3 Benutzerverwaltung

Systemadministratoren verbringen viel Zeit mit dem Verwalten der Benutzer und ihrer Konten.

Typische Arbeiten sind dabei

- das Anlegen und Löschen von Benutzerkonten,
- die Prüfung der Qualität von Passwörtern,
- Änderungen von Passwörtern, welche die Benutzer vergessen haben sowie
- die Überwachung des von den Anwendern belegten Speicherplatzes.

Wegen ihrer Überlastung brauchen Systembetreuer in vielen Organisationen mehrere Tage, bis sie neuen Mitarbeitern vollen Systemzugang eingerichtet haben – und oft noch länger, bis sie ausscheidenden Mitarbeitern alle Zugänge entzogen haben.

Viele Benutzer neigen dazu, leicht zu erratende Passworte zu wählen. Da dies die Sicherheit des Systems gefährdet, sollten Systemverwalter die Qualität der Passworte regelmäßig überprüfen.

Viele Anwender müssen sich mehrere Dutzend Passworte merken. Da kann es schon passieren, dass sie sich nach einem erholsamen Urlaub nicht mehr an alle erinnern.

Großzügig bemessener Speicherplatz verleitet Benutzer leicht zu einer chaotischen Datenorganisation. Wenn ein Verzeichnis unübersichtlich wird, dann legen sie einfach ein neues an, ohne das alte zu löschen, da sie ja eine der darin enthaltenen Dateien irgendwann noch brauchen könnten.

Für all diese Systemarbeiten gibt es freie und kommerzielle Produkte. Sparsame Systemverwalter setzen u. a.

- das freie Tool Webmin, das Sie unter `http://www.webmin.com/webmin/` finden, oder
- eine freie Version des Lightweight Directory Access Protocol (LDAP) ein.

Systemverwalter mit großen Budgets und Liebe zu kommerziellen Produkten ziehen dagegen vielleicht

- die NDS für Linux von Novell (`http://www.novell.de) oder`
- Volution von Caldera `(http://www.caldera.com)`vor.
- Viele Tools sollten nur von erfahrenen Systemadministratoren installiert und konfiguriert werden.

3.1 Überblick

Die Autoren stellen Ihnen in diesem Kapitel vor, wie Verwalter

mit YaST Benutzer verwalten können (Kapitel 3.2),

mit dem Programm *john* die Qualität der Passworte der Benutzer prüfen können (Kapitel 3.3),

mit Disk-Quotas den Speicherplatz für Benutzer begrenzen können (Kapitel 3.4),

mit einer deutschsprachigen Tool-Sammlung Benutzer in kleinen Umgebungen administrieren können (Kapitel 3.5) und

dies in etwas größeren Umgebung mit dem Lightweight Directory Access Protocol (*LDAP*) tun können.

3.2 Benutzerverwaltung mit YaST

Die Benutzerverwaltung von Linux mit `useradd` ist nicht besonders komfortabel. Einfacher legen Sie neue Benutzer mit YaST an.

Im YaST-Kontrollzentrum finden Sie unter *Sicherheit und Benutzer* • *Benutzer bearbeiten und anlegen* ein Menü für das Verwalten der Benutzer.

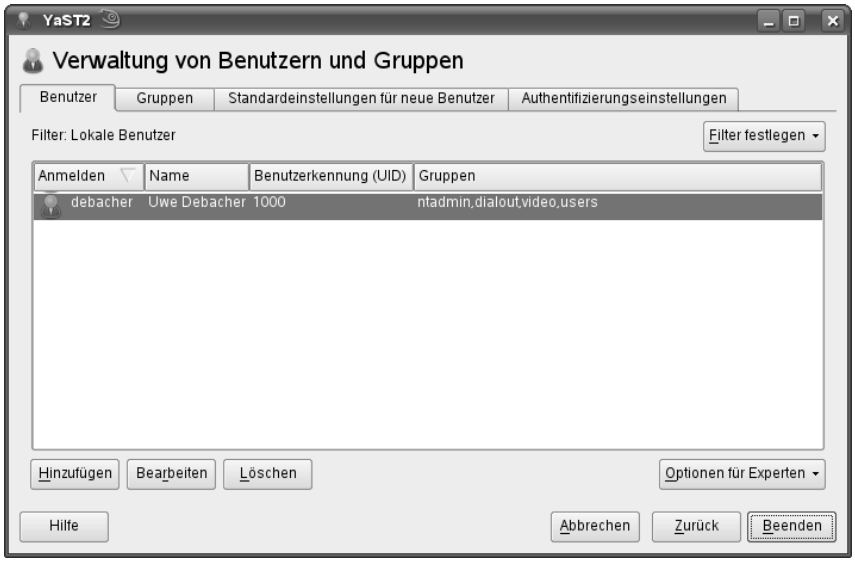

Abbildung 3.1: Benutzerverwaltung mit YaST

In der Benutzerliste finden Sie nur den Benutzer, den Sie bei der Grundinstallation angelegt haben. Diesen Account können Sie über *Bearbeiten* verändern oder über *Löschen* entfernen.

Mit der Funktion *Hinzufügen* richten Sie weitere Benutzer ein.

Abbildung 3.2: Benutzer Hinzufügen mit YaST

Besondere Arbeitsumgebungen wie einen anderen Pfad für das Home-Verzeichnis können Sie über die Schaltfläche *Details* in einem Formular festlegen. Wenn Sie in alle Daten eingegeben haben, richtet ein Klick auf die Schaltfläche *Anlegen* den neuen Benutzer-Account endgültig ein.

3.3 Erkennen schwacher Passwörter

Passwörter in Unix-Systemen können normalerweise noch nicht einmal die Systemverwalter ermitteln, weil Linux die Passwörter nur verschlüsselt ablegt. Die zugehörige Verschlüsselungsfunktion ist eine Einwegfunktion, die kein Entschlüsseln vorsieht. Meldet sich ein Benutzer am System an, verschlüsselt Unix dieses Passwort und vergleicht es mit der in der Shadow-Datei abgelegten Version. Eine Entschlüsselung ist also nicht notwendig.

Es gibt trotzdem theoretisch ein einfaches Verfahren, die Passwörter zu knacken: Sie probieren einfach alle Möglichkeiten durch. Der Aufwand hierfür hängt stark von der Passwortlänge ab, wie Sie an der folgenden Tabelle sehen können. Diese Tabelle geht davon aus, dass 62 verschiedene Zeichen zur Verfügung stehen, die 26 lateinischen

Buchstaben einmal klein, einmal groß und die zehn Ziffern. Weiter geht die Berechnung davon aus, dass Sie 10 Millionen Kennwörter pro Sekunde überprüfen können.

Passwortlänge	*Zahl der möglichen Passwörter*	*Zeitbedarf zum Knacken*
1	62	keiner
2	3844	keiner
3	238.328	keiner
4	14.776.336	1,4 Sekunden
5	916.132.832	1,5 Minuten
6	56.800.235.584	1,5 Stunden
7	3.521.614.606.208	4 Tage
8	218.340.105.584.896	8 Monate
9	13.537.086.546.263.552	43 Jahre
10	839.299.365.868.340.224	2660 Jahre

Tabelle 3.1: Sicherheit in Abhängigkeit von der Passwortlänge

Die Sicherheit eines Passworts hängt nicht nur von seiner Länge, sondern auch stark von den verwendeten Zeichenketten ab. Die folgende Tabelle geht von einer einheitlichen Passwortlänge von 8 Zeichen aus, wobei wieder 10 Millionen Passwörter pro Sekunde geprüft werden.

Zeichensatz	*Zeichenzahl*	*Zahl der möglichen Passwörter*	*Zeitbedarf zum Knacken*
8-Bit ASCII	256	18.446.744.073.709.551.616	58.500 Jahre
7-Bit ASCII	128	72.057.594.037.927.936	228 Jahre
Buchstaben und Ziffern	62	218.340.105.584.896	8 Monate
nur Buchstaben	52	53.459.728.531.456	62 Tage
nur Kleinbuchstaben	26	208.827.064.576	6 Stunden
Wörter aus Wörterbuch	-	ca. 250.000	nahezu keiner

Tabelle 3.2: Sicherheit in Abhängigkeit vom Zeichensatz bei jeweils 8 Zeichen

Da viele Benutzer Passwörter mit deutlich weniger als acht Zeichen benutzen, gibt es eine durchaus realistische Chance, diese zu knacken. Die Chance erhöht sich noch dadurch, dass Einbrecher nicht alle Kombinationen durchprobieren müssen. Viele Anwender benutzen Namen, Telefonnummern oder Ähnliches, die sie sich leicht merken können.

Wenn Sie Ihren Knack-Tests ein Wörterbuch zu Grunde legen, können Sie bei einer Passwortlänge von acht Zeichen bereits in wenigen Minuten Erfolg haben.

Sie können damit zwar nicht die Passwörter aller Benutzer knacken, aber 50 % innerhalb weniger Minuten sind ein durchaus realistischer Wert.

Hinweis: Schon ein einziger geknackter Zugang ist ein Sicherheitsrisiko. Einbrecher, die einen Zugang zu Ihrem System haben, können dort nach weiteren Schwachpunkten suchen.

Sie sollten daher regelmäßig versuchen, die Passwörter Ihrer Benutzer zu knacken, um wenigsten die unsichersten Kandidaten zu ermahnen.

Beim Knacken und beim Ermahnen der Benutzer kann das Programm `john` helfen, das Sie bei OpenSUSE im Paket `john` in der Paketgruppe *Sicherheit* finden. Nach dem Installieren dieses Programms finden Sie das Programm unter `/usr/sbin/john` und seine Komponenten unter `/var/lib/john/`.

Das Programm kann mit einem Wörterbuch arbeiten; eine englische Version liefert es bereits mit. Ein deutsches Wörterbuch müssten Sie dagegen erst erstellen. Hinweise dazu finden Sie im Verzeichnis `/usr/share/doc/packages/john/`.

Auch ohne diesen Aufwand zu treiben, genügt es meist, mit den Daten in den Benutzerdateien zu arbeiten. Damit können Sie Passwörter knacken, die aus Namen oder Variationen davon bestehen.

Wechseln Sie in das Verzeichnis `/var/lib/john/`.

```
cd  /var/lib/john
```

Nun lassen Sie aus `passwd` und `shadow` eine einheitliche Datei montieren; im Beispiel heißt sie `passwd.john`:

```
unshadow  /etc/passwd  /etc/shadow  >  passwd.john
```

Wenn Sie `john` mit den Daten aus dieser Datei arbeiten lassen, werden Sie staunen, wie viele Passwörter er ermittelt.

```
john -single passwd.john
```

Mit diesem Befehl nutzt `john` nur die Benutzerdatenbank als Grundlage, keines der zusätzlich verfügbaren Wörterbücher.

Wenn Sie bereits viele Benutzer angelegt haben, dauert das Knacken schon eine Weile. Wenn Sie den Fortschritt kontrollieren wollen, drücken Sie einmal die Leertaste, worauf `john` den aktuellen Stand anzeigt.

```
Loaded 1037 passwords with 426 different salts (Standard DES
                                                  [24/32 4K])
Burak            (bs1002)
laura            (lc1001)
sandra           (kj1002)
laura            (lt1002)
christi          (sw1002)
gast0            (gast)
ahmad-fa         (ak1005)
```

```
ann-kath         (ag1005)
wolf-die         (wm1004)
walter           (ja1001)
guesses: 10  time: 0:00:00:05 71%  c/s: 370569  trying: &tc3001& -
*j5c*
```

Hier hat `john` nach knapp 5 Sekunden bereits 10 von etwa 1000 Passwörtern geknackt. Bei dem Datenbestand aus dem Beispiel hatte `john` nach knapp 2 Minuten bereits mehr als 70 Passwörter geknackt, und das im einfachsten Modus.

Wenn Sie `john` unterbrechen, setzt er bei einem Neustart seine Arbeit an der Stelle fort, an der Sie ihn unterbrochen hatten. Die bereits geknackten Passwörter hält er in der Datei `john.pot` fest. Falls Sie erneut alle Passwörter testen wollen, müssen Sie diese Datei vorher löschen.

Das Ergebnis der Arbeit von `john`, eine Liste der Benutzerdaten inklusive Passwort im Klartext, können Sie mit

```
john  -show  passwd.john
```

ansehen. John zeigt dabei nur die Accounts, deren Passwort es ermitteln konnte.

Wenn `john` mit der Arbeit fertig ist, können Sie ihn auch veranlassen, eine Mail an alle Benutzer zu schicken, deren Passwörter er knacken konnte. Dazu finden Sie im Verzeichnis ein Programm `mailer`, das Sie zuerst mit

```
chmod  u+x  mailer
```

ausführbar machen und dann folgendermaßen aufrufen, um alle nachlässigen Benutzer zu ermahnen:

```
./mailer  passwd.john
```

Den im Original englischen Text der Mail an die Benutzer kann man in dem Perl-Programm `mailer` relativ leicht ändern. Wenn englischsprachige Warnungen einige Ihrer Benutzer überfordern könnten, sollten Sie den Text übersetzen.

```
#!/bin/bash
#
# This file is part of John the Ripper password cracker,
# Copyright (c) 1996-98 by Solar Designer
#

if [ $# -ne 1 ]; then
        echo "Usage: $0 PASSWORD-FILE"
        exit 0
fi

# There's no need to mail users with these shells
SHELLS=-,/bin/false,/dev/null,/bin/sync

# Look for John in the same directory with this script
```

```
DIR="`echo "$0" | sed 's,/[^/]*$,,'`"

# Let's start
$DIR/john -show "$1" -shells:$SHELLS | sed -n 's/:.*//p' |
(
        SENT=0

        while read LOGIN; do
                echo Sending mail to "$LOGIN"...
# You'll probably want to edit the message below
                mail -s 'Unsicheres Passwort' "$LOGIN" << EOF
Hallo!

Das Passwort für den Account "$LOGIN" ist unsicher. Bitte ändern Sie
es umgehend, sonst macht das Ihr Systembetreuer;-)

Gruss,
        Password Checking Robot
        im Auftrag Ihres Systembetreuers
EOF

                SENT=$(($SENT+1))
        done

        echo $SENT messages sent
)
```

Die Dokumentation von `john` nennt noch mehr Möglichkeiten, um weitere Passwörter zu knacken. Eventuell hilft Ihnen diese Erfahrung, selbst sicherere Passwörter zu verwenden.

Machen Sie Ihren Benutzern immer wieder klar, dass Sicherheit kein Zustand ist, sondern ein anstrengender Prozess. Ein Teil dieses Prozesses ist u. a. die Wahl geeigneter Passwörter.

3.4 Disk-Quotas

Einzelne speicherhungrige Benutzer können die Arbeit auf Linux-Systemen blockieren:

- wenn die Systemverwalter für die Home-Verzeichnisse keine eigene Partition angelegt haben, können sie die gesamte(n) Server-Festplatte(n) füllen und dadurch die Funktionsfähigkeit des Systems erheblich einschränken.
- Liegen die Home-Verzeichnisse in eigenen Partition, so können Vielspeicherer zumindest die Home-Partition so weit mit Daten füllen, dass für keinen Anwender mehr Speicherplatz bleibt.

Zum Schutz vor unmäßigem Verbrauchen von Speicherplatz kann man für jeden Benutzer eine Obergrenze (Quota) für die Nutzung der Festplatten festlegen. Während man für kommerzielle Betriebssysteme eine zusätzliche Quota-Software erwerben muss, enthalten die meisten Linux-Distributionen freie und oft für bestimmte Nutzungsarten kostenlose Quota-Programme.

Die von OpenSUSE gelieferte Version der Quota-Software kommt mit allen wichtigen Linux-Partitionstypen wie `ext2`, `ext3` oder auch `reiserfs` zurecht. Die Software erlaubt Quotas sowohl für Benutzer als auch für Gruppen. Die Beschränkungen gelten jeweils für eine einzelne Partition.

