagen Lebensdauer:

```
kinit -l 3d benutzername
aklog
```

Dies funktioniert nur, wenn die maximale Lebensdauer mindestens genau so hoch eingestellt ist. Siehe dazu 7.1.1

Abmelden geht mit:

```
unlog
```

9 Workstations mit AFS-Login

Situation: Eine Workstation soll so eingerichtet werden, dass man sich mit dem Login-Manager (z.B. kdm) gleich auf dem Kerberos/AFS Server anmeldet.

Die Authentifizierung geschieht über den Kerberos-Server und das Homeverzeichnis wird über den AFS-Server zur Verfügung gestellt. Weder Kerberos noch Openafs können die Einträge in /etc/passwd (Shell, Heimverzeichnis,...) ersetzen. Die einfachste Lösung wäre /etc/passwd vom Server auf den Client zu kopieren. Auch NIS wäre möglich. Wir haben uns aber für die "Luxusvariante" mit OpenLDAP entschieden.

9.1 Login-Manager

Folgende Zeilen in /etc/pam.d/kdm an erster Stelle eintragen:

```
auth sufficient /lib/security/pam_krb5.so ignore_root
account sufficient /lib/security/pam_krb5.so ignore_root
session optional /lib/security/pam_krb5.so ignore_root
session optional /lib/security/pam_openafs_session.so ignore_root
```

Es können sich sowohl AFS-Benutzer als auch lokal eingerichtete Benutzer über KDM anmelden.

10 Unterschiede zwischen Debianpaketen und den Originalquellen

10.1 Pfade

Debian	Doku von openafs.org
/etc/openafs/	/usr/afs/etc/
/var/log/openafs	/usr/afs/logs
/etc/openafs/server/	/usr/afs/etc/
/etc/openafs/server-local/sysid	/usr/afs/local/sysid
/var/lib/openafs/db	/usr/afs/db

11 Server-Replikation

11.1 Kerberos Server Replikation

http://tldp.org/HOWTO/Kerberos-Infrastructure-HOWTO/server-replication.html

Ergaenzungen:

In /etc/inetd.conf muss folgende Zeile sein:

```
krb5_prop       stream  tcp     nowait  root    /usr/sbin/kpropd kpropd
```

inetd muss neu gestartet werden.

kpropd.acl muss bei Debian unter /etc/krb5kdc/kpropd.acl liegen.

In kpropd.acl muss der kanonical name (Ausgabe von /bin/hostname) des Adminservers eingetragen werden, nicht ein alias.

In /etc/krb5.conf kann aber der alias für kdc = ... benutzt werden.

/etc/krb5kdc/stash muss vom Kerberos Adminserver (Hauptserver) auf den zweiten Kerberosserver kopiert werden.

11.2 AFS Server Replikation

Zuerst muss ein weiterer AFS-Server eingerichtet werden, wie im entsprechenden Kapitel beschrieben.

Nun können Readonly-Replikationen erstellt werden. Das heisst, ein AFS-Volume wird auf einen zweiten Server kopiert. Beim lesenen Zugriff wird die Last nachher auf beide Server verteilt, bei schreibendem Zugriff wird auf den Hauptserver geschrieben.

Beispiel mit qloc als zweitem Server:

```
vos addsite qloc /vicepb root.afs # legt die Replikation auf Server qloc an
vos release root.afs              # Synchronisiert die Replikation mit dem Origi
```

ACHTUNG: Daten müssen periodisch mit “vos release ¡volumename¿” synchronisiert werden.

Werden Daten geschrieben, sind diese zuerst nur auf dem Hauptserver verfügbar. Ein Client, der eine Replikation verwendet, sieht noch die alte Version. Erst ein “vos release ¡volumename¿” synchronisiert die Daten wieder.

DIESE ART REPLIKATION IST DESHALB NICHT FUER DYNAMISCHE VERZEICHNISSE WIE HOME ODER SIGNALS GEEIGNET

Ich empfehle aber zumindest root.afs und root.cell zu replizieren, damit die AFS-Zelle auch funktioniert, wenn der Hauptserver ausfällt.

11.3 LDAP Server Replikation

Das LDAP-Verzeichnis kann von einem Masterserver auf mehrere Slaves repliziert werden. Wir verwenden den Syncrepl Mechanismus.

