8 Network File System einrichten

Um Client-Rechnern ganze Verzeichnisse von Servern zum Lesen oder Lesen und Schreiben zur Verfügung zu stellen, benutzt man im Unix-Umfeld und generell in heterogenen Umgebungen ein spezielles Dateisystem, das *Network File System*, kurz NFS. Vor Samba (siehe Kapitel 9) war dies nahezu die einzige Möglichkeit, Windows-Clients Verzeichnisse anzubieten, die auf Linux-Servern lagen.

Im zweiten Teil dieses Buchs lernen Sie, stabile und kostengünstige Linux-Arbeitsplätze zu nutzen. Dabei ist es erforderlich, den Linux-Clients

- Verzeichnisse auf Festplatten von Linux-Servern zum Lesen und Schreiben
- und von CD-ROMs/DVDs nur zum Lesen zur Verfügung zu stellen.

Im Teil 2 können Sie lesen, wie Sie *Thin Clients* ohne Festplatte einrichten, die sogar ihr gesamtes Linux-Dateisystem per Network File System von einem Linux-Server beziehen.

Wenn Sie mehrere Linux-Server in Ihrem Netz einsetzen, z. B. zur Lastverteilung, oder mit Linux-Clients arbeiten, benötigen Sie auf jedem dieser Rechner eine eigenständige Benutzerverwaltung. Einfacher ist es, alle Benutzer nur einmal auf einem zentralen Anmeldeserver anzulegen, von dem die anderen Rechner dann die Anmeldedaten beziehen. Im Windows-Bereich würden Sie dies mit Anmeldeservern für Arbeitsgruppen bzw. Domänen erledigen (siehe Kapitel 9).

Eine Lösung für dieses Problem ist NIS, der *Network Information Service*. Dieser Dienst war früher unter dem Namen *YP* (*YellowPages*) zu finden. Aus rechtlichen Gründen darf dieser Name nicht mehr benutzt werden, trotzdem tragen viele der Programmkomponenten und Variablen immer noch YP im Namen.

Eine NIS-Installation nutzt meist NFS, da die Benutzer immer das gleiche Home-Verzeichnis erwarten, egal auf welchem Rechner sie sich anmelden. Daher mountet man in der Regel das Home-Verzeichnis vom Anmeldeserver per NFS.

Open Source-NFS-Server und -Clients für Windows-Abarten sind den Autoren bisher nicht bekannt. Daher ist NFS hier nur für Linux-Server und Linux-Clients beschrieben. Stabile lizenzpflichtige NFS-Server und -Clients für Windows gibt es u. a. von Hummingbird (`http://www.hummingbird.com`).

Um NFS im Linux-Umfeld benutzen zu können, muss man den Linux-Server und den Linux-Client vorbereiten:

- Nach dem Einrichten von NFS auf dem Server
- müssen Sie bestimmen, welche Verzeichnisse der Server welchen Clients für welche Zugriffe zur Verfügung stellen soll und
- dann auf den Clients diese Verzeichnisse jeweils in den lokalen Verzeichnisbaum einhängen.

8.1 Einsatzfelder für NFS

NFS brauchen Sie immer dann, wenn sich Linux-Rechner untereinander Laufwerke – dazu gehören auch CD-ROM-Laufwerke – gegenseitig zur Verfügung stellen. Zwar könnten Sie hierzu auch Samba (siehe Kapitel 9) verwenden. Generell ist aber der Zugriff per NFS deutlich flexibler als der per Samba.

Da