Gruppen-Quotas geben die Summe des Speicherplatzes an, den alle Mitglieder dieser Gruppe gemeinsam belegen dürfen. Diese Werte müssen Sie bei vielen Benutzern daher recht hoch ansetzen.

Mit der Software kann man die individuelle Festplattenkapazität der Benutzer über zwei Angaben einschränken:

- Speicherplatz in Bytes und
- Zahl der Dateien über die Inodes.

Die Beispiele in diesem Kapitel beschränken jeweils den Speicherplatz in Bytes, nicht aber die Zahl der Dateien.

Bei beiden Möglichkeiten können Sie zwei unterschiedliche Grenzen setzen:

- Hard-Limits können Benutzer auf keinen Fall überschreiten,
- Soft-Limits dürfen Benutzer eine bestimmte Zeit (meist eine Woche) lang überschreiten, aber nur bis zum Hard-Limit. Sie bestimmen auch
- die Dauer, für die ein Benutzer das Soft-Limit überschreiten darf.

Bei OpenSUSE finden Sie die Quota-Software im Paket *`quota`* der Paketgruppe *`System`*.

Bevor Sie die Quotas konfigurieren können, müssen Sie noch Module nachinstallieren. Das Quota-System benötigt Unterstützung durch den Kernel. Diese Unterstützung hat OpenSUSE zwar eingebaut, aber als eigenständiges Modul. Genau dieses Modul müssen Sie noch laden lassen. Gehen Sie dazu im YaST-Kontrollzentrum auf *System • Editor für /etc/sysconfig-Dateien* und dort auf *System • Kernel* und erweitern dort die Variable *INITRD_MODULES*. Normalerweise steht dort z. B. *ext3*, eventuell sogar einige Einträge mehr. Zu den Einträgen gehören jeweils Module, die der Kernel gleich beim Systemstart laden muss, vor der eigentlichen Modulverwaltung. Hier finden Sie also die Module für bestimmte Festplattenhardware, z. B. *SCSI* und besondere Partitionstypen, z. B. *reiserfs*.

Ergänzen Sie die Zeile um die Angabe `quota_v2` und lassen bitte zwischen den bisherigen Einträgen und Ihrer Eingabe ein Leerzeichen. Abschließend müssen Sie noch die `initrd`-Datei neu erzeugen lassen, welche die Module für den Systemstart enthält.

```
mkinitrd
```

Normalerweise installiert OpenSUSE bei der Standardinstallation den Boot-Manager `grub`, der die Veränderungen automatisch registriert. Falls Sie jedoch noch `lilo` als Bootmanager benutzen, müssen Sie nun `lilo` noch einmal von der Konsole aus aufrufen, damit der Bootmanager die veränderte `initrd` übernimmt.

Nach einem Reboot ist dann die Änderung aktiv und das Modul für das Quota-System geladen. Statt den PC zu rebooten, kann man das Modul auch manuell mit `modprobe` laden:

```
modprobe -v quota_v2
```

Um die Quota-Unterstützung für eine Partition zu aktivieren, müssen Sie die Datei `/etc/fstab` erweitern, die alle Dateisysteme enthält, welche das Linux-System beim Hochfahren automatisch mounten soll.

Die Datei können Sie entweder direkt mit Ihrem Lieblingseditor bearbeiten oder etwas sicherer vom YaST-Kontrollzentrum aus über *System* • *Partitionierer*. Die Warnung von YaST, »Verwenden Sie das Programm nur, wenn Sie mit dem Partitionieren von Festplatten vertraut sind.« sollten Sie auf alle Fälle ernst nehmen.

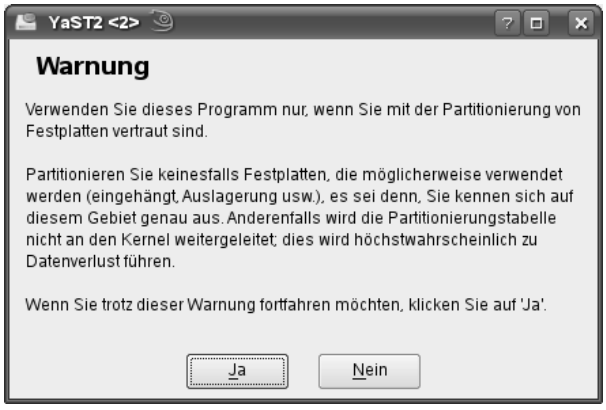

Abbildung 3.3:
Partitionieren • Warnung

Wenn Sie sicher sind, dass Sie Partitionen verändern wollen, klicken Sie auf *Ja*. YaST öffnet eine Liste aller vorhandenen Partitionen, aus der Sie die Home-Partition (`/dev/hda9`) auswählen. In dem folgenden Formular ist in diesem Zusammenhang nur ein Button wichtig.

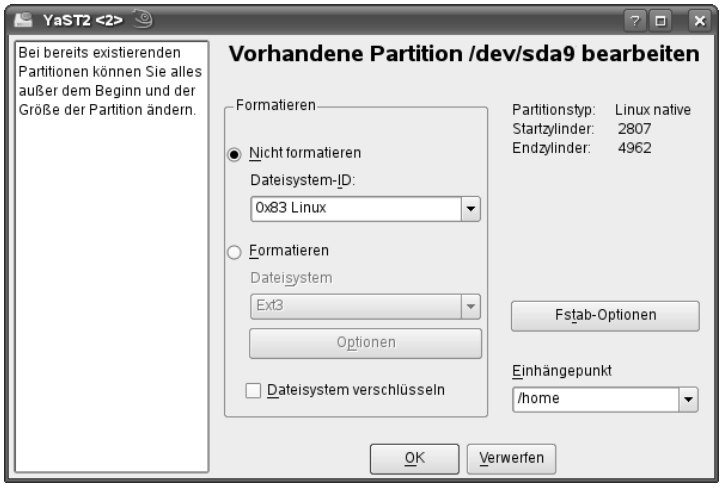

Abbildung 3.4: Partitionieren • Home-Partition

Sie sollten hier nur auf *Fstab-Optionen* klicken. Die benötigte Einstellung `usrquota`, `grpquota` können Sie in dem dann folgenden Formular unterbringen.

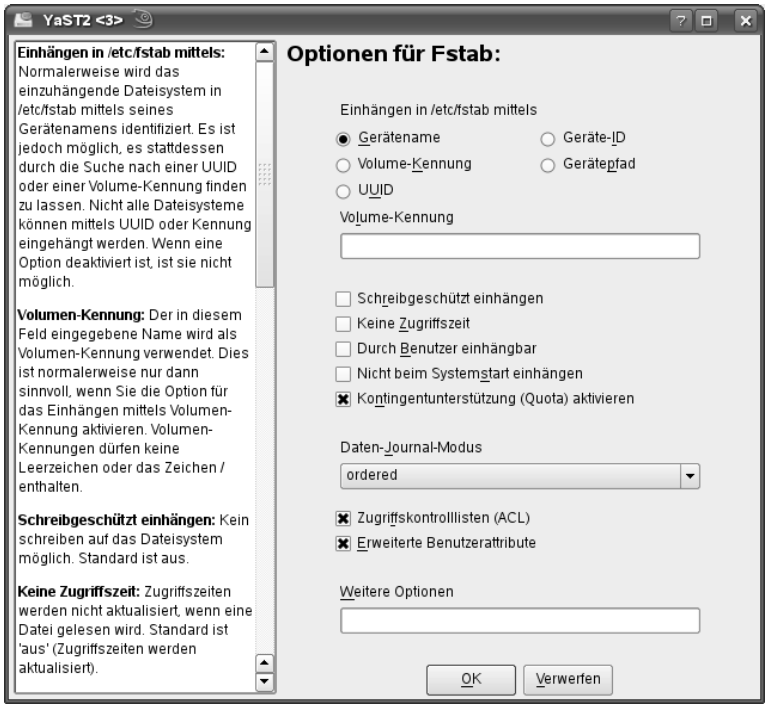

Abbildung 3.5: Partitionieren • Optionen

Entscheidend ist hier das Feld *Kontingentunterstützung (Quota) aktivieren*. Dieses Feld enthält normalerweise kein Kreuzchen. Setzen Sie es hier also ein.

Damit aktivieren Sie für diese Partition sowohl eine Benutzer-Quota als auch eine Gruppen-Quota.

Wenn Sie dann auf *Ok* klicken und das Partionierungsmenü verlassen, ändert YaST die Datei `/etc/fstab`, nachdem es Sie vorher noch einmal gewarnt hat.

Tipp: Wenn Sie die Datei `/etc/fstab` direkt mit einem Editor bearbeiten, dürfen bei der Aufzählung `acl,user_xattr,usrjquota=aquota.user,grpjquota=aquota.group,jqfmt=vfsv0` keine Leerzeichen zwischen diesen Parametern stehen!

Bei einer Installation mit der im Kapitel 2 vorgeschlagenen Partitionierung hat diese Datei den folgenden Inhalt:

```
/dev/hda6   /              ext3    acl,user_xattr        1 1
/dev/hda9   /home          ext3    acl,user_xattr        1 2
/dev/hda7   /tmp           ext3    acl,user_xattr        1 2
/dev/hda8   /var           ext3    acl,user_xattr        1 2
/dev/hda5   swap           swap    pri=42                0 0
devpts      /dev/pts       devpts  mode=0620,gid=5       0 0
proc        /proc          proc    defaults              0 0
usbfs       /proc/bus/usb  usbfs   noauto                0 0
sysfs       /sys           sysfs   noauto                0 0
```

Um die Nutzung von Partitionen zu beschränken, müssen Sie das Schlüsselwort `usrquota` für Beschränkungen auf Benutzerebene oder `grpquota` für Beschränkungen auf Gruppenebene hinzufügen. Sie können auch beide Beschränkungen gleichzeitig aktivieren.

```
/dev/hda6   /              ext3    acl,user_xattr        1 1
/dev/hda9   /home          ext3
acl,user_xattr,usrjquota=aquota.user,grpjquota=aquota.group,jqfmt=vfs
v0 1 2
/dev/hda7   /tmp           ext3    acl,user_xattr        1 2
/dev/hda8   /var           ext3    acl,user_xattr        1 2
/dev/hda5   swap           swap    pri=42                0 0
devpts      /dev/pts       devpts  mode=0620,gid=5       0 0
proc        /proc          proc    defaults              0 0
usbfs       /proc/bus/usb  usbfs   noauto                0 0
sysfs       /sys           sysfs   noauto                0 0
```

Da Sie das Dateisystem geändert haben, müssen Sie es neu mounten, am einfachsten durch Booten des Linux-Servers.

Tipp: Beschränken Systemverwalter den Speicherplatz nur für ganze Benutzergruppen mit Gruppen-Quotas, verhindert dies nicht, dass ein einzelner Benutzer den gesamten zulässigen Speicherplatz belegt und damit die Arbeit der anderen Benutzer blockiert. Benutzer-Quotas sind auf alle Fälle zum Sicherstellen eines geordneten IT-Betriebs geeigneter als Gruppen-Quotas.

Nach dem Neustart des Linux-Servers können Sie die Quota-Software den momentanen Belegungsstand der Festplatte erfassen lassen. Dazu geben Sie ein:

```
quotacheck  -vagu
```

Der Parameter `v` bewirkt eine ausführliche Ausgabe, mit dem Parameter `a` überprüft das Programm alle Partitionen, für die in der Datei `/etc/fstab` eine Quota-Unterstützung angegeben ist. Den Schalter `g` benötigen Sie für Gruppen-Quotas und den Schalter `u` für User-Quotas (also Benutzer-Quotas).

Sollte die Partition aktiv sein, so verweigert `quotacheck` seinen Dienst. Sie können dann entweder dafür sorgen, dass die Partition nicht aktiv ist oder zusätzlich den Schalter `m` mit angeben.

Das Untersuchen der Festplatte kann je nach Belegungsgrad einige Minuten dauern. Danach hat das Programm für jede quotierte Partition die Belegungsdaten in die Dateien `aquota.user` und `aquota.group` im Wurzelverzeichnis der jeweiligen Partition geschrieben.

Nach diesen Vorbereitungen können Sie die Quotas scharf schalten, falls YaST das nicht schon für Sie gemacht hat. Dazu starten Sie das YaST-Kontrollzentrum, gehen dort in das Menü *System* • *Runlevel-Editor* • *Expertenmodus* und aktivieren hier den Dienst *boot.quota* für die Runlevel B (Start beim Booten), indem Sie den Leuchtbalken auf die Zeile mit *quota* bringen und dann das mit *B* beschriftete Kästchen anklicken. Anschließend können Sie den Dienst auch gleich starten: Klicken Sie dazu auf *Starten/Anhalten/Aktualisieren* und wählen dann *Starten*. Damit ist der Dienst aktiv.

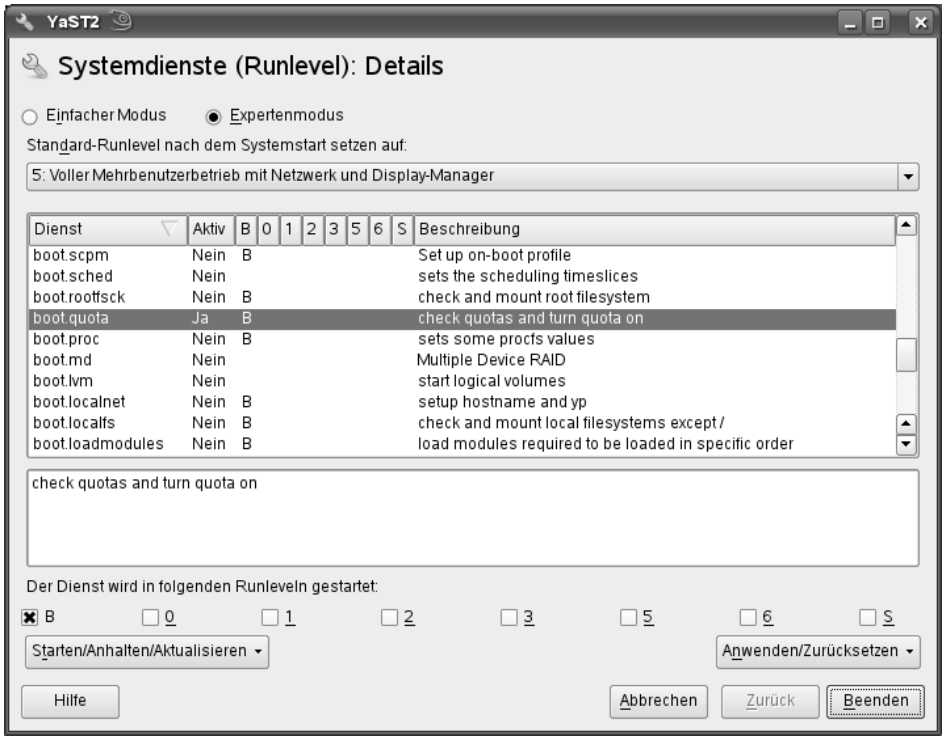

Abbildung 3.6: Runlevel-Editor QUOTA

Um die Funktion Ihrer Quotas zu testen, richten Sie (als *root*) für einen Ihrer Benutzer eine Beschränkung ein:

```
edquota  -u  debacher
```

Daraufhin startet der von Ihnen eingestellte Editor mit folgendem Text:

```
Disk quotas for user debacher (uid 1000):
  Filesystem   blocks    soft    hard   inodes   soft   hard
  /dev/hda9      1028       0       0      167      0      0
```

Der Benutzer belegt 1028 KByte Speicherplatz auf dem System mit 167 Dateien. Verändern Sie die Einstellungen zu

```
Disk quotas for user debacher (uid 1000):
  Filesystem   blocks    soft    hard   inodes   soft   hard
  /dev/hda9      1028    4000    5000      167      0      0
```

Damit erlauben Sie dem Benutzer, maximal 5000 KByte Speicherplatz zu belegen.