Auf dem Hauptserver muss folgendes in /etc/ldap/slapd.conf eingetragen werden:

```
# Schema and objectClass definitions
include         /etc/ldap/schema/core.schema
include         /etc/ldap/schema/cosine.schema
include         /etc/ldap/schema/nis.schema
include         /etc/ldap/schema/inetorgperson.schema

# Schema check allows for forcing entries to
# match schemas for their objectClasses's
schemacheck     on

# Where the pid file is put. The init.d script
# will not stop the server if you change this.
pidfile         /var/run/slapd/slapd.pid

# List of arguments that were passed to the server
argsfile        /var/run/slapd.args

# Read slapd.conf(5) for possible values
loglevel        0

# Where the dynamically loaded modules are stored
modulepath      /usr/lib/ldap
moduleload      back_bdb

#######################################################################
# Specific Backend Directives for bdb:
# Backend specific directives apply to this backend until another
```

```
# 'backend' directive occurs
backend         bdb
checkpoint 512 30

#######################################################################
# Specific Backend Directives for 'other':
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
#backend                <other>

#######################################################################
# Specific Directives for database #1, of type bdb:
# Database specific directives apply to this databasse until another
# 'database' directive occurs
database        bdb

# The base of your directory in database #1
suffix          "dc=sed,dc=ethz,dc=ch"
rootdn          "cn=Manager,dc=sed,dc=ethz,dc=ch"
rootpw          "daspasswort"

# Where the database file are physically stored for database #1
directory       "/var/lib/ldap"

# Indexing options for database #1
index           objectClass eq

# Save the time that the entry gets modified, for database #1
lastmod         on

# sessionlog is needed for syncrepl
sessionlog 123 999


access to attrs=userPassword
        by dn="cn=admin,dc=sed,dc=ethz,dc=ch" write
        by dn="cn=Manager,dc=sed,dc=ethz,dc=ch" write
        by anonymous auth
        by self write
        by * none
```

```
access to dn.base="" by * read

access to *
        by dn="cn=admin,dc=sed,dc=ethz,dc=ch" write
        by * read
```

Auf den Replikationsservern muss folgendes in /etc/ldap/slapd.conf eingetragen werden:

```
# Schema and objectClass definitions
include         /etc/ldap/schema/core.schema
include         /etc/ldap/schema/cosine.schema
include         /etc/ldap/schema/nis.schema
include         /etc/ldap/schema/inetorgperson.schema


# Where the pid file is put. The init.d script
# will not stop the server if you change this.
pidfile         /var/run/slapd/slapd.pid


# List of arguments that were passed to the server
argsfile        /var/run/slapd.args

# Read slapd.conf(5) for possible values
loglevel        0

# Where the dynamically loaded modules are stored
modulepath      /usr/lib/ldap
moduleload      back_bdb
moduleload      back_monitor
#moduleload      syncprov.la
#moduleload      back_ldap

# We don't need any access to this DSA
#restrict      "all"


# consumer proxy database definitions
backend         bdb
```

```
checkpoint 512 30

database bdb
suffix   "dc=sed,dc=ethz,dc=ch"
rootdn   "cn=Manager,dc=sed,dc=ethz,dc=ch"
rootpw   "daspasswort"

# Where the database file are physically stored for database #1
directory       "/var/lib/ldap"

# Indexing options for database #1
index           objectClass eq

# Save the time that the entry gets modified, for database #1
lastmod         on


access to attrs=userPassword
    by dn="cn=admin,dc=sed,dc=ethz,dc=ch" write
#    by dn="cn=Manager,dc=sed,dc=ethz,dc=ch" write
    by anonymous auth
    by self write
    by * none

access to dn.base="" by * read
access to *
        by dn="cn=admin,dc=sed,dc=ethz,dc=ch" write
        by * read


# HACK: use the RootDN of the monitor database as UpdateDN so ACLs apply
# without the need to write the UpdateDN before starting replication
acl-bind        bindmethod=simple
                binddn="cn=Monitor"
                credentials=monitor

# HACK: use the RootDN of the monitor database as UpdateDN so ACLs apply
# without the need to write the UpdateDN before starting replication
syncrepl        rid=1
                provider=ldap://ldap.seismo.ethz.ch:389
                type=refreshOnly
```

```
                interval=00:00:05:00
                binddn="cn=Manager,dc=sed,dc=ethz,dc=ch"
                bindmethod=simple
                credentials=daspasswort
                searchbase="dc=sed,dc=ethz,dc=ch"
                filter="(objectClass=*)"
                attrs="*,structuralObjectClass,entryUUID,entryCSN,creatorsName,c
                schemachecking=off
                scope=sub
                type=refreshAndPersist
                retry="5 5 300 5"
#overlay         syncprov
database        monitor
```

12 Backup

Beim Backup mit tar oder anderen einfachen Backupprogrammen ist zu beachten, dass die afs ACLs (Zugriffsrechte) ignoriert werden.

Wird ein solches Backupsystem verwendet, sollten die ACLs vor dem Backup in separaten Dateien gespeichert werden.