Der Wert 0 bedeutet hier immer keine Beschränkung. Ein Hard-Limit können Benutzer auf keinen Fall überschreiten, ein Soft-Limit (hier 4000) nur für eine einstellbare Dauer. Diesen Zeitrahmen konfiguriert man mit `edquota -t`.

Melden Sie sich nun mit dem Benutzernamen an, für den Sie soeben die Beschränkungen erstellt haben. Jeder Benutzer kann seine eigenen Werte abfragen mit:

```
quota
```

Das erzeugt die folgende Ausgabe:

```
Disk quotas for user debacher (uid 1000):
 Filesystem  blocks  quota  limit  grace  files  quota  limit  grace
  /dev/hda9    1028   4000   5000           167      0      0
```

Der Benutzer belegt momentan mit 167 Dateien 1028 KByte Speicherplatz. Er darf beliebig viele Dateien anlegen, aber maximal 5000 KBytes verbrauchen.

Das Soft-Limit ist nicht erreicht, damit entfällt auch die Angabe einer Gnadenfrist (`grace`) für das noch erlaubte Überschreiten dieses Limits.

Versuchen Sie nun, das Limit zu überschreiten, indem Sie große Dateien erstellen oder kopieren. Im einfachsten Fall geht das mit folgendem Befehl:

```
dd  if=/dev/zero  of=/home/debacher/test
```

Damit kopieren Sie von dem Gerät, welches ständig Nullen liefert, in eine beliebige Datei, hier `/home/debacher/test`. Dieser Kopiervorgang läuft so lange, bis die Beschränkung erreicht oder die Festplatte voll ist.

Nach kurzer Zeit sollten Sie eine Fehlermeldung erhalten:

```
hda9: write failed, user block limit reached.
dd: Schreiben in "/home/debacher/test": Der zugewiesene Plattenplatz
(Quota) ist überschritten
569+0 Datensätze ein
568+0 Datensätze aus
290816 Bytes (291 kB) kopiert, 0,0903587 s, 3,2 MB/s
```

Ein erneuter Aufruf von `quota` liefert jetzt als Ausgabe:

```
Disk quotas for user debacher (uid 1000):
 Filesystem  blocks  quota  limit  grace  files  quota  limit   grace
  /dev/hda9    5000*  4000   5000           194      0      0
```

Die Datei `test` hat eine Größe von etwa 5 MB angenommen, danach hat die Quota-Begrenzung den Kopiervorgang abgebrochen.

Die Quota-Begrenzung ist damit funktionsfähig und kann eingesetzt werden.

Leider bietet die in der OpenSUSE-Distribution enthaltene Quota-Software keine Möglichkeit, einen Standardwert für alle Benutzer festzulegen. Dies kann in der betrieblichen Praxis auch sinnvoll sein, wenn Sie den Vorstand ihres Unternehmens nicht zu sehr gängeln wollen. Daher müssen Systemverwalter die User-Quotas für jeden Benutzer einzeln festlegen oder ggf. mit dem Befehl `edquota` vervielfältigen.

Um die für einen Benutzer (hier `debacher`) definierte Quota auf den Benutzer `schultz` zu übernehmen, geben Sie den Befehl:

```
edquota  -p  debacher  schultz
```

3.5 Die Linuxbu.ch/Tools

Die Linuxbu.ch/Tools sind eine bewährte, leicht konfigurierbare Sammlung einfacher deutschsprachiger Administrations-Programme mit Browserschnittstelle.

Diese Tools erfordern nur einen geringen Installationsaufwand und nehmen keine weiteren Veränderungen am System vor. Sie unterstützen das Arbeiten mit Changed-Root-Umgebungen (siehe Kapitel 7, »Dateiarchive per FTP bereitstellen«) und den Umgang mit Disk-Quotas (siehe letzter Abschnitt).

Weiterhin unterstützen die Tools das Arbeiten mit verschlüsselten Passwörtern, deren Bedeutung Sie im Kapitel 9 (»Linux als File- und Print-Server für Windows-Clients«) kennen lernen werden.

Sie arbeiten mit drei Benutzergruppen, denen Sie unterschiedliche Rechte zuordnen können:

- `ntadmin`
- `leiter`
- `mitarbeiter`

Jede der drei Gruppen hat unterschiedliche Zugriffsrechte auf die Funktionen. *Mitarbeiter* können mit den Tools lediglich ihr eigenes Passwort verändern, *leiter* können zusätzliche *mitarbeiter*-Accounts einrichten und die Internet-Verbindung aktivieren sowie Gruppen einrichten. Die Update-Funktion können hingegen nur Angehörige der Gruppe *ntadmin* nutzen.

Hinweis: Die ersten Versionen der Linuxbu.ch/Tools haben statt der Gruppe `ntadmin` einfach `admin` benutzt. In gemischten Umgebungen benötigen Windows-Rechner für Administrationszwecke jedoch unbedingt die Gruppe `ntadmin`.

Die Tools bieten momentan folgende Funktionen:

- Eigenes Passwort ändern (alle Benutzer),
- Gruppenverwaltung (*ntadmin*),
- Benutzerverwaltung (*ntadmin* und *leiter*),
- Internetverbindung auf- und abbauen (*ntadmin* und *leiter*),
- Software-Update (*ntadmin*).

Die Linuxbu.ch/Tools ändern an keiner Stelle die Konfiguration Ihres Rechners oder der Software. Verwalter können sie einfach erweitern und anpassen und müssen lediglich den Webserver Apache so konfigurieren, dass er die Programme aus dem Verzeichnis `/srv/www/htdocs/tools` ausführt.

Hinweis: Da SUSE den Webserver in der Standardinstallation nicht mehr einrichtet, müssen Sie den Apache Webserver zuerst installiert haben. Eine ausführliche Beschreibung dazu lesen Sie im Kapitel 6 dieses Buches.

Sie können die Software vom Server zum Buch (`www.linuxbu.ch`) beziehen und kostenlos nutzen. Installieren Sie sie in drei Schritten:

- Auspacken des Archivs `tools4_3.tgz` und Initialisieren der Programme,
- Erweitern der Apache-Konfigurationsdatei und
- Einrichten von Administratoren-Accounts und Tools-Gruppen.

3.5.1 Auspacken des Archivs und Initialisieren der Programme

Laden Sie die Datei `tools4_3.tgz` vom Server `www.linuxbu.ch` und speichern Sie sie im Verzeichnis `/srv/www/htdocs`. Wechseln Sie in dieses Verzeichnis und entpacken Sie die Datei mit:

```
tar  xvfz  tools4_3.tgz
```

Dabei entsteht ein Verzeichnis `tools`, in das Sie nun wechseln:

```
cd  tools
```

Der größte Teil der Tools besteht aus Programmen in der Programmiersprache Perl. Diese Programme erkennen Sie an der Endung `.pl`. Für viele Funktionen benötigen die Linuxbu.ch/Tools die besonderen Rechte des Benutzers *root*. Diese Rechte geben Sie den Perl-Programmen, indem Sie im Verzeichnis `tools` als Benutzer *root* folgenden Befehl eingeben:

```
./makecgi
```

`makecgi` erstellt nach einer Sicherheitsabfrage zu jedem Programm mit der Endung `.pl` ein C-Programm mit der Endung `.cgi`, das diese besonderen Rechte besitzt.

Sollten Sie beim Aufruf des Programms Fehlermeldungen der Art

```
./makecgi: line 30: gcc: command not found
```

bekommen, dann ist auf Ihrem Rechner der C-Compiler `ggc` noch nicht eingerichtet. Sie müssen dann das Paket *gcc* nachträglich installieren. Sie finden das Paket in der Paketgruppe *Programmierung*.

Sofern der C-Compiler vorhanden ist, kann `makecgi` seiner Arbeit nachgehen.

```
makecgi - erstellt die .cgi Dateien.

Grundlage ist die Datei source/setroot.c
Alle bestehenden .cgi Dateien werden ueberschrieben.

Sind Sie sich sicher, dass Sie fortfahren moechten ? [J/Y/N] j
Mache admin/internet/index.cgi
Mache admin/index.cgi
Mache admin/passwd/index.cgi
Mache admin/passwd/chpasswd.cgi
Mache admin/gruppen/shgroupdata.cgi
Mache admin/gruppen/shgrouplist.cgi
Mache admin/gruppen/newgroup.cgi
Mache admin/gruppen/addgroup.cgi
Mache admin/gruppen/delgroup.cgi
Mache admin/update/index.cgi
Mache admin/benutzer/shuserdata.cgi
Mache admin/benutzer/shuserlist.cgi
Mache admin/benutzer/newuser.cgi
Mache admin/benutzer/deluser.cgi
Mache admin/benutzer/multiadd.cgi
Mache admin/benutzer/chuserdata.cgi
Mache admin/benutzer/adduser.cgi
Mache admin/benutzer/shuser.cgi
```

Damit sind die Tools einsatzbereit und Sie können diese in die Konfiguration des Webservers einbinden.

3.5.2 Erweitern der Apache-Konfigurationsdatei

Im Verzeichnis `/srv/www/htdocs/tools/` finden Sie die Datei `httpd.conf.erg` mit den notwendigen Ergänzungen für die Konfigurationsdatei des Apache-Servers.

```
#
# Erweiterung fuer die Linuxbu.ch/Tools
# einfach ueber
# YaST->Editor fuer /etc/sysconfig->Network->WWW->Apache2
# den vollen Pfad zu dieser Datei in die sysconfig
# aufnehmen:
#
# APACHE_CONF_INCLUDE_FILES="/srv/www/htdocs/tools/httpd.conf.erg"
#
# anschliessend den Apache neu starten
#
#

<Directory /srv/www/htdocs/tools/admin>
Addtype application/x-httpd-cgi .cgi
```

```
Options Indexes FollowSymLinks EXECcgi
authType Basic
authuserFile /etc/apache2/yfh.pwd
authName LinuxBuchTools
require valid-user
</Directory>

<Directory /srv/www/htdocs/tools>
Addtype application/x-httpd-cgi .cgi
Options Indexes FollowSymLinks EXECcgi
</Directory>
```

Zum Aktivieren dieser Änderung müssen Sie anschließend im YaST-Kontrollzentrum unter *System* • *Editor für /etc/sysconfig-Daten* • *Network* • *WWW* • *Apache2* für die Variable `APACHE_CONF_INCLUDE_FILES` den Wert `/srv/www/htdocs/tools/httpd.conf.erg` angeben und damit die Erweiterung in die Konfiguration des Webservers einbinden.

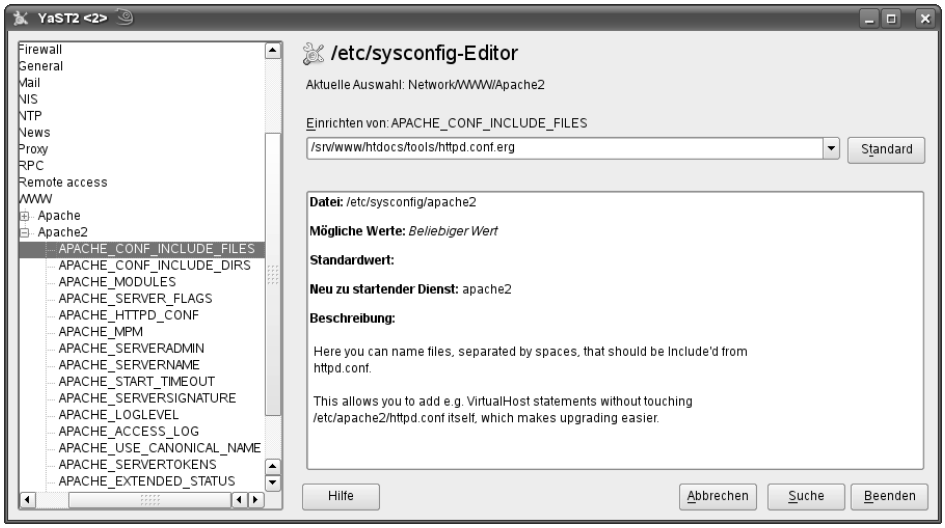

Abbildung 3.7: Eigene Konfigurationsdatei einbinden

Damit binden Sie die mit den Tools mitgelieferte Konfigurationsdatei in die Konfigurationsdatei des Webservers ein, ohne diese selber bearbeiten zu müssen. Genauere Informationen über den Webserver finden Sie im Kapitel 6, »Informationen per Webserver verteilen«.

Durch diese Ergänzungen führt Apache die Programme im Verzeichnis `tools` aus und authentifiziert Benutzer für alle Zugriffe auf die Linuxbu.ch/Tools.

Nach diesen Änderungen müssen Sie den Apache neu starten:

```
rcapache restart
```

3.5.3 Einrichten von Administrator-Accounts und Tools-Gruppen

Für die Nutzung der Tools müssen Sie die zwei Gruppen

- *leiter*
- *mitarbeiter*

anlegen und mindestens einen Administrator-Account einrichten.

Um die Verwaltungs-Funktionen leiten zu können, sollten Sie sich selbst mit Ihrem persönlichen Account (nicht *root*) in die Gruppe ntadmin aufnehmen.

Am einfachsten geht das mit dem `usermod`-Befehl wie hier im Beispiel:

```
usermod -G ntadmin debacher
```

Im YaST-Kontrollzentrum gehen Sie dafür auf *Sicherheit und Benutzer* • *Benutzer- und Gruppenmanagement* und dort auf den Reiter *Gruppen*. Um alle Gruppen sehen zu können, klicken Sie hier auf *Filter festlegen* • *Systemgruppen,*. Dann wählen Sie die Gruppe *ntadmin* aus und *Bearbeiten*. Hier müssen Sie nun die Checkbox vor Ihrem Benutzer-Account aktivieren und sodann die Konfiguration mit *Weiter* beenden.

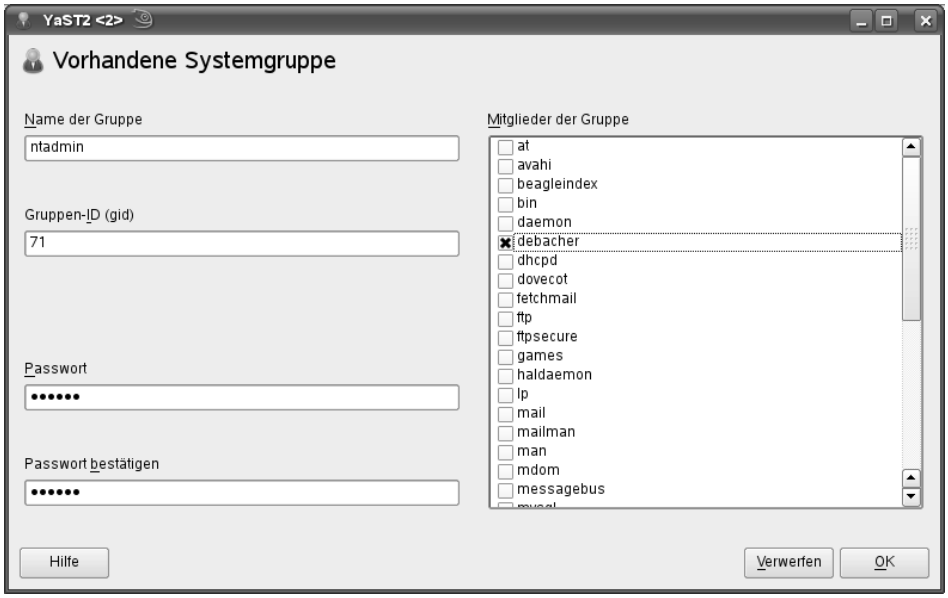

Abbildung 3.8: YaST: Hinzufügen zur Gruppenverwaltung

Starten Sie dann auf einem über das Netz angeschlossenen Rechner einen Browser und rufen Sie die URL `/tools/` auf dem Linux-Server auf, auf dem Sie die Tools ausführen, hier `http://192.168.1.2/tools/` (auch der letzte Schrägstrich ist wichtig).

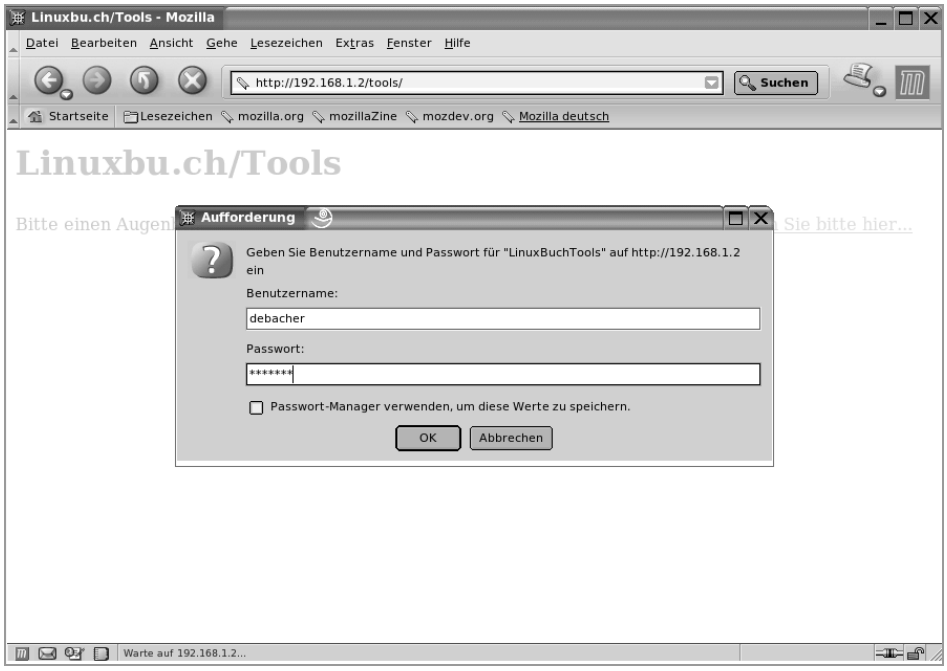

Abbildung 3.9: Tools: Anmeldung

Im Dialogfenster geben Sie Ihren Benutzernamen und Ihr Passwort ein. Danach steht Ihnen das Hauptmenü zur Verfügung.

Dort gehen Sie zunächst auf *Gruppenverwaltung* und dann auf *Neue Gruppe anlegen*. Hier können Sie nacheinander die Gruppen *leiter* und *mitarbeiter* anlegen.

Abbildung 3.10: Tools: Hauptmenü

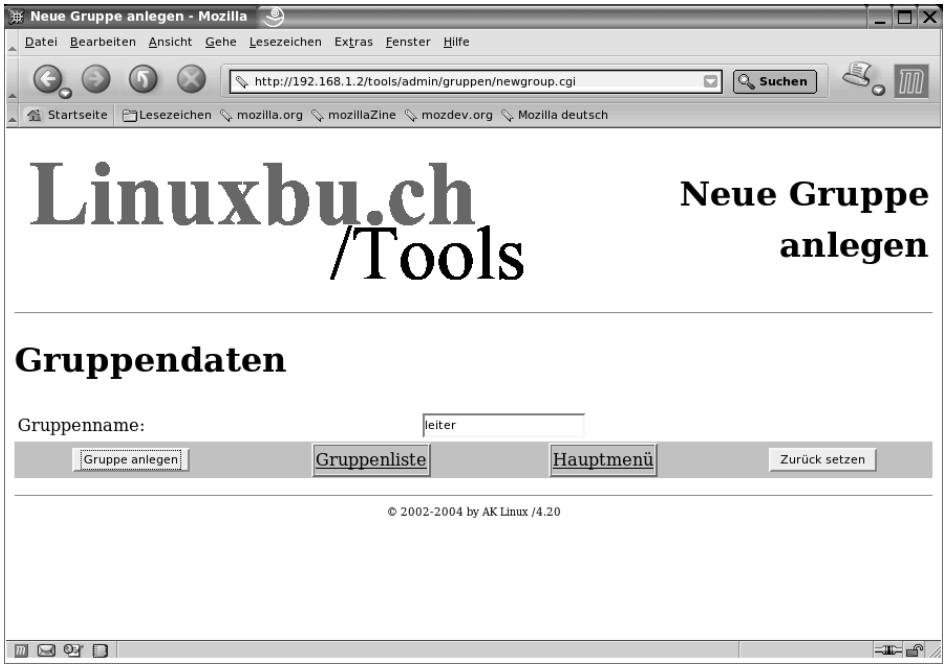

Abbildung 3.11: Tools: Neue Gruppe anlegen

Nach dem Anlegen dieser beiden Gruppen sollte die Gruppenliste wie im nächsten Fenster aussehen:

Abbildung 3.12: Tools: Gruppenliste

Um abschließend die Angaben für Ihren eigenen Account zu vervollständigen, gehen Sie auf *Benutzerverwaltung*, dort auf *Benutzerliste,* und klicken dort Ihren Benutzer-Account an.

Sie sollten vor allem darauf achten, dass Sie auch für sich eine *Abteilung* und Ihren vollen Namen angeben, da die Tools Ihren Namen bei allen Benutzern eintragen, die Sie mit den Linuxbu.ch/Tools anlegen.

Wenn Sie die Daten eingegeben haben, klicken Sie auf *Daten ändern*, worauf das Programm bestätigt, dass es die Daten übernommen hat.

Abbildung 3.13: Tools: Daten ändern

Abbildung 3.14: Tools: Daten geändert

Damit sind die Linuxbu.ch/Tools installiert und einsatzbereit.

3.5.4 Anlegen von Benutzern mit den Tools

Alle Administratoren und die Leiter können mit den Tools jetzt Benutzer einrichten. Nur Administratoren können Leiter einrichten. Die Administratoren haben vollen Zugriff auf alle Benutzer und können deren Daten sowie Passwörter ändern. Die Leiter können nur die Daten (einschließlich Passwort) der Mitarbeiter ändern, die sie selber eingerichtet haben.

Legen Sie zuerst die Abteilungsleiter an, im Beispiel den *Klaus Sparsam*. Gehen Sie dazu auf *Benutzerverwaltung* • *Benutzer anlegen* und füllen das Formular nach dem Muster wie in der Abbildung 3.15 aus.

Zwingend erforderlich ist nur die Angabe der Abteilung und des vollständigen Namens. Wenn Sie keine weiteren Daten angeben, erzeugen die Tools den Login-Namen aus den Initialen und einer laufenden Nummer, in diesem Fall also `ks1001`. Als Anfangs-Passwort stellen die Tools den Vornamen `klaus` ein. Wenn Sie andere Login-Namen und Passwörter für Ihre Benutzer haben möchten, müssen Sie diese in die dafür vorgesehenen Felder eintragen.

Abbildung 3.15: Tools: Benutzer anlegen, hier Abteilungsleiter

Wenn Sie die Eingaben für einen Benutzer abgeschlossen haben, startet ein Klick auf *Benutzer anlegen* das Erstellen des Benutzer-Accounts.

Die Tools legen auch das Home-Verzeichnis des Benutzers an, in diesem Fall wäre das `/home/ksparsam`. Zusätzlich können die Tools auch Quotas für die neuen Benutzer anlegen. Dazu müssen Sie für einen Beispiel-Account die Quotas sorgfältig konfigurieren

und diesen Account den Tools als Muster nennen. Die Einstellungen des Musters übernimmt das Programm dann für alle neuen Benutzer.

Um die Quota-Unterstützung zu aktivieren, müssen Sie die Konfigurationsdatei `/srv/www/htdocs/tools/config.pl` bearbeiten.

Die Quota-Untersützung aktivieren Sie, indem Sie in der drittletzten Zeile das Kommentarzeichen # entfernen und den Benutzernamen `beispiel` durch einen passenden Benutzer ersetzen.

`/srv/www/htdocs/tools/config.pl` (Auszug, Ende der Datei):

```
# $FIRST_CH_UID gibt die UserID an, ab der Benutzer zum Aendern
angezeigt
# werden. Wenn man das Veraendern/Loeschen des root-Account
verhindern moechte,
# sollte man diesen Wert entsprechend hoch setzen.
$FIRST_CH_UID = 1000;

# $LAST_CH_UID gibt die letzte UID an, nach der Benutzer zum Xndern
nicht mehr
# angezeigt werden. (nobody hat 65534)
$LAST_CH_UID = 10000;

# $FIRST_NEW_UID gibt die erste UID an, die fuer neue Benutzer
vergeben wird.
$FIRST_NEW_UID = 1000;

# $FIRST_CH_GID gibt die GruppenID an, ab der Gruppen verwendet
werden
# duerfen. Zum Aendern der Gruppendaten, oder zum Aendern von
Benutzerdaten.
$FIRST_CH_GID = 70;

# $LAST_CH_GID gibt die Letzte GruppenID an, bis zu der Gruppendaten
ver-
# aendert werden duerfen, oder Gruppendaten fuer Benutzer verwendet
werden
# duerfen.
$LAST_CH_GID = 10000;

# $NEWUSER_SHELL gibt an, welche Shell ein Neuer Benutzer
Standartmaessig
# bekommt.
$NEWUSER_SHELL = "/usr/bin/passwd";

# $USERADMINPFAD gibt den Pfad zum Benutzerverwaltungsmodul an.
$USERADMINPFAD = "benutzer/";
```

```
# $QUOTAUSER gibt den Benutzer an, dessen Quotas kopiert werden
#$QUOTAUSER="beispiel";

# $INTERFACE gibt an, ueber welches Geraet die Internetverbindung
laeuft
$INTERFACE="ippp0";
```

Machen Sie sich ruhig auch mit den anderen Konfigurationseinstellungen in dieser Datei vertraut, sie ist ausführlich kommentiert.

3.5.5 Internet Start/Stop

Mit den Linuxbu.ch/Tools kann man festlegen, welche Benutzer über das lokale Netz das Internet anwählen können. In der Grundeinstellung können diese Funktion alle Mitglieder der Gruppen *ntadmin* und *leiter* aufrufen.

Abbildung 3.16: Tools: Internet-Verbindung

Wollen Sie dies erweitern oder einschränken, so müssen Sie die Datei `modinfo.dat` im Verzeichnis der jeweiligen Funktion, hier `/srv/www/htdocs/tools/admin/internet/modinfo.dat`, bearbeiten:

```
index.cgi
Internet Start/Stop
Starten/Stoppen der Internet-Verbindung
1
1
0
0
```

```
0
/htmldoc/mods/internet.html
# Ende der Datei
```

Der Aufbau dieser Konfigurationsdatei ist immer gleich:

- Zeile: Startprogramm des Moduls
- Zeile: Kurztext für das Menü
- Zeile: Langtext für die Statuszeile im Menü
- Zeile: Ausführungsrechte für `ntadmin 0 = nein, 1 = ja`
- Zeile: Ausführungsrechte für `leiter 0 = nein, 1 = ja`
- Zeile: Ausführungsrechte für `mitarbeiter 0 = nein, 1 = ja`
- Zeile: Logging für `Aktionen 0 = nein, 1 = ja`
- Zeile: Logging für `Fehler 0 = nein, 1 = ja`
- Zeile: frei
- Zeile: Hilfetext (spätere Erweiterung)

Entscheidend für das Vergeben von Rechten sind die Zeilen 4, 5 und 6. Hier stehen die Werte `1` und `0`. Damit verbieten Sie nur den Mitgliedern der Gruppe *mitarbeiter*, eine Verbindung aufzubauen. Wollen Sie erlauben, dass auch diese die Funktion nutzen, so müssen Sie die erste `0` durch eine `1` ersetzen.

Die Internet-Einwahl kann sehr unterschiedlich erfolgen, per Modem, ISDN oder T-DSL. Die Linuxbu.ch/Tools erwarten daher, dass Sie in der Konfigurationsdatei das Interface korrekt angegeben haben.

`/srv/www/htdocs/tools/config.pl` (Auszug, Ende der Datei):

```
# $INTERFACE gibt an, über welches Geraet die Internetverbindung
laeuft
$INTERFACE="ippp0";
```

Die Tools benutzen für die Steuerung der Internetverbindung das Programm `cinternet`, welches Sie im Kapitel 12 kennenlernen werden.

Die Linuxbu.ch/Tools können Sie relativ leicht um weitere Module erweitern. Eventuell finden sich ja Leser, die bereit sind, eigene Entwicklungen beizutragen.

3.6 Benutzerverwaltung in großen Netzen

Wenn Sie Linux-PCs im Netz betreiben, werden Sie nicht alle Administrationsaufgaben der Benutzerverwaltung auf allen Rechnern wiederholen wollen.

Das noch vor Jahren hierfür meist eingesetzte Network Information System (*NIS*), (Yellow Pages) entspricht seit langem nicht mehr den heutigen Sicherheitsanforderungen und ist weder hinreichend flexibel noch erweiterbar. Deshalb setzten sich hier hierarchische Datenbanken durch. Von der X.500 Protokollfamilie, welche einen umfangreichen Verzeichnisdienst definiert, stammt das Lightweight Directory Access Protocol (*LDAP*) ab. *Directory* bezeichnet im englischen Sprachgebrauch Verzeichnis. LDAP ist leseoptimiert. Daher eignet es sich besonders für Aufgaben wie das Authentifizieren von Benutzern und Adressbücher, bei denen Abfragen überwiegen.

LDAP ist nicht ursprünglich als Benutzerverwaltung entwickelt worden. Sie können darin weit mehr als die Daten und Passwörter Ihrer Benutzer ablegen. So könnten Sie beispielsweise Mitarbeitern ihre Fotos zuordnen, zusätzliche Telefonnummern speichern oder die URL ihrer Homepage hinterlegen. Sie sind hier nicht an Vorgaben gebunden, die Sie vielleicht bei NIS als einschränkend empfunden haben. LDAP ist kein Linux/Unix-Spezifikum. Microsoft verwendet seit Windows2000-Server seine eigene Version davon unter dem Namen Active Directory (*AD*). Dieses macht nichts anderes als eine LDAP-Datenbank: es verwaltet insbesondere Benutzer- und Maschinenkonten.

Damit Sie die Benutzer Ihrer Organisation mit LDAP verwalten können, benötigen Sie neben der Datenbank weitere Komponenten.

- LDAP selbst stellt lediglich die Funktionen zur Datenverwaltung bereit. Es speichert die Informationen und gibt sie bei Bedarf an Berechtigte heraus. LDAP läuft als Serverprozess auf einem der Linux-Server. Alternativ zu einem LinuxServer können Sie ein Active Directory eines Windows 2000/2003-Servers verwenden. In diesem Fall installieren Sie die Samba-Komponente `samba-winbind`, die mit PAM (s. u.) zusammenarbeitet.
- Weiterhin konfigurieren Sie den *Name Service Switch* (*NSS*). Dieser macht Ihre Benutzer auf Ihren Linux-PCs gegenüber dem System bekannt. Vernetzte PCs mit zentraler Benutzerverwaltung verfügen über keine oder über nur sehr wenige Benutzerdaten. Verschiedene Benutzerdatenbanken können Sie mit NSS miteinander verknüpfen und gemeinsam nutzen.
- Die *Pluggable Authentication Modules* (*PAM*) sind eine Erweiterung der zentralen C-Bibliothek Ihres Linux-Systems. Sie bewachen die Zugänge zur Maschine. Sie können Benutzer authentifizieren, das Neusetzen der Passwörter koordinieren und weitere Routinen für den Zugang zu PCs anbieten. So könnte beispielsweise eine PAM-Komponente ein Samba-Homeverzeichnis einbinden, da dieser Vorgang ebenfalls Benutzername und Passwort benötigt. So braucht ein PC seine Benutzer nicht zweimal nach diesen Daten zu fragen.
- Der *Name Service Caching Daemon* (*NSCD*) merkt sich für eine bestimmte Zeit Zuordnungen, beispielsweise zwischen numerischen User-IDs und den Namen von Accounts, damit Ihr Linux-PC nicht bei jedem Aufruf von `ls` den LDAP-Server fragen muss.