Alle ACLs werden in ein File geschrieben.

```
#! /bin/sh
mv /afs/sed.ethz.ch/.afsacl.list /afs/sed.ethz.ch/.afsacl.list.old
find /afs/sed.ethz.ch/  -type d  -exec fs listacl {} \; > /afs/sed.ethz.ch/.afsa
```

To be continued...

Siehe auch **??**

13 Einen weiteren AFS-Server in der gleichen Zelle einrichten

Eine eigene Partition für /vicepa muss auf dem neuen Server (in diesem Beispiel qloc.ethz.ch) erzeugt und gemountet werden:

```
mkfs.ext3 /dev/sdb1
mkdir /vicepa
mount /dev/sdb1 /vicepa
```

(Eintrag in /etc/fstab nicht vergessen!)

Der neue AFS-Server muss zuerst als AFS-Client konfiguriert werden. Dann muss zusätzlich die Server-Software installiert werden:

```
apt-get install openafs-fileserver
```

Nun müssen /etc/krb5.key* und /etc/openafs/server/KeyFile vom Hauptserver auf den neuen Server kopiert werden, und dann der Fileserver neu gestartet werden.

```
/etc/init.d/openafs-fileserver restart
```

Dann müssen die Serverprozesse aktiviert werden:

```
bos create -server qloc.ethz.ch -instance fs -type fs \
-cmd /usr/lib/openafs/fileserver \
-cmd /usr/lib/openafs/volserver \
-cmd /usr/lib/openafs/salvager  -localauth
```

Nun kann ein AFS-Volume erzeugt und gemountet werden:

```
vos create qloc.ethz.ch  /vicepa offline -localauth
fs mkmount /afs/sed.ethz.ch/offline offline
fs setquota /afs/sed.ethz.ch/offline/ 0
fs setacl /afs/sed.ethz.ch/offline sed rl
```

14 Probleme lösen

14.1 Gleicher User hat nur in einer session AFS-Zugriff (ein Token)

Dies wird Ã¼ber sogenannte PAGs geregelt und ist sicherer. Wenn man das nicht will muss man das in /etc/pam.d in der entsprechenden Datei konfigurieren. z.B. so (Option nopag auf der letzten Zeile):

```
auth sufficient /lib/security/pam_krb5.so ignore_root debug
account sufficient /lib/security/pam_krb5.so ignore_root debug
session optional /lib/security/pam_krb5.so ignore_root debug
session optional /lib/security/pam_afs_session.so ignore_root nopag
```

14.2 Falsche Adressen werden verwendet, einzelne AFS-Server lassen sich nicht mehr ansprechen

AFS Prozesse, die Netzwerkverbindungen aufbauen, hören grundsätzlich auf allen IP-Adressen des Computers. Diese Adressen werden auch auf den AFS-Datenbankservern und Volume Location Servern eingetragen.

Probleme gibt es wenn:

- Lokale und Globale Netzwerke verwendet werden
- Die IP-Adressen von Servern nachträglich geändert werden
- Fehlkonfigurationen des Netzwerks vorliegen. z.B. falsche Einträge in /etc/hosts

Manchmal setzen sich die Fehler in den AFS-Datenbanken fest und lassen sich nur schwer korrigieren. Es kann helfen die Volume Location Datenbank einfach zu löschen. Diese enthält keine wichtigen Daten, sie wird nach einem Neustart einfach wieder aus den Daten der einzelnen Servern neu aufgebaut.

Um die Datenbank zu löschen müssen alle AFS-Server kurzzeitig gleichzeitig gestoppt werden, die Datenbankdateien gelöscht werden, und dann die AFS-Server neu gestartet werden.

Das Vorgehen ist folgendes:

Erst mal die Grundkonfiguration des Netzwerks unter Linux richtig einstellen.

Bei Bedarf kann man Openafs mitteilen, welche Adressen es verwenden soll. In /etc/openafs (für den Client) und /etc/openafs/server (für den Server) können die Dateien NetInfo und NetRestrict abgelegt werden. NetInfo enthält IP-Adressen, die verwendet werden sollen, NetRestrict solche, die nicht verwendet werden sollen. Die IP-Adressen in diesen Dateien stehen übereinander. Also jede Zeile eine IP-Adresse. Die Zahl 255 wird als Wildcard verwendet um ganze Subnetze auszuwählen. Folgende NetRestrict-Datei könnte die Verwendung von lokalen Netzen und Adressen verhindern:

```
127.0.0.1
192.168.255.255
```

Zu beachten ist, dass die Einstellungen in NetInfo und NetRestrict nur den Clients mitteilen welche IP-Adressen verwendet werden sollen. Die Server binden sich trotzdem an alle Interfaces des Rechners. Des Weiteren bleiben die falschen Verbindungen von früher in den Datenbanken, deshalb scheinen NetInfo und NetRestrict unter manchen Bedingungen wirkungslos. Auch da muss dann die Datenbank bereinigt, oder einfacher, gelöscht und neu erstellt werden. Dies wird im folgenden Kapitel erklärt.

14.2.1 Volume Location Datenbank (vldb) neu erzeugen

Auf allen ausser einem AFS-Server folgende Befehle ausführen:

```
/etc/init.d/openafs-fileserver stop
rm -f /var/lib/openafs/db/vldb.*
```

Auf dem letzten Server folgendes ausführen:

```
/etc/init.d/openafs-fileserver stop
rm -f /var/lib/openafs/db/vldb.*
/etc/init.d/openafs-fileserver start
```

Nun auf den verbleibenden Rechnern auch noch die Server neu starten:

```
/etc/init.d/openafs-fileserver start
```

Nacher müssen alle Fileserver angewiesen werden, ihre Konfiguration wieder in die Datenbank zu übermitteln. Dazu folgenden Befehl für alle Fileserver ausführen:

```
vos syncvldb fileservername
```

Wichtig ist, dass die Datenbankserver einen Moment lang alle aus sind, dann alle Files gelöscht werden, und die Server erst dann wieder gestartet werden. Wenn ein Server wieder gestartet wird während ein anderer noch mit falschen Daten läuft, werden diese sofort wieder verteilt und nichts ist gewonnen.

Anmerkung: Angebich (**??**, Kapitel VLDB neu aufbauen) sollte man auch den Volserver alleine ohne den Fileserver anhalten können (bos stop servername vlserver), dies ist mir aber bei meinen Versuchen aus unerklärlichen Gründen nicht gelungen.

14.3 fs: Invalid argument; it is possible that /afs is not in AFS

Problem:

```
fs setacl /afs system:anyuser rl
fs: Invalid argument; it is possible that /afs is not in AFS.
```

Leider noch keine Lösung bekannt.

14.4 No such file or directory while initializing kadmin.local interface

```
kadmin.local -q "ank -randkey afs"
Authenticating as principal root/admin@INFORMATIK.UNI-MANNHEIM.DE with password
kadmin.local: No such file or directory while initializing kadmin.local interfac
```

Lösung:

Die Principal-Datenbank muss initialisiert werden:

```
kdb5_util create -s
```

14.5 Decrypt integrity check failed

Symptom: kinit geht, aber login mit pam geht nicht.

Fehlermeldung auf dem Kerberosserver:

```
May 09 17:37:59 newseismo krb5kdc[660](info): AS_REQ (7 etypes
 {18 17 16 23 1 3 2}) 129.132.17.2: NEEDED_PREAUTH: guest@SED.ETHZ.CH
  for krbtgt/SED.ETHZ.CH@SED.ETHZ.CH, Additional pre-authentication required
May 09 17:37:59 newseismo krb5kdc[660](info): preauth (timestamp) verify failure
  Decrypt integrity check failed
May 09 17:37:59 newseismo krb5kdc[660](info): AS_REQ (7 etypes {18 17 16 23 1 3
  129.132.17.2: PREAUTH_FAILED: guest@SED.ETHZ.CH for
  krbtgt/SED.ETHZ.CH@SED.ETHZ.CH, Decrypt integrity check failed
```

Grund: User war auf dem Clientsystem nicht bekannt. Muss in /etc/password eingetragen sein oder über LDAP, NIS, ... verfügbar sein.

14.6 Credentials cache I/O operation failed XXX when initializing cache

Symptom:

Der beschriebene Fehler trat auf bei “kinit guest” nach Eingabe des Passworts. Grund:

Die Platte des Clientrechners war voll.

14.7 bos: you are not authorized for this operation

Problem:

```
newseismo:~# bos create -server newseismo -instance ptserver -type simple -cmd /
bos: failed to create new server instance ptserver of type 'simple' (you are not
```

Lösung: Zugriffsreche /etc/openafs/server und /etc/openafs/server-local

```
chmod  755 /etc/openafs/server
chmod  770 /etc/openafs/server-local
pkill bosserver
bosserver -noauth
```

14.8 Bosserver läuft nicht richtig

Symptom:

```
rx failed to send packet: rx_sendmsg: Invalid argument
rx failed to send packet: rx_sendmsg: Invalid argument
rx failed to send packet: rx_sendmsg: Invalid argument
```

Problem (zu finden in /var/log/openafs/BosLog):

```
Mon Feb 21 14:58:59 2005: unhappy with /etc/openafs/server which
 is a dir that should have at least rights 755, at most rights 775 ,
 owned by root
Mon Feb 21 14:58:59 2005: Server directory access is not okay
```

Die Zugriffsrechte für /etc/openafs/server stimmen nicht.