Die nächsten Abschnitte beschreiben, wie Sie diese Komponenten zu einer flexiblen, sicheren und leicht erweiterbaren Benutzerverwaltung zusammenführen.

3.6.1 Kurzeinführung in LDAP

LDAP ist eine hierarchische Datenbank. Anders als bei relationalen Datenbanken legt sie ihre Daten nicht in miteinander verknüpften Tabellen, sondern in einer Baumstruktur ab. LDAP eignet sich dadurch für sehr kompaktes Speichern von Benutzerinformationen jeder Art. Die Datenbasis lässt sich sogar so erweitern, dass LDAP Aufgaben für einen Samba-basierten Primary Domain Controller (PDC) übernehmen kann.

LDAP arbeitet objektorientiert. Jeder Directory-Eintrag beschreibt ein Objekt, welches eine Person, eine Verwaltungseinheit oder auch ein Server, ein Drucker usw. sein kann. Jeder Eintrag kann weitere Attribute besitzen, die einen Typ und einen bzw. mehrere Werte haben. Im hierarchischen Verzeichnis gibt es immer eine einzige Wurzel *root*, ähnlich wie beim UNIX/Linux-Verzeichnisbaum. Die Baumwurzel lässt sich hier wie dort weder verschieben noch im Betrieb verändern.

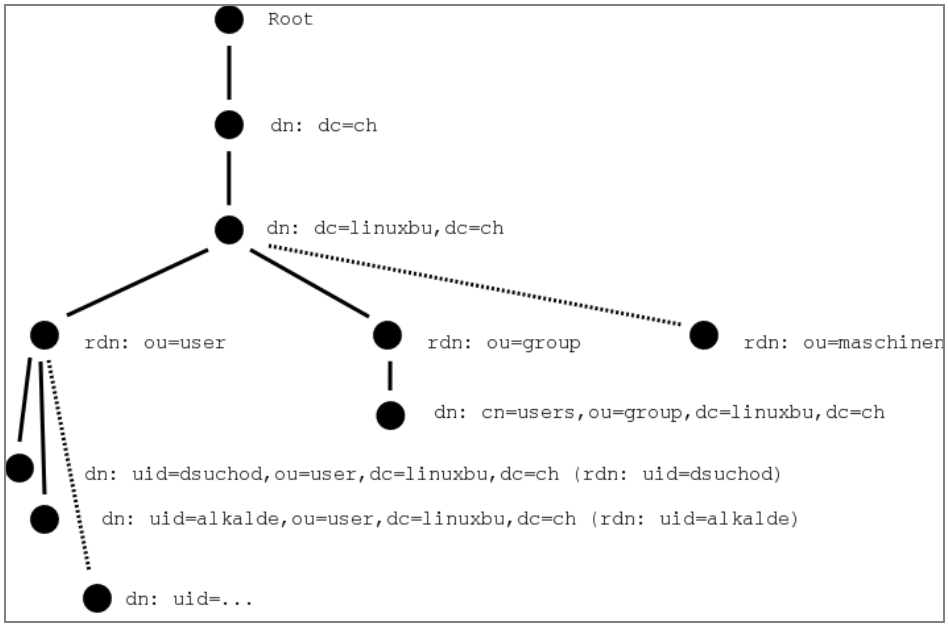

Abbildung 3.17: Eine LDAP Beispielhierarchie

Clients müssen später Datensätze eindeutig identifizieren. Deswegen besitzt jedes Objekt im Verzeichnis einen eindeutigen Namen, den *Distinguished Name* (*dn*), deutsch »ausgezeichneter Name«. Dieser setzt sich, von der Wurzel aus gelesen, aus den bisherigen Distinguished Names zusammen. Je tiefer Sie in die Baumhierarchie hinab-

steigen, desto länger wird er. Es gibt mehrere Möglichkeiten, um eindeutige Namen zu erreichen.

- *Domain Components* (dc) oder
- *Country* (c), *Organization* (o)

Microsoft beispielsweise setzt in seinen Active Directories Domain Components ein. Sie stellen auf eine sehr leicht verständliche Weise sicher, dass die Objekte auf jeder Hierarchieebene eindeutig sind. Die Wurzel bezeichnet man beginnend mit der Toplevel-Domain einer Site, also beispielsweise mit *de*. In den darunter liegenden Hierarchien folgen Second-Level-Domain-Namen, wie *mydomain* und, falls erforderlich, Sublevel-Domain-Namen, wie *entwicklung*. Ab dann verwenden sie meistens andere Bezeichner, wie *Organizational Unit* (*ou*). Sie können alternativ dazu das traditionelle Verfahren einsetzen, die Top-Level-Objekte *Country* und *Organization* zu verwenden.

Einträge zu einem Objekt heißen Attribute. Der *Common Name* (*cn*) ist ein allgemeiner Bezeichner, ein für Menschen gut les- und merkbares Attribut, ähnlich einem Rechnernamen. An den Baumenden ist dieses Attribut häufig Bestandteil des *dn*. In den folgenden Beispielen ist die User-ID *uid* Bestandteil des *dn*, da sie auf jeden Fall eindeutig ist.

Für viele Standarddaten sind bereits Klassen vordefiniert. Diese können voneinander Eigenschaften und definierte Attribute erben. So ist die üblicherweise für Personendaten verwendete Klasse `InetOrgPerson` von `OrganizationalPerson` und diese wieder von `Person` abgeleitet. Zu einer Person gehören zwingend als sogenannte Must-Attribute die Objektklasse `objectClass` selbst, der Nachname `sn` und der `commonName`, üblicherweise Vor- und Nachname. Zusätzlich gibt es mit MAY gekennzeichnet optionale Attribute, wie eine beliebige Beschreibung `description`, Verweise auf ein anderes Objekt: `seeAlso`, eine Telefonnummer `telephoneNumber` oder ein Passwort: `userPassword`. Da mit einer Person häufig noch weitere Eigenschaften verknüpft sind, gibt es die abgeleitete Objektklasse `organizationalPerson`. Diese erbt die Eigenschaften von `Person` und definiert darüber hinaus optionale Eigenschaften.

```
dn: uid=alkalde,ou=people,dc=mydomain,dc=site
cn: Anna Alkalde
gidNumber: 100

givenName: Anna
homeDirectory: /home/alkalde
loginShell: /bin/bash
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
sn: Alkalde
uid: alkalde
uidNumber: 1001
userPassword:: e2NyeXB0fUZBdzBSbGhXU3ZkRzY=
```

Das Beispiel zeigt einen typischen LDAP-Eintrag einer Person. Die Objektklasse ist `InetOrgPerson`. Sie hat von `Person` die zwingenden Attribute `sn` und `cn` geerbt und erlaubt weitere Attribute wie `employeeNumber`. Kombiniert ist `InetOrgPerson` mit `posixAccount` und `shadowAccount`. Diese Objektklassen liefern weitere Felder, wie `uidNumber` oder `homeDirectory`.

3.6.2 Benutzerverwaltung mit LDAP

Für eine einfache Benutzerverwaltung benötigt ein Linux-Sytem folgende Objekte:

- einen Common Name als Bezeichner des tatsächlichen Namens einer Person,
- eine eindeutige Zeichenfolge als UserID,
- eine eindeutige Benutzer- und Gruppennummer,
- ein Heimatverzeichnis,
- eine Login-Shell und
- eventuell ein Benutzerpasswort.

Soll die Datenbank die Einheitlichkeit der Adressbücher der Mitarbeiter sicherstellen, sollten Sie außerdem Daten wie Telefonnummer, E-Mail-Adresse, persönliche Webseite usw. speichern.

LDAP bietet Administratoren viel Freiheit beim Organisieren der Datensätze. Solange sie sich an die LDAP-Standards halten, können sie die Benutzerdaten in der Datenbank in sehr verschiedener Weise ablegen:

- Sie könnten auf einer Hierarchieebene verschiedene Unterbäume für einzelne Abteilungen anlegen und diesen Abteilungen ihre Benutzer zuzuordnen.
- Viele Administratoren ordnen alle Mitarbeiter in einem einzigen Baum an und vermerken in einem weiteren Attribut die Abteilung des Mitarbeiters. Dieses Modell erleichtert das Aktualisieren der Datenbank nach einem Wechsel der Abteilung.

Die Designentscheidung hat später Einfluss auf die Angabe des Suchfilters für die LDAP-Client-Konfiguration. Für den Anfang bieten die YaST2-Komponenten von OpenSuSE einen guten Einstieg. Machen Sie sich zunächst damit vertraut, bevor es danach tiefer in Details geht.

3.6.3 Aufsetzen eines OpenLDAP-Servers

Wenn in Ihrem Netz schon ein LDAP-Server arbeitet und Sie diesen benutzen möchten oder sollen, überspringen Sie bitte diesen Abschnitt.

OpenLDAP2 ist eine freie Implementierung der Version 3 des LDAP-Standards. Bei OpenSuSE 11 können Sie den Server mit Tools und Hilfsprogrammen in der YaST-Paketauswahl durch die Suche von `ldap` finden.

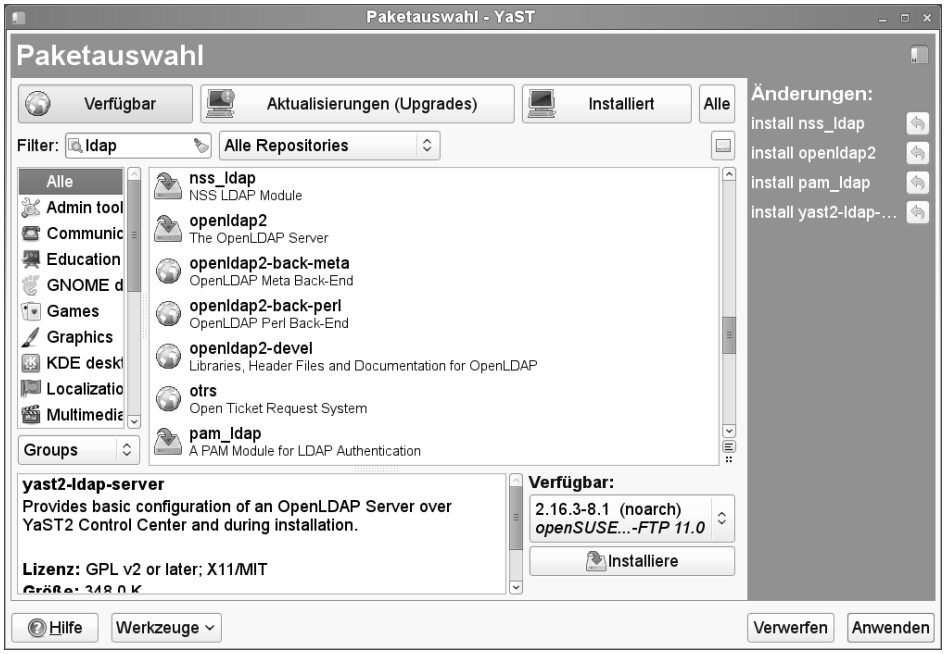

Abbildung 3.18: Installation der benötigten LDAP-Komponenten zur Benutzerverwaltung

Für einen Linux-LDAP-Server zur Benutzerverwaltung benötigen Sie die folgenden Pakete:

- `openldap2` – den OpenLDAP2-Server,
- `nss_ldap` – LDAPv3-Modul des NSS für LDAP-Benutzer-Identifiation
- `pam_ldap` – LDAPv3-Modul für PAM für LDAP-Benutzer-Authentifizierung
- `yast2-ldap-server` – zur LDAP-Konfiguration via YaST-Modul. Nach der Installation zeigt Ihnen das YaST-Menü den Eintrag *Netzwerkdienste • LDAP-Server.*

Die Installation legt einige Verzeichnisse und Konfigurationsdateien für OpenLDAP an. Der LDAP-Server erwartet seine Konfigurationdatei `slapd.conf` unterhalb von `/etc/openldap`. Die Dateien der laufenden Datenbank landen üblicherweise im Verzeichnis `/var/lib/ldap`. Dieses Verzeichnis können Sie in der Konfigurationsdatei wie voreingestellt verwenden oder anders angegeben.

Zusammen mit dem LDAP-Paket erhalten Sie etliche Kommandozeilenprogramme. Die Werkzeuge `ldapsearch`, `ldapadd`, `ldapdelete` und `ldapmodify` für Operationen auf der LDAP-Datenbank stehen im Verzeichnis `/usr/bin`.

Bevor Sie mit ihren frisch installierten Server jetzt Daten erfassen, richten Sie ihn durch Erstellen einer Konfiguration in YaST via *Netzwerkdienste • LDAP-Server* ein.

Abbildung 3.19:
Einrichtung des LDAP-Servers durch Klicken auf *Konfigurieren*

Im nächsten Konfigurationsdialog legen Sie unterhalb des Punktes *Datenbanken* eine neue Datenbank an.

Abbildung 3.20:
Konfiguration des LDAP-Servers

Dieser Klick eröffnet ein neues Dialogfeld *Konfiguration des LDAP-Servers*, in dem Sie die grundsätzlichen Einstellungen Ihrer LDAP-Benutzerverwaltung eintragen wie die Bezeichnung des Basis-DN und das Passworts für den LDAP-Administrator. Die *Globalen Einstellungen* können Sie mit den Vorgabewerten belassen. Unterhalb von *Datenbanken* aktivieren Sie *Datenbank hinzufügen.*

Abbildung 3.21: Legen Sie hier die *Allgemeinen Einstellungen* fest

Die Festlegung der Passwortrichtlinien klappt nicht immer gleich, passen Sie sie am besten später an. Wenn Sie später z. B. die Passwortrichtlinien ändern wollen, können Sie das im vorherigen Dialog tun, der nun einen Eintrag Ihrer gerade erstellten Datenbank enthält.

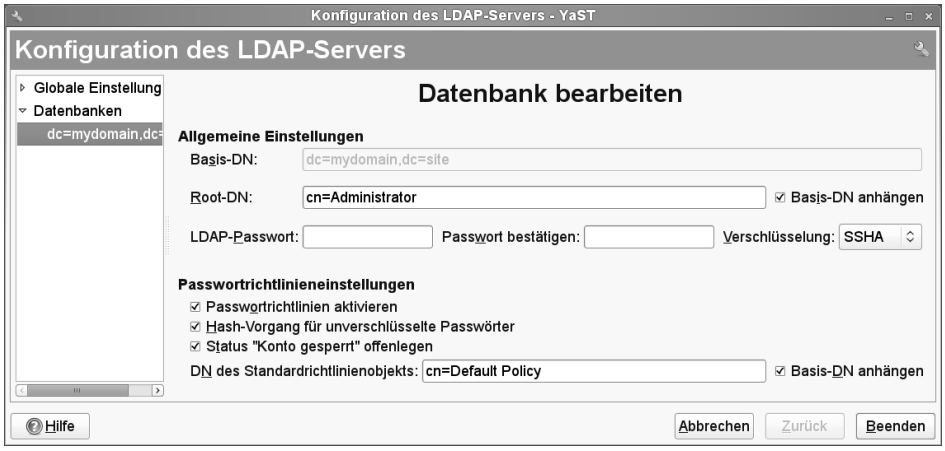

Abbildung 3.22: Aktivieren der Passwortrichtlinien.

Beim Klicken auf *Beenden* fragt das Dialogfenster. ob Sie das Richtlinienobjekt anlegen wollen. Dieses bestätigen Sie und nehmen dann ihre gewünschten Einstellungen vor.