Lösung:

```
chmod  755 /etc/openafs/server
chmod  770 /etc/openafs/server-local
pkill bosserver
bosserver -noauth
```

Anderes Problem: bosserver Funktioniert immer noch nicht

Lösung:

Loopback Device war nicht konfiguriert. Eintrag in /etc/hosts und /etc/network/interfaces überprüfen. ping localhost muss funktionieren.

14.9 Server not found in Kerberos database

Auf dem Client beim aklog Befehl:

```
aklog
aklog: Couldn't get sed.ethz.ch AFS tickets:
aklog: Server not found in Kerberos database while getting AFS tickets
```

In krb5kdc.log:

```
 129.132.17.2: UNKNOWN_SERVER: authtime 1107795618,  admin@SED.ETHZ.CH
   for krbtgt/ETHZ.CH@SED.ETHZ.CH, Server not found in Kerberos database
```

Mögliche Lösungen:

- DNS/etc/hosts falsch konfiguriert. Server wird in ETHZ.CH statt in SED.ETHZ.CH gesucht.

- In /etc/krb5.conf fehlt der Eintrag für die eigene Domain und das eigene Realm. In dem Fall wird bei Rechnern der DNS-Domain ethz.ch und das Kerberos-Realm ETHZ.CH angenommen, anstatt dem von uns konfigurierten SED.ETHZ.CH. Lösung:

```
[domain_realm]
  .ethz.ch = SED.ETHZ.CH
  ethz.ch  = SED.ETHZ.CH
```

Andere Lösung: Zelle und Realm explizit angeben:

```
aklog sed.ethz.ch -k SED.ETHZ.CH
```

(Frage: Wieso wird default_realm = SED.ETHZ.CH ignoriert?)

Wenn es so auch nicht funktioniert: Auf dem Client ist in /etc/krb5.conf für den kdc ein Alias statt dem canonical Name verwendet.

z.B. gibt es bei uns einen Kerberos-Server mit den Namen newseismo.ethz.ch und newseismo.ethz.ch. /bin/hostname auf dem Server gibt newseismo aus, also muss auf dem Client in /etc/krb5.conf kdc=newseismo.ethz.ch gesetzt werden, und nicht newseismo.ethz.ch.

14.10 Client not found in Kerberos database while getting initial credentials

Problem:

```
newseismo:/mnt/boot/grub# kinit admin && klist
kinit(v5): Client not found in Kerberos database while getting initial credentia
```

Grund:

Principal admin existiert nicht. Aus irgend einem Grund wurde admin/admin statt admin eingetragen.

Lösung:

```
kadmin.local
kadmin.local:  addprinc admin
```

14.11 Couldn't get sed.ethz.ch AFS tickets

Problem:

```
newseismo:~# aklog
aklog: Couldn't get sed.ethz.ch AFS tickets:
aklog: Server not found in Kerberos database while getting AFS tickets
```

Debuginformation anzeigen:

```
newseismo:~# aklog -d
Authenticating to cell sed.ethz.ch (server newseismo.ethz.ch).
We've deduced that we need to authenticate to realm ETHZ.CH.
Getting tickets: afs/sed.ethz.ch@ETHZ.CH
Kerberos error code returned by get_cred: -1765328377
aklog: Couldn't get sed.ethz.ch AFS tickets:
aklog: Server not found in Kerberos database while getting AFS tickets
```

Offensichtlich wird versucht, sich im Realm ETHZ.CH statt SED.ETHZ.CH anzumelden. Angeblich sei das wegen einer Fehlkonfigurtion des Hosts. Der Fehler lässt sich aber umgehen, wenn man die afs Zelle und das Kerberos-Realm explizit angibt:

```
newseismo:~# aklog -d  sed.ethz.ch -k SED.ETHZ.CH
Authenticating to cell sed.ethz.ch (server newseismo.ethz.ch).
We were told to authenticate to realm SED.ETHZ.CH.
Getting tickets: afs/sed.ethz.ch@SED.ETHZ.CH
Identical tokens already exist; skipping.
```

14.12 ioctl failed oder pioctl failed

```
aklog: unable to obtain tokens for cell sed.ethz.ch (status: a pioctl failed).
```

Grund: Eine Inkompatibilität zwischen den verwendeten Versionen von AFS und dem Linux-Kernel.

Lösung: Eine ältere Kernelversion (2.4.x) oder eine neuere AFS-Version und einen ganz neuen Kernel verwenden. Hier funktionieren Kernel 2.4.27 und AFS 1.3.74 gut zusammen. Die gleiche AFS-Version funktionierte nicht richtig mit Kernel 2.6.8.

OpenAFS 1.3.81 funktioniert problemlos mit dem Debian-Kernel 2.6.8-2-686.

14.13 status: Cache Manager is not initialized / afsd is not running

Problem:

```
aklog: unable to obtain tokens for cell sed.ethz.ch (status: Cache Manager is no
```

Lösung:

afsd läuft nicht.

Das Startskript will das libafs modul laden, es sucht es in /lib/modules/‘uname -r‘/fs/. Das Startskript sucht nach openafs.ko, bei uns heisst das Modul aber libafs.ko. Auch wenn das Modul schon geladen ist, gibt /etc/init.d/openafs start einfach auf.

Aus dem fehlerhaften Skript muss der Abschnitt mit load_client durch modprobe libafs ersetzt werden.

14.14 vicepa does not exist

Problem:

```
vos create -server newseismo.ethz.ch -partition /vicepa -name root.afs -cell sed
vos : partition /vicepa does not exist on the server
```

Lösung:

a) /vicepa existiert nicht

b) /vicepa war nicht gemountet, als der AFS-Server gestartet wurde.

```
mount /dev/sda1 /vicepa
/etc/init.d/openafs-fileserver stop
/etc/init.d/openafs-fileserver start
```

14.15 VLDB: no permission access for call

Problem:

```
VLDB: no permission access for call
```

Lösung:

Wenn man mit “kinit admin” und “aklog” angemeldet ist und bosserver ohne -noauth gestartet ist, muss man bei den Befehlen -noauth auch weglassen.

14.16 ldap_add: No such object (32)

Problem:

```
newseismo:/etc/ldap# ldapadd  -x  -D "cn=admin,dc=sed,dc=ethz,dc=ch" -W  -f /roo
Enter LDAP Password:
adding new entry "uid=conti,ou=people,dc=sed,dc=ethz,dc=ch"
ldap_add: No such object (32)
        matched DN: dc=sed,dc=ethz,dc=ch
```

Lösung:

ou=People muss initialisiert werden.

Siehe Abschnitt 6.1 ("Einrichtung des LDAP-Servers").

14.17 no quorum elected

Problem:

```
newseismo:~# pts createuser -name admin -cell sed.ethz.ch -noauth
pts: no quorum elected ; unable to create user admin
```

Lösung:

Bei uns: Falsche IP-Nummer in /etc/openafs/server/CellServDB

Andere Möglichkeiten:

Man liest, es habe etwas mit der Systemzeit auf verschiedenen Maschinen zu tun. Manchmal löse sich das Problem von selbst, wenn man wartet. Leider nicht hier.

Die Uhr von allen Dbservern und Fileservern muss synchronisiert werden.

Auch ein Zeitsprung auf einem Server kann Probleme bereiten. Man muss dann so lange warten, wie der Zeitsprung gedauert hat.

14.18 fs: cell dynroot not in /etc/openafs/CellServDB

Problem:

```
newseismo:/var/log/openafs# fs mkmount /afs/sed.ethz.ch root.cell
fs: cell dynroot not in /etc/openafs/CellServDB
```

Lösung:

Die Option dynroot muss in /etc/openafs/afs.conf und /etc/openafs/afs.conf.client ausgeschaltet werden, danach:

```
/etc/init.d/openafs-client stop
/etc/init.d/openafs-client start
kinit admin
aklog
```

14.19 Cannot contact any KDC

Problem:

```
slave1:/afs/sed.ethz.ch/home/conti# aklog sed.ethz.ch -k SED.ETHZ.CH
aklog: Couldn't get sed.ethz.ch AFS tickets:
aklog: Cannot contact any KDC for requested realm while getting AFS tickets
```

In krb5kdc.log:

```
Apr 20 12:20:06 newseismo krb5kdc[987](info): AS_REQ (7 etypes {18 17
16 23 1 3 2}) 192.168.1.31: NEEDED_PREAUTH: admin@SED.ETHZ.CH for
krbtgt/SED.ETHZ.CH@SED.ETHZ.CH, Additional pre-authentication required
Apr 20 12:20:09 newseismo krb5kdc[987](info): AS_REQ (7 etypes {18 17
16 23 1 3 2}) 192.168.1.31: ISSUE: authtime 1145528409, etypes {rep=16
tkt=16 ses=16}, admin@SED.ETHZ.CH for krbtgt/SED.ETHZ.CH@SED.ETHZ.CH
Apr 20 12:20:20 newseismo krb5kdc[987](info): TGS_REQ (1 etypes {1})
192.168.1.31: UNKNOWN_SERVER: authtime 1145528409,  admin@SED.ETHZ.CH
for afs/sed.ethz.ch@SED.ETHZ.CH, Server not found in Kerberos database
Apr 20 12:20:20 newseismo krb5kdc[987](info): TGS_REQ (1 etypes {1})
192.168.1.31: ISSUE: authtime 1145528409, etypes {rep=16 tkt=1 ses=1},
admin@SED.ETHZ.CH for afs@SED.ETHZ.CH
```

Lösung:

Wir haben zwei Netze: Internet und ein lokales LAN. slave1 ist im lokalen LAN und kommt nur über einen NAT/Masquerading Gateway ins Internet. Der krb und afs Server ist in beiden Netzen. Der Zugriff über die Internetadresse des Servers funktionierte nicht, aber wenn man in /etc/krb5.conf für den kdc die Adresse im lokalen Netz angibt funktioniert alles.