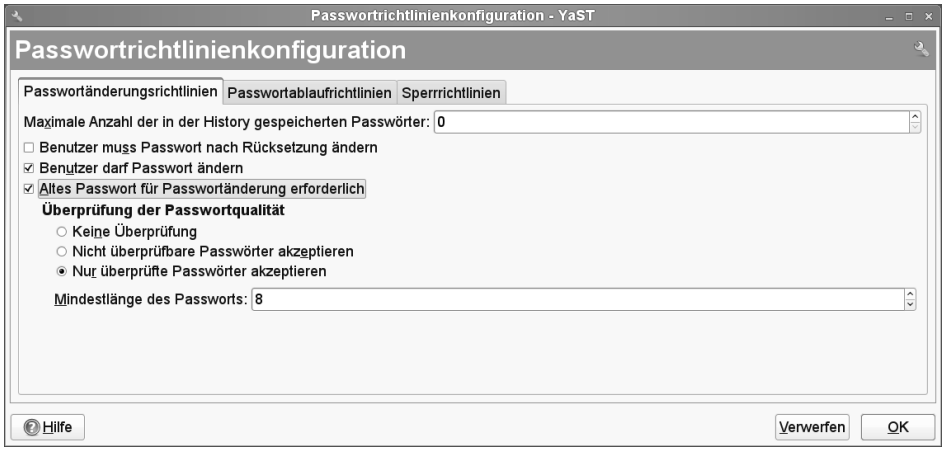

Abbildung 3.23: Standards für Passworte festlegen

Nach Abschluss der Konfiguration mit *Beenden* startet YaST den LDAP-Server neu. Wenn Sie das Startverhalten des Servers ändern wollen, können Sie dieses in YaST mittels *System* • *Runlevel-Editor* einstellen.

Abbildung 3.24: Bearbeiten der Runlevel

Hinweis: Das Benutzerinterface der LDAP-Server ist nicht ganz ausgereift. Wenn Sie eine Datenbank löschen wollen, sollten Sie dieses per Hand tun: Löschen Sie das gleichnamige Verzeichnis in `/var/lib/ldap` und entfernen Sie die Einträge aus `/etc/openldap/slapd.conf`.

Nach dem Start des Dienstes erfolgt ein kleiner Test, ob der Server antwortet. Er müsste sich melden, kann aber noch keine Informationen ausgeben, da Sie noch keine Benutzerdaten eingetragen haben. Mit dem folgenden Befehl an der Kommandozeile prüfen Sie, ob der Directory-Administrator sich erfolgreich nach Eingabe seines Passwortes mit der Datenbank verbinden darf.

```
ldapsearch -x -D "cn=Administrator,dc=mydomain,dc=site" -W
# extended LDIF
#
# LDAPv3
# base <> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# search result

search: 2
result: 32 No such object
# numResponses: 1
```

Wenn Sie diese Ausgabe sehen, hat alles geklappt. Die Schalter für die Kommandozeile gelten für die meisten OpenLDAP-Tools. Sie haben folgende Bedeutung:

- `x - simple Authentication`, sollte eigentlich immer angegeben werden.
- `D` – Distinguished Name des Datenbankadministrators oder eines anderen Benutzers, unter dessen ID Sie auf die Datenbank zugreifen wollen.
- `W` – fragt interaktiv nach einem Passwort, meistens zusammen mit `-D`. Mit *w* können Sie das Passwort direkt angeben, z. B.: `-w Geheim`.
- `f` – lesen aus einer Datei mit anschließender Angabe des Dateinamens. Spielt bei `ldapadd` oder `ldapmodify` eine Rolle.
- `h` – gibt den Rechnernamen oder die IP des LDAP-Servers an, z. B. `-h 127.0.0.1`.
- `H` – gibt den Unified Resource Indicator (*URI*) für eine LDAP-Quelle an. Für eine verschlüsselte Verbindung auf den Server ldap.mydomain.site sieht die Angabe so aus: `-H ldaps://ldap.mydomain.site:636`. Den Port müssen Sie nicht angeben, wenn Sie den Standardport für verschlüsselte Verbindungen `636` wählen. Der Standardport für unverschlüsselte Verbindungen lautet `389`. Diesen Parameter benötigen Sie nur, wenn Sie oder YaST2 die LDAP-Quelle nicht in der `/etc/openldap/ldap.conf` konfiguriert haben.
- b – setzt den obersten Knoten für die Suche auf dem LDAP-Baum. So können Sie die Suche einschränken und dadurch beschleunigen. Suchen Sie beispielsweise nur Benutzer, ist die folgende Angabe sinnvoll: `-b ou=user,dc=mydomain,dc=site`. Sie müssen ebenfalls einen Startknoten für allgemeines Suchen angeben, wenn

dieser nicht in der `ldap.conf` definiert ist, da Ihr Suchergebnis sonst leer bleiben könnte.

- `d` – legt den Debug-Level fest. Das ist hilfreich, um je nach Level ausführlichere Ausgaben zu erhalten, wenn Sie auf Fehlersuche gehen.

3.6.4 Benutzer einfügen

Nun wird es Zeit, Daten zu erfassen. Hierzu stehen Ihnen viele Wege offen. Ein Weg führt über die YaST2-LDAP-Client-Einstellung: *Netzwerkdienste* • *LDAP-Client* aus.

Abbildung 3.25: Das erste Konfigurationsfenster der Client-Einstellung

Als erstes aktivieren Sie im Bereich Benutzerauthentifizierung *LDAP verwenden*. Dann tragen Sie Ihren LDAP-Base-DN ein: `dc=mydomain,dc=site` und schalten die verschlüsselte Übertragung zum Server aus, solange Sie nur lokal arbeiten. Die Adressen des LDAP-Servers können Sie für die lokale Verwendung auf der Maschine selbst vorerst so belassen. Der neu konfigurierte Server startet damit vorerst nur auf dem lokalen Interface `127.0.0.1`, womit Sie Angriffe in der Setup- und Testphase vermeiden. Dann geht es über *Erweiterte Konfiguration* zum nächsten Dialog.

Abbildung 3.26: Client- und Verwaltungseinstellungen

Die meisten Felder hat YaST2 bereits automatisch mit Einträgen gefüllt. Die Client-Einstellungen können Sie so belassen. In den Verwaltungseinstellungen legen Sie noch den Administrator-DN festlegen, so wie Sie ihn bei der Serverkonfiguration festgelegt haben. Die hier eingetragenen Daten finden Sie nach dem Speichern in `/etc/sysconfig/ldap` wieder. Sie können die Standardkonfigurationsobjekte durch Anklicken von *OK* nun automatisch erzeugen lassen. Anschließend geht es weiter mit der Schaltfläche *Einstellungen für die Benutzerverwaltung konfigurieren* .... Beim Zugriff auf diese Maske fragt YaST2 nach dem Kennwort des Directory-Administrators. Die Authentifizierung benötigt es, um Daten in der Datenbank einzutragen.

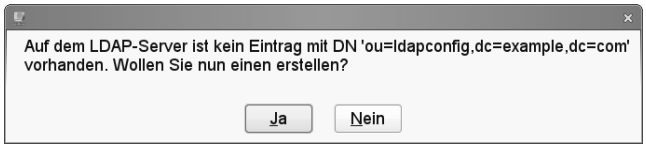

Abbildung 3.27: Diese Frage bestätigen Sie mit *OK*

Hinweis: Schlägt eine Eintragung im LDAP-Server mit dem Hinweis auf unzureichende Rechte fehl, starten Sie die LDAP-Client-Einstellung neu und authentifizieren Sie sich erneut als LDAP-Administrator.

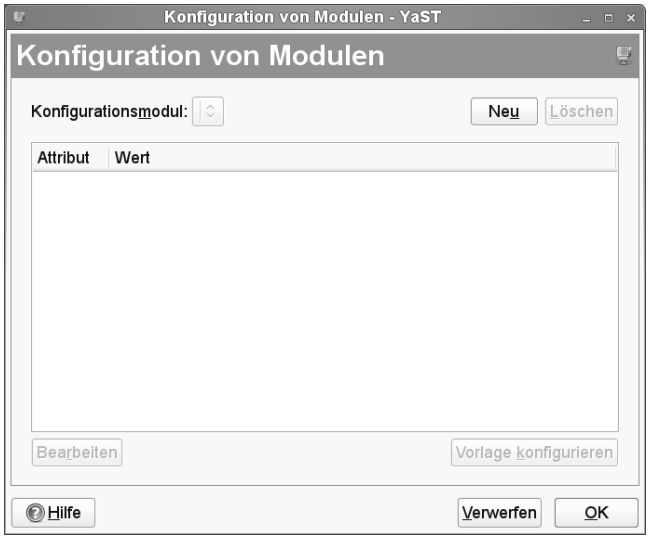

Abbildung 3.28: *Anlegen und Konfiguration von Modulen*

Das Fenster *Konfiguration von Modulen* legt Konfigurationsgruppen an. Um einen neuen Abschnitt des LDAP-Baums zu erzeugen, in dem LDAP Gruppen ablegen soll, hier `group`, klicken Sie auf *Neu*.

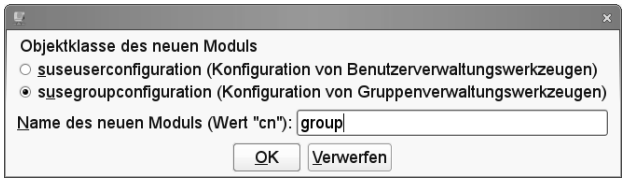

Abbildung 3.29: Anlegen von »group«

Anschließen legen Sie mit *Neu* den Knoten für die Benutzerverwaltungswerkzeuge an, hier `user` genannt. In beiden Fällen können Sie durch *Vorlage konfigurieren* weitere Details einstellen.

Schließen Sie dann mit *OK* und *Beenden* die LDAP-Client-Konfiguration. Die Datenbank enthält nun schon eine mehrere Einträge. Diese zeigt Ihnen `ldapsearch -x`. Jedoch kennt ihre Datenbank immer noch keine Benutzer. Diese fügen Sie in YaST2 in *Sicherheit und Benutzer* mit dem Unterpunkt *Benutzer- und Gruppenkonfiguration* hinzu.

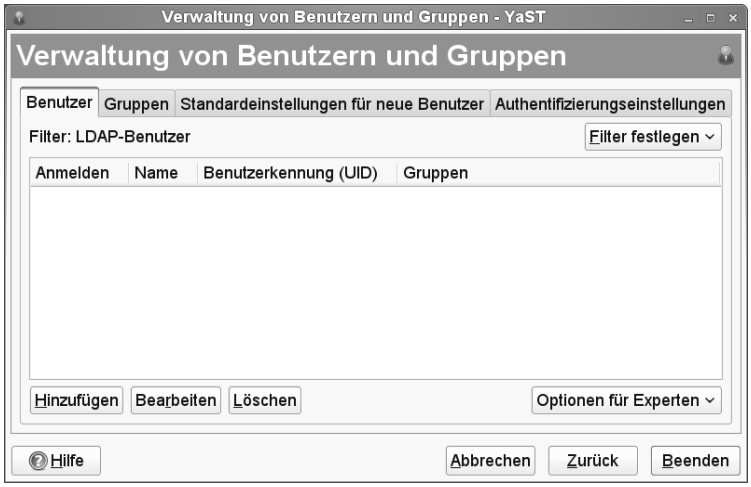

Abbildung 3.30 LDAP-Datenbank mit Benutzerdaten füllen

In der Tabelle der Benutzer sehen Sie zuerst nur alle bereits angelegten lokalen Accounts, da im LDAP selbst noch keine Accounts eingetragen sind. Über *Filter festlegen • LDAP-Benutzer* erhalten Sie eine neue Sicht der Tabelle. Das Dialogfenster fragt Sie nach dem Kennwort des LDAP-Administrators.

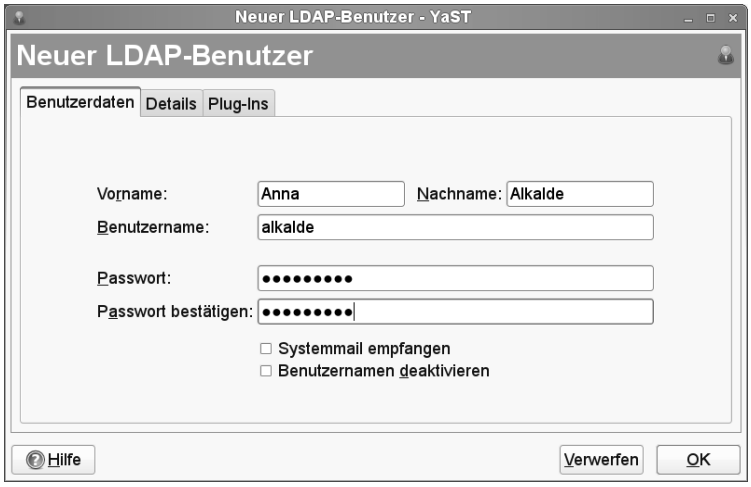

Abbildung 3.31: neue Benutzerin im LDAP hinzufügen

So können Sie Ihre Benutzer im LDAP nacheinander eintragen. Einstellungen unter Details oder Plug-Ins können Sie üblicherweise beibehalten.

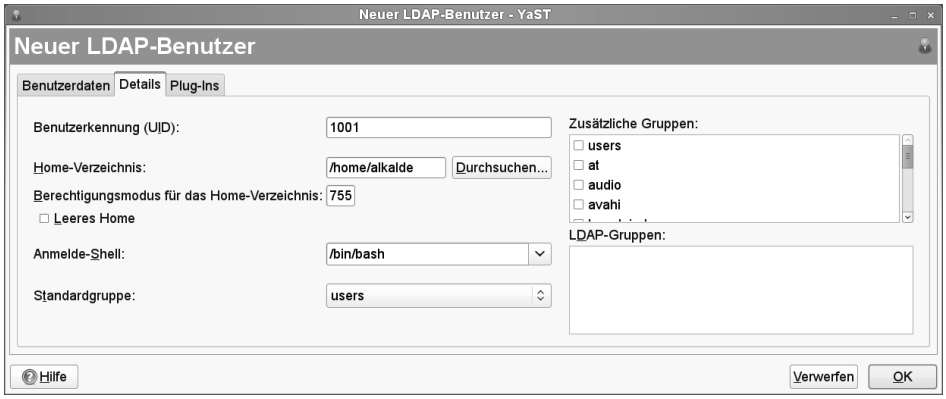

Abbildung 3.32: Account-Details anpassen

Wenn Sie nun über *Filter festlegen* • *Benutzerdefinierte Filtereinstellung* ... die LDAP-Benutzer aktivieren, listet eine Tabelle sowohl lokale als auch LDAP-Accounts in auf.

Mit *Beenden* aktivieren Sie Ihre Einstellungen. Mit dem Befehl

```
linux:~ # id alkalde
uid=1001(alkalde) gid=100(users) Gruppen=100(users)
```

können Sie nachsehen, ob Ihre neu angelegte Benutzerin dem System bekannt ist. Nun sollte sich diese Nutzerin auch an einer Konsole oder mit `ssh` an dem PC anmelden können. Das Home-Verzeichnis ist dank Ihrer Einstellungen in der Benutzerverwaltung durch das YaST-Modul *Benutzer und Gruppenkonfiguration* bereits automatisch erstellt worden.