TODO: Weshalb geht es nicht über NAT und die Internetadresse?

14.20 no such partition

Problem: Beim Versuch, eine neue Partition anzulegen, erscheint folgende Fehlermeldung, obwohl unter /vicepb eine Partition gemountet ist.

```
newseismo:~# vos create newseismo /vicepb signals
vos : partition /vicepb does not exist on the server
```

Lösung: Partitionen müssen gemountet werden, bevor der AFS Fileserver gestartet wird. Also muss der Fileserver neu gestartet werden.

```
newseismo:~# /etc/init.d/openafs-fileserver stop
Stopping AFS Server: bosserver.
newseismo:~# /etc/init.d/openafs-fileserver start
Starting AFS Server: bosserver.
```

14.21 Ein User kann sich nicht einloggen

Problem:

Ein User kann sich über ssh nicht einloggen, alle anderen funktionieren. In /var/log/auth.log steht unter anderem folgende Fehlermeldung:

```
Aug  4 11:50:05 newseismo sshd[16759]: pam_krb5: pam_sm_acct_mgmt(ssh conti): en
Aug  4 11:50:05 newseismo sshd[16759]: pam_krb5: pam_sm_acct_mgmt(ssh conti): ex
Aug  4 11:50:05 newseismo sshd[16759]: error: PAM: Authentication service cannot
```

Lösung:

Im Homeverzeichnis des Users befindet sich eine Datei .k5login. Nachdem diese gelöscht wurde funktionierte es bei uns wieder.

14.22 Cron hängt manchmal

Problem:

Cron führt jobs nicht aus, ps -ef zeigt viele Prozesse mit dem Namen /USR/SBIN/CRON (grossgeschrieben).

Das Problem wird auch hier beschrieben:

http://groups.google.ch/group/linux.debian.user/browse_thread/thread/f8466ceeee6

Lösung:

Das Problem scheint in der Zusammenarbeit von Cron mit LDAP zu liegen. Eine neue Version von libnss-ldap muss installiert werden. Ich habe die Quellen für Debian Etch genommen und in Sarge kompililert.

```
echo ‘‘deb-src http://debian.ethz.ch/debian/ testing main  non-free contrib’’ >>
apt-get update
apt-get install cdbs libldap2-dev libsasl2-dev automake1.9
apt-get source --compile libnss-ldap
dpkg -i libnss-ldap*deb
```

14.23 Einzelne Volumes sind nicht mehr erreichbar

Problem:

Einzelne Volumes sind nicht mehr erreichbar. In den Logfiles steht etwas von "volume needs salvage"

Kontrolle, z.B. für das Volume "home":

```
vos examine -id home
```

Lösung:

```
bos salvage -server 129.132.17.20 -all
/etc/init.d/openafs-fileserver stop
/etc/init.d/openafs-fileserver start
```

14.24 Bei ssh login kein aklog, HOME nicht lesbar

Problem:

Bei ssh login kein aklog, HOME nicht lesbar

Lösung:

/lib/security/pam_openafs_session.so fehlt

14.25 asetkey: can't initialize conf dir '/etc/openafs/server'

Problem:

```
asetkey: can't initialize conf dir '/etc/openafs/server'
```

Lösung:

In /etc/openafs/server/ThisCell muss eine AFS-Zelle eingetragen sein.

14.26 Probleme beim Failover auf zweite Maschine

Problem: Timeout auf Client beim Zugriff, manche Serverprozesse laufen nicht.

Lösung: Folgende Befehle auf dem zweiten Server nochmals ausführen (wie bei Grundinstallation auf dem ersten Server):

```
bos create -server newseismo.ethz.ch -instance fs -type fs \
  -cmd /usr/lib/openafs/fileserver \
  -cmd /usr/lib/openafs/volserver \
  -cmd /usr/lib/openafs/salvager \
  -cmd /usr/lib/openafs/vlserver  -cell sed.ethz.ch
```

```
heimers@newseismo:~$ kpasswd
Password for xxxx@SED.ETHZ.CH:
Enter new password: :
Enter it again: :
kpasswd: Connection timed out changing password
```

Lösung:

Der Prozess kadmind läuft nicht.

```
 /etc/init.d/krb5-admin-server start
```

14.27 Server nach Failover nicht mehr erreichbar

Problem:

In einem Failoversystem sollte ein zweiter Server die AFS-Partitionen von einem externen RAID mounten und den Server starten. Nach dem Absturz des einen Servers wird der ander zwar gestartet, aber Clients erhalten keinen Zugriff.

Lösung:

Der Salvager-Prozess wurde nicht gestarter, weil die Datei /etc/openafs/server-local/SALVAGE.fs auf dem alten Server liegt.

Das Verzeichnis /etc/openafs/server-local muss ebenfalls auf gemeinsamem Storage abgelegt werden.

14.28 Disk quota exceeded

Problem: Disk quota exceeded

Lösung:

```
fs setquota /afs/sed.ethz.ch/ 0
```

14.29 aklog: Couldn't figure out realm for cell sed.ethz.ch.

Problem:

```
aklog
aklog: Couldn't figure out realm for cell sed.ethz.ch.
```

Lösung:

In /etc/openafs/CellServDB muss vorne die IP, hinter dem # der Hostname stehen. Im Problemfall stand hinten auch die IP.

Richtig:

```
>sed.ethz.ch
129.132.17.20           #newseismo.ethz.ch
```

14.30 Wrong principal in request

Problem bei kprop:

```
newfront:/etc/krb5kdc# /usr/sbin/kprop -d   -f  /var/lib/krb5kdc/slave_datatrans
/usr/sbin/kprop: Server rejected authentication (during sendauth exchange) while
Generic remote error: Wrong principal in request
```

Lösung:

In /etc/hosts auf dem client muss die IP-Adresse und der Hostname mit Domain stehen. In unserem Problemfall hat kprop nach host/zak@SED.ETHZ.CH gesucht, während in kerberos host/zak.ethz.ch@SED.ETHZ.CH eingetragen war.

14.31 Connection timed out

Problem:

Ohne ersichtlichen Grund kommt es immer wieder zu Fehlermeldungen "Connection timed ou".

Lösung:

Nicht gesichert, eine Vermutung, die in einer Newsgroup genannt wurde:

Der AFS-Cache (bei Debian /var/cache/openafs) ist in nicht konsistentem Zustand (z.B. nach Systemabsturz oder Stromausfall).

Eventuell hilft "fs flush". Sonst muss der AFS-Client gestoppt und dann /var/cache/openafs/* gelöscht werden. Danach den Client wieder starten.

15 FAQ

15.1 Was heisst "key salt"?

In Kerberos wird die Identität dadurch bewiesen, dass man Daten mit einem Schlüssel (Key) verschlüsseln und entschlüsseln kann, den man mit dem KDC teilt.

In Kerberos 5 wird der Schlüssel durch die Funktion string2key() aus dem Passwort und einem Zusatz generiert. Diesen Zusatz nennt man "key salt". In Kerberos 5 wird der Principal (sowas wie dem login-namem) inklusive Kerberos-Realm als "key salt"verwendet.

15.2 Was heisst "kvno"?

"Key version number"

Die kvno gehört zu einem principal, und wird jedes mal erhöht, wenn der Passwort/Verschlüsselungsschlüssel geändert wird.

15.3 Was heisst "sync site"?

"sync site" ist derjenige afs-Server, der änderungen an der Datenbank entgegennehmen kann und diese nacher den anderen Servern zur replikation weiterleitet.

Die verschiedenen AFS-Server bestimmen selbst, welcher "sync site" sein soll. Dies muss auch nicht immer der gleiche sein. Für Details suche in der Dokumentation nach dem UBIK Protokoll.

Folgender Befehl zeigt den "sync host":

```
udebug newseismo  7002
```

Ist sync host 0.0.0.0 liegt ein Problem vor.

Literatur

[1] *Installing OpenAFS*, http://www.debianplanet.org/node.php?id=816

[2] *Kerberos FAQ* http://www.cmf.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html oder http://www.faqs.org/faqs/kerberos-faq/general/

[3] *OpenLDAP: A Quick-Start Guide* http://www.openldap.org/doc/admin23/quickstart.html

[4] *User's Guide* http://www.slac.stanford.edu/comp/unix/afs/transarc/Html/user/user02.htm

[5] *The Moron's Guide to Kerberos* http://www.isi.edu/ brian/security/kerberos.html

[6] *OpenAFS mit MIT Kerberos5* http://de.gentoo-wiki.com/OpenAFS_mit_MIT-Kerberos5

[7] *AFS FAQ* http://www.angelfire.com/hi/plutonic/afs-faq.html

[8] *Administration Reference: Tapeconfig* http://www.openafs.org/pages/doc/AdminReference/

[9] *Dokumentationen rund um AFS* http://fbo.no-ip.org/cgi-bin/twiki/view/Instantafs/DokuMentation