3.6.5 Mit OpenLDAP direkt arbeiten

Bisher hatten Sie mit den Komponenten LDAP, NSS und PAM nicht direkt zu tun. Die Einstellungen erledigten die YaST2-Module. Wenn Sie viele Benutzer gleichzeitig anlegen wollen, möchten Sie das vielleicht nicht interaktiv machen. Der Eintrag unserer soeben angelegten Benutzerin sieht wie folgt aus:

```
linux:~ # ldapsearch -x -D "cn=Administrator,dc=mydomain,dc=site" -W
uid=alkalde
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=mydomain,dc=site> (default) with scope subtree
# filter: uid=alkalde
# requesting: ALL
#

# alkalde, people, mydomain.site
```

```
dn: uid=alkalde,ou=people,dc=mydomain,dc=site
cn: Anna Alkalde
gidNumber: 100
givenName: Anna
homeDirectory: /home/alkalde
loginShell: /bin/bash
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
sn: Alkalde
uid: alkalde
uidNumber: 1001
userPassword:: e2NyeXB0fUZBdzBSbGhXU3ZkRzY=

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Wollen Sie weitere Benutzer nicht-interaktiv im LDAP anlegen, können Sie auf Basis des oben gezeigten LDIFs eine Datei erzeugen:

```
# LDIF-Datei für ein neuen Beispielnutzer: debacher
dn: uid=debacher,ou=people,dc=mydomain,dc=site

cn: Uwe Debacher

gidNumber: 100

givenName: Uwe

homeDirectory: /home/debacher

loginShell: /bin/bash

objectClass: top

objectClass: posixAccount

objectClass: inetOrgPerson

sn: Debacher

uid: debacher

uidNumber: 1002

userPassword: TestPW
```

Die Beispieleinträge wurden einfach von der mit YaST erzeugten Vorlage abgeleitet. Das Passwort wird automatisch in die kodierte Form beim Eintrag übersetzt. Da das Home-Verzeichnis nicht automatisch generiert wird, müssen Sie dies manuell vornehmen. Das Kommando `ldapadd` zum Einfügen von Objekten in die Datenbank wie im folgenden Listing kennen Sie schon aus dem vorherigen Abschnitt.

```
linux:~# ldapadd -c -x -D "cn=Administrator,dc=mydomain,dc=site" -W -
f ldif

Enter LDAP Password:

adding new entry "uid=debacher,ou=people,dc=mydomain,dc=site"
```

Der Kommandozeilenschalter `-c` sorgt dafür, dass das Kommando nicht abbricht, wenn schon Daten eingetragen sind. So können Sie Ihr LDIF einfach erweitern und erneut laden, um einen oder mehrere weitere Accounts hinzuzufügen. Hier im Beispiel legen Sie einfach nur einen Unterknoten für den Benutzer (`debacher`) an. Für weitere Benutzer nehmen Sie den Eintrag als Vorlage.

Sind Sie mit einem Eintrag nicht einverstanden, können Sie diesen mit `ldapdelete` wieder entfernen:

```
ldapdelete -x -D "cn=Administrator,dc=mydomain,dc=site" -W
"uid=debacher,ou=people,dc=mydomain,dc=site"
```

Sie können jedoch keine Knoten entfernen, in denen noch Einträge vorhanden sind. Der Versuch, den Baum `ou=people,dc=mydomain,dc=site` zu löschen, schlägt fehl, solange darunter noch ein Benutzer (`alkalde`) in der Datenbank steht. Um den Inhalt einer kompletten Datenbank zu löschen, können Sie den Dienst stoppen und dann alle Dateien im Verzeichnis `/var/lib/ldap` entfernen.

```
rclinux:~/ldap # ldap stop
rclinux:~/ldap # rm /var/lib/ldap/*
rclinux:~/ldap # rcldap start
```

In der Standardeinstellung startet der LDAP-Server mit dem oben angegebenen Befehl nur auf der IP `127.0.0.1`. In dieser Einstellung arbeitet OpenLDAP nur unverschlüsselt. Möchten Sie den Server über das Netzwerk von entfernten Rechnern aus ansprechen, öffnen Sie erneut YaST *Netzwerkdienste* • *LDAP-Server*. Öffnen Sie den Firewall-Port und aktivieren Sie TLS. Dann sollten Sie auf entfernten Rechnern im YaST *Netzwerkdienste* • *LDAP-Client* öffnen und dort die IP Ihres LDAP-Servers eintragen. Ein Eintrag von `192.168.75.128:389 127.0.0.1:389` bewirkt, dass der Server-Prozess `slapd` auf den Interfaces mit den IP-Nummern `192.168.75.128` und `127.0.0.1` auf ankommende Verbindungen lauscht.

Fehler beim Versuch, Daten in LDAP einzufügen, können verschiedenste Ursachen haben. Diese sind manchmal nicht auf den ersten Blick sichtbar. Die folgenden Tipps helfen Ihnen hoffentlich bei der Fehlersuche:

- In der aktuellen Version des OpenLDAPs muss das Attribut, welches zum Aufbau des Distinguished Name verwendet wird, noch einmal in der Attributliste auftauchen.
- Weiterhin ist es erforderlich, immer die Objektklasse `top` anzugeben. Dieses ist eine generelle Klasse, die keine eigenen Must-Attribute definiert. Must-Attribute sind kumulativ: Mindestens eine Klasse ist neben `top` erforderlich, Werden mehrere Objektklassen angegeben, müssen alle Musts dieser Objektklassen gemeinsam erfüllt sein.
- Man kann nur leere Knoten löschen. Entfernen Sie alle Einträge nacheinander, ausgehend von denen, die am weitesten von der Wurzel weg sind zu dieser hin. Den Inhalt der gesamten Datenbank löschen Sie einfacher direkt wie oben gezeigt.

Zwei Dateien spielen für LDAP-Clients eine Rolle. Sie können das Verhalten eines LDAP-Clients mit den DNS-Einstellungen Ihres Linux-Systems vergleichen. Anders als bei NIS starten Sie für den LDAP-Client keinen eigenen Dienst. Die LDAP-Programme können die Daten zur Verbindung auf den Server aus der Datei `/etc/openldap/ldap.conf` lesen:

```
#BASE   dc=example, dc=com
#URI    ldap://ldap-master.example.com:666

#SIZELIMIT      12
#TIMELIMIT      15
#DEREF          never
TLS_REQCERT     allow
host    127.0.0.1
base    dc=mydomain,dc=site
```

Einige Parameter können Sie direkt auf der Kommandozeile angeben. Lesen Sie hierzu die Liste der Kommandozeilenschalter im Abschnitt zum OpenLDAP-Server.

Die beiden Erweiterungen der C-Bibliothek für die Benutzeranmeldung und -zuordnung PAM und NSS greifen ihrerseits auf die Datei `/etc/ldap.conf` zu. Diese Datei ist zur Erleichterung Ihrer Anpassungen ausführlich kommentiert.

Abbildung 3.33: Konfiguration eines entfernten LDAP-Servers

3.6.6 PAM im Einsatz

Im vorletzten Abschnitt hatten Sie die YaST2-Unterstützung zur Verwendung von LDAP zur Benutzerverwaltung kennen gelernt. Nun möchten Sie vielleicht mehrere Linux-Maschinen mit LDAP-Authentifizierung einrichten und nicht jedes Mal die grafische Oberfläche verwenden. Sie möchten ebenfalls verstehen, was sich hinter den Kulissen abspielt.

PAM alleine genügt nicht zum Einbinden von LDAP-Benutzern. Damit Ihr Linux-Rechner mit diesen Nutzern umgehen kann, nachdem sich diese angemeldet haben, wurde auch der *Name Service Switch* durch YaST entsprechend eingerichtet. Hierzu sehen Sie sich die Datei `/etc/nsswitch.conf` an. Diese definiert die Quellen, aus denen die zentrale C-Bibliothek – beispielsweise aus einer im Dateisystem gespeicherten numerischen User-ID – den Namen des Accounts ermittelt:

```
# /etc/nsswitch.conf
[ ... ]
passwd: compat
group:  files ldap nis
[ ... ]
```

In diesem Beispiel befragt NSS zuerst die lokalen Dateien `/etc/passwd` oder `/etc/group` nach Benutzer- oder Gruppenzuordnungen. Anschließend nutzt sie hierzu LDAP. Zur Demonstration, dass Sie weitere Quellen angeben können, sehen Sie hier noch einen Eintrag für das traditionelle *Network Information System* (*NIS*). Damit greift ihr Name Service Switch nach LDAP noch auf NIS zu. So können Sie schrittweise von NIS zu LDAP migrieren. Der `nscd` (Name Service Caching Daemon) speichert Anfragen an verschiedene Datenquellen zwischen (Caching). Der `nscd` merkt sich auch

unbekannte Nutzer im sogenannten Negativ-Cache. Diese schlägt er dann nicht mehr nach, sondern meldet sofort ihre Nichtexistenz. Das kann ein kurzfristiges Problem erzeugen. Wenn `nscd` diesen Eintrag noch nicht vergessen hat, ist ein Benutzer nach dem Hinzufügen einer neuen Datenquelle nicht sofort dem System bekannt.

Ein Aufruf von

```
rcnscd restart
```

löscht den Cache und ermöglicht damit den sofortigen Zugriff auf den Benutzeraccount.

LDAP ist nur eines der Beispiele einer netzwerkbasierten gemeinsamen Benutzerverwaltung. Während früher NIS eingesetzt wurde, wird zukünftig Kerberos diese Rolle übernehmen, da es beispielsweise bei Microsofts ADS bereits integriert ist. Damit die zentrale C-Bibliothek eines Systems und Applikationen nicht bei jeder Änderung angepasst werden müssen, wählt PAM einen völlig neuen Ansatz. Die einzelnen Anwendungen benutzen zur Benutzeranmeldung Funktionen aus einer Bibliothek, die PAM bereitstellt. Die Funktionen benutzen ihrerseits passende PAM-Module zum Authentifizieren.

PAM kann jedoch noch mehr. Neben dem *Authentication Management* kann es sich ebenfalls um die Aufgaben *Account Management*, *Session Management* und *Password Management* kümmern. Jedes PAM-Modul muss mindestens einen, kann aber auch mehrere dieser Jobs abdecken.

Diese Aufteilung findet sich in den Konfigurationsdateien wieder: Für jede Aufgabe finden Sie keinen, einen oder mehrere Einträge.

Für einige Anwendungen ist es ein Problem, dass PAM rein passiv ist und stets von einer Applikation aufgerufen werden muss. Wenn Sie automatisch nach dem Durchziehen Ihrer Chipkarte oder dem Auflegen Ihres Fingers auf einem biometrischen Leser eingeloggt werden wollen, muss eine Applikation regelmäßig pollen und die Authentifizierung anstoßen.

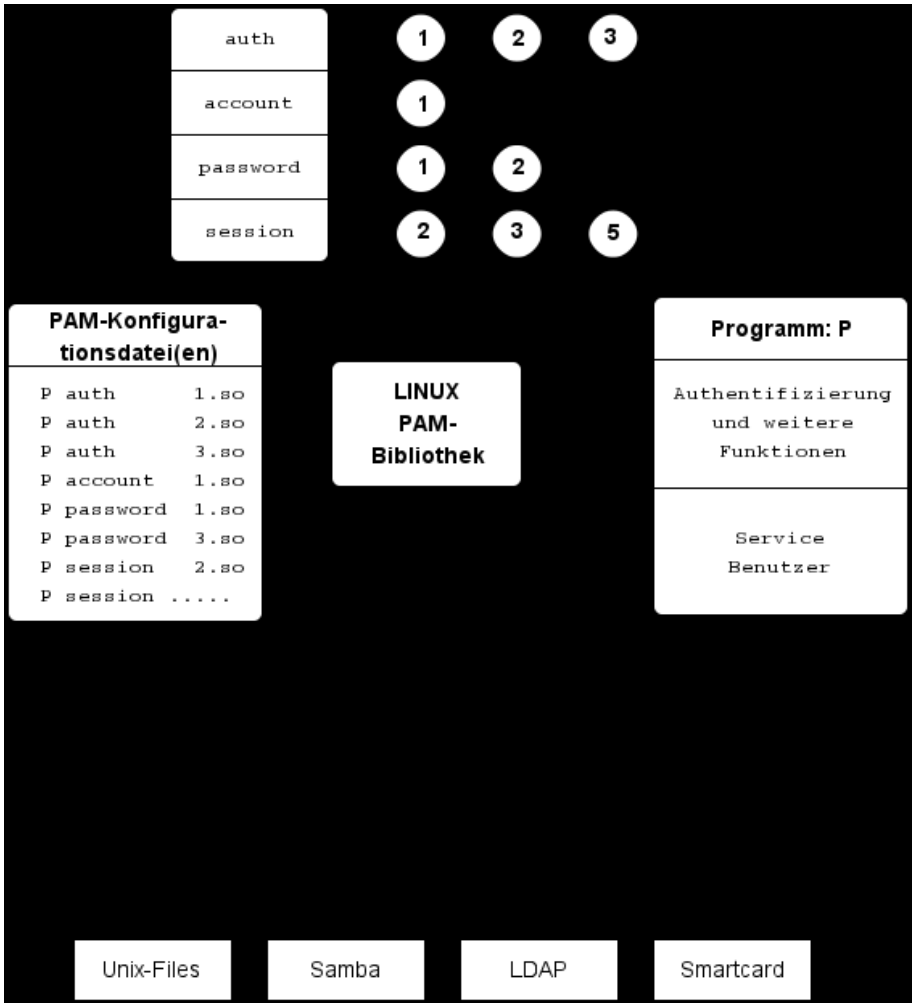

Abbildung 3.34: Die Funktionsweise von PAM

Die schematische Darstellung zeigt den Kommunikationsfluss zwischen den Applikationen und PAM. Die PAM-Bibliothek wird zur Laufzeit eines Programmes mit Benutzeranmeldung geladen. Das PAM-Modul kommuniziert zum einen mit der PAM-Bibliothek, um seine Parameter auszulesen. Zum anderen tauscht es Daten mit der Applikation aus, um an die Benutzerdaten wie Account-Name und Passwort zu gelangen.

Die PAM-Bibliothek ist bei SUSE ein eigenes RPM (`pam`, `pam-config` und `pam-modules`). Spezielle Module, die weitere Bibliotheken benötigen, wie das LDAP-Modul finden sich in separaten Paketen, die Sie bei Bedarf zusätzlich installieren (`pam_ldap`).

Bei der Installation legt PAM drei Verzeichnisse an:

- `/lib/security`
- `/etc/security`
- `/etc/pam.d`

`/lib/security` nimmt die Bibliotheksdateien auf. Dieses Verzeichnis enthält alle aktuell auf einem System installierten Standard-PAM-Module:

```
pam_access.so           pam_limits.so     pam_stress.so
pam_ck_connector.so     pam_listfile.so   pam_succeed_if.so
pam_cracklib.so         pam_localuser.so  pam_tally.so
pam_debug.so            pam_loginuid.so   pam_thinkfinger.so
pam_deny.so             pam_mail.so       pam_time.so
pam_echo.so             pam_make.so       pam_tty_audit.so
pam_env.so              pam_mkhomedir.so  pam_umask.so
pam_exec.so             pam_motd.so       pam_unix2.so
pam_faildelay.so        pam_namespace.so  pam_unix_acct.so
pam_filter              pam_nologin.so    pam_unix_auth.so
pam_filter.so           pam_permit.so     pam_unix_passwd.so
pam_ftp.so              pam_pwcheck.so    pam_unix_session.so
pam_gnome_keyring.so    pam_resmgr.so     pam_unix.so
pam_group.so            pam_rhosts.so     pam_userdb.so
pam_homecheck.so        pam_rootok.so     pam_warn.so
pam_issue.so            pam_rpasswd.so    pam_wheel.so
pam_keyinit.so          pam_securetty.so  pam_winbind.so
pam_lastlog.so          pam_shells.so     pam_xauth.so
pam_ldap.so             pam_smbpass.so
```

Einige Module verfügen über eigene Konfigurationsdateien, die ihr generelles Verhalten unabhängig von der aufrufenden Applikation steuern. Diese finden Sie im Verzeichnis `/etc/security`. Für die Konfiguration der PAM-Bibliotheken zu den einzelnen Diensten gibt es eine eigene Konfigurationsdatei unterhalb von `/etc/pam.d`.

```
atd                                 gnome-screensaver
chage                               gnome-screensaver-smartcard
chfn                                gnomesu-pam
chsh                                login
common-account                      login.old
common-account.pam-config-backup    other
common-account-pc                   passwd
common-auth                         polkit
common-auth.pam-config-backup       ppp
common-auth-pc                      rpasswd
common-password                     samba
common-password.pam-config-backup   shadow
common-password-pc                  smtp
common-session                      sshd
```

```
common-session.pam-config-backup   su
common-session-pc                  sudo
crond                              su-l
cups                               useradd
gdm                                xdm
gdm-autologin                      xdm-np
gnome-passwd                       xscreensaver
```

Die Menge der Dateien in diesem Verzeichnis leitet sich aus der Zahl der PAM-fähigen Dienste ab, die Sie auf dem PC installiert haben. Die meisten Dienste nutzen gemeinsame Konfigurationsdateien wie `common-auth`, damit Sie nicht alle Einstellungen für jeden Dienst wiederholen müssen. Trotzdem unterscheiden sich die Einstellungen: So gewinnen Sie nichts, wenn das System bei einem entfernten SSH-Zugriff, wie beim Konsolenlogin, auf ein sicheres TTY überprüft. Ebenso verbieten Sie *root*-Logins besser in der SSH-Konfiguration direkt als mit PAM.

Beim Start legt jedes Programm in einer meist gleichnamigen Konfigurationsdatei seine Parameter fest, z. B. `xdm`. Leider stimmt das nicht immer, so dass es Netzwerkdienste gibt, bei denen das abschließende `d` fehlt, wie bei `ppp` oder die abweichend bezeichnet sind, wie `samba`. Falls PAM keine passende Konfigurationsdatei findet, verwendet es die Datei `/etc/pam.d/other`. Wenn man nicht genau weiß, welcher Dienst diese Datei benutzt, sollte man sie wie im folgenden Beispiel möglichst restriktiv einstellen.

Generell sind alle Konfigurationsdateien einheitlich strukturiert.

```
#modultyp   modulsteuerung  modulpfad      argumente
auth        sufficient      pam_unix2.so   nullok
auth        required        pam_ldap.so    use_first_pass
account     required        pam_unix2.so
password    required        pam_unix2.so
session     required        pam_unix2.so
session     required        pam_env.so
session     required        pam_devperm.so
```

Der *Modultyp* legt die Management-Funktion eines Eintrags fest. Es gibt vier Modultypen:

- `auth` – Module in dieser Kategorie dienen der Benutzeridentifizierung durch klassische Abfragen von Benutzername/Passwort, durch biometrische Verfahren, Smartcard mit PIN oder Ähnliches. Voraussetzung sind Schnittstellen der angeschlossenen Geräte, wie Smartcard-Leser. In diesen Bereich fallen auch spezielle Module, die Benutzerkennung und Passwort abgreifen, um sie für einen authentifizierten Mount-Prozess des Home-Verzeichnisses z. B. von einem Samba-Server einzusetzen oder damit ein Kerberos-Token für NFSv4 zu holen.
- `account` – Diese Module verwalten den Zugriff auf Accounts nach der Anmeldeprozedur, um den Zugriff auf einen Dienst abhängig von der Uhrzeit oder dem Wochentag zu steuern.

- `password` – Diese Module steuern das Aktualisieren von Passworten- oder Tokens. Verwendet das Kommando `passwd` die PAM-Bibliotheken, lässt sich festlegen, welche Passwörter das Programm ändert und akzeptiert. Gleichzeitig können Sie dadurch Passworte netzwerktransparent ändern.
- `session` – Module dieses Typs verwalten Einstellungen für die Sitzung des Benutzers. Hiermit kann man die im System verbrachte Zeit abrechnen. Ebenso fällt in diese Kategorie das Setzen von Limits oder Zugriffsberechtigungen auf Unix-Devices sowie das Mounten von Verzeichnissen. Benötigen Sie für letzteres ein Passwort, sollte dieser Vorgang im `auth`-Modul erfolgen. Das wäre bei Heimatverzeichnissen von einem Samba-Server der Fall.

In der Spalte `modulsteuerung` legen Sie fest, wie PAM auf Erfolgs- oder Fehlermeldungen der einzelnen Module reagiert. Die Standardbedingungen können Sie bei Bedarf verfeinert aufgliedern, indem Sie PAM-Module aufeinander stapeln. Diese Stacking genannten Stapel arbeitet PAM in Reihenfolge ihrer Auflistung ab. Unter bestimmten Bedingungen können weiter unten stehende Module nicht erreicht werden. Das aufrufende Programm erfährt von diesem Vorgang der Abarbeitung das zusammengefasste Endergebnis als Statusbericht über Erfolg (*success)* oder Misserfolg (*fail)*.

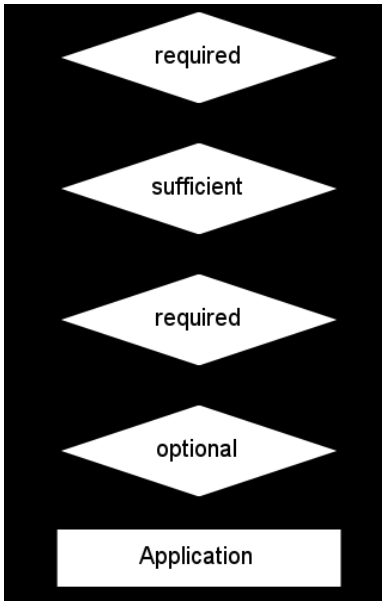

Abbildung 3.35: Die Funktion der Modulsteuerung

Für den Gesamterfolg eines PAM-Stapels kommt es auf die Einzelergebnisse an. Diese können mit unterschiedlichen Auswirkungen in die Gesamtwertung eingehen.

- `required` – Dieses Modul muss mit dem Status *success* beendet werden, damit das zusammengefasste Ergebnis aller Module dieses Typs erfolgreich sein kann. Ein

Misserfolg dieses Moduls zeigt sich erst, nachdem alle Module dieser Kategorie durchlaufen wurden.

- `requisite` – verhält sich ähnlich wie *required*, gibt jedoch die Kontrolle sofort an die Anwendung zurück. Der Rückgabewert ist der des ersten fehlgeschlagenen Moduls mit *required*- oder *requisite*-Steuerung.
- `sufficient` – Liefert dieses Modul die Statusmeldung *success* zurück, genügt dieses für PAM zur positiven Gesamtmeldung, wenn nicht zuvor ein vorher eingetragenes Modul mit *required* oder *requisite* fehlgeschlagen ist. Anschließend ruft PAM keine weiteren Module dieses Typs mehr auf. Das Ende der Prozedur wird zum Problem, wenn ein folgendes Modul mit dem User-Passwort ein Mount-Vorgang ausführen soll. Liefert ein Modul mit *sufficient*-Steuerung keine Erfolgsmeldung, ruft PAM die nachfolgenden Module auf. Der Misserfolg eines *sufficient*-Moduls bedeutet nicht das Fehlschlagen der Gesamtfunktion.
- `optional` – Bei diesem Modul entscheiden Erfolg oder Misserfolg nicht über den Gesamtrückgabewert. Ausnahmsweise geschieht dieses nur, wenn alle anderen Module im Stapel keinen definitiven Erfolg oder Misserfolg meldeten. Das Modul sollten Sie mit Vorsicht einsetzen, weil Sie Ihren Rechner unbedacht öffnen oder komplett absperren könnten. Die Beispiele weiter unten demonstrieren diese Risiken.

Je nach Art des Programms oder Dienstes ruft dieses bei Bedarf einen der vier Modul-Typen auf. Die PAM-Bibliothek geht nun alle Module, die in der Konfigurationsdatei zur dieser Kategorie vermerkt sind, nacheinander durch. Nach Erreichen des letzten Moduls oder dem vorzeitigen erfolgreichen Abbruch bei *sufficient* liefert sie den Gesamtstatus an die aufrufende Applikation zurück.

Der Modulpfad in der darauffolgenden Dateispalte legt fest, wo ein Modul installiert ist. Standardmäßig erwartet PAM seine Module in `/lib/security`. In diesem Fall reicht die Angabe des Modulnamens. Andernfalls gilt wie sonst bei Unix – Pfadnamen mit `/` beginnend werden als absolute Pfade interpretiert. Fehlt `/` ,nimmt es einen Pfad relativ zu `/lib/security` an. So unterscheiden Sie selbst hinzugefügte Module von den üblicherweise installierten.

Die letzte Spalte *Argumente* ist optional und enthält nur bei einigen Modulen einen Eintrag. Argumente werden als Liste angegeben und dort durch Leerzeichen getrennt. Es gibt einfache Flags, wie `nullok` oder `use_first_pass` und Zuweisungen, z. B. `strict=false`. Es gibt von fast allen Modulen verstandene Argumente: `debug` liefert Diagnosemeldungen an den Systemlog-Dienst, `no_warn` unterdrückt diese. Authentifizierungs- und Passwortmodule kennen darüberhinaus `use_first_pass`. Sie versuchen dann, das Passwort des vorhergehenden `auth`-Moduls zu verwenden. Schlägt es fehl, meldet das Modul den Status *fail* zurück. Ähnlich wirkt für Authentifizierungsmodule `try_first_pass`. Das Modul versucht dann das Passwort des vorhergehenden Moduls zu verwenden. Wenn dies fehlschlägt, fordert es Benutzer auf, ihr Passwort erneut einzugeben.

Inzwischen gibt es für PAM fast alle erdenklichen Module. Ein großer Teil ist bereits im Standard-RPM dabei oder durch die Installation der LDAP-Komponenten erfolgt.

Die nachstehende Datei `/etc/pam.d/xdm` zeigt ein typisches Beispiel einer PAM-Konfiguration mit drei hintereinander geschalteten Authentifizierungsmodulen:

- Das erste Modul überprüft, wenn die User-ID ungleich Null ist, ob eine Datei `/etc/nologin` existiert. Es unterbindet in diesem Fall eine weitere Anmeldung, da das Fehlschlagen eines `required`-Moduls zum Gesamtergebnis *fail* führt.
- Mit der nächsten Zeile versucht PAM, sich anmeldende Benutzer gegen die Standard-Unix-Dateien zu authentifizieren. Gelingt dieses, arbeitet es keine weiteren Module mehr ab.
- Schlägt dieser Schritt fehl, verwendet es das nächste Modul im Stapel.

Klappt die Anmeldung des Benutzers gegen den LDAP-Server, ist der Rückmeldewert *success*.

```
#%PAM-1.0
auth     include        common-auth
account  include        common-account
password include        common-password
session  required       pam_loginuid.so
session  include        common-session
session  required       pam_resmgr.so
```

Der größte Teil der Einstellungen wird in den `Include`-Dateien gemeinsam für eine Reihe von Diensten festgelegt, wie beispielsweise in `/etc/pam.d/common-auth`:

```
#%PAM-1.0
#

# This file is autogenerated by pam-config. All changes
# will be overwritten.
#
# Authentication-related modules common to all services
#
auth    required        pam_env.so
auth    sufficient      pam_thinkfinger.so
auth    sufficient      pam_unix2.so
auth    required        pam_ldap.so     use_first_pass
```

Diese Konfiguration eignet sich für Umgebungen, in denen es bis auf den Administrator keine lokal eingetragenen Benutzer gibt. Zuerst schaut PAM, ob eine Authentifizierung gegenüber den klassischen Unix-Dateien Erfolg hat. Dieses trifft nur bei *root* zu. Da der *Root*-Benutzer nicht im LDAP gespeichert ist und kein zentrales Home-Verzeichnis hat, darf er auch nicht mehr bei den auf `pam_unix2` folgenden Modulen vorbeikommen. Das ist durch *sufficient* unterbunden.

pam_unix2 ist zusätzlich als Account-, Passwortänderungs- und Session-Modul eingesetzt. Die Accounting-Funktion dieses Moduls prüft anhand der Felder in der Shadow-Datei, ob der Account noch gültig ist. Ebenfalls testet das Modul, ob das Passwort abgelaufen ist. In diesem Fall kann es die Authentifizierung verschieben, bis der Benutzer sein Passwort aktualisiert hat. Es kann ebenfalls eine Warnung an den Benutzer ausgeben, dass er sein Passwort ändern sollte. Als Session-Modul zeichnet es einfach nur den Benutzernamen und den Diensttyp über den Syslog-Dienst auf. Das Modul pam_env.so setzt Umgebungsvariablen, die in /etc/security/pam_env.conf stehen. pam_mail.so ist als *optional* eingetragen, darf also ohne Konsequenzen fehlschlagen. Es teilt lediglich Benutzern mit, ob neue Mail für sie vorliegt. Da dieses jedoch nur für lokal vom E-Mail-Server abgelegte Posteingangsordner klappt, ist dieses Modul eher selten einsetzbar.

Bei fehlkonfiguriertem PAM können sich Benutzer möglicherweise auch ohne ausreichende Authentifizierung anmelden. Das demonstriert das folgende Beispiel:

```
auth     required   pam_nologin.so
auth     optional   pam_unix2.so     set_secrpc
auth     optional   pam_ldap.so      use_first_pass
[ ... ]
```

Drei Module sind hintereinander geschaltet: Das erste Modul testet auf die Existenz der /etc/nologin. Gibt es diese Datei nicht, liefert es den Status *success*. Selbst wenn nun alle drei folgenden Module fehlschlagen, meldet PAM als Gesamtergebnis *success*. So könnte ein Benutzer auch mit falschem Passwort ins System kommen. Wenn Sie es noch einfacher haben wollen, setzen Sie auf pam_permit. Dieses garantiert den Erfolg der Anmeldung, wenn Sie irgendeinen existierenden Benutzer beim Login angegeben haben.

Umgekehrt können Sie durch eine Fehlkonfiguration ihre Maschine für jegliches Login (auch *root*) sperren:

```
auth     required   pam_nologin.so
auth     required   pam_unix2.so     set_secrpc
auth     required   pam_ldap.so      use_first_pass
```

Die Kontrolleinstellung required sorgt dafür, dass ein Benutzer sowohl in den Unix-Dateien passwd und shadow als auch im LDAP bekannt sein muss. In verteilten Authentifizierungsarchitekturen liegt dies oft nicht vor: Der Systemadministrator ist immer nur lokal eingetragen, normale Benutzer sinnvollerweise nie. Die Aussperrung von der Maschine ist perfekt. Deshalb veranstalten Sie solche Tests am besten auf einem unkritischen Dienst (z. B. xdm). Anschließend übertragen Sie die erfolgreich getestete Kombination auf die gewünschten anderen. Dabei müssen Sie nicht alle Dienste einheitlich einstellen: Vielleicht dürfen sich Benutzer an Ihrem Server nur per SSH anmelden, aber nicht per FTP. Dann tragen Sie das LDAP-Modul nur in der /etc/pam.d/sshd ein.

Wenn Sie PAM für Netzwerkdienste verwenden, wie für die FTP-Authentifizierung, sollten Sie beachten, dass PAM selbst die Übertragung von Passwörtern nicht schützen kann. Dieses ist immer Aufgabe der Applikation selbst.

Für alle PAM-Dateien und -Verzeichnisse dürfen nur Systemadministratoren Schreibrechte haben. Nur *root* darf die Verzeichnisse `/lib/security`, `/etc/security` und `/etc/pam.d` sowie die darin befindlichen Dateien besitzen. Außerdem sollten Sie keine PAM-Module verwenden, die von Benutzern schreibbare Programmbibliotheken verwenden, da dies Angreifern erleichtert, an verschiedenen Stellen vom PAM ihre Schwachstellen zu finden. Darüber hinaus definieren Sie ein Fallback für unkonfigurierte Dienste:

```
auth      required       pam_warn.so
auth      required       pam_unix2.so
account   required       pam_warn.so
account   required       pam_unix2.so
password  required       pam_warn.so
password  required       pam_cracklib.so
password  required       pam_unix2.so      use_first_pass
session   required       pam_warn.so
session   required       pam_unix2.so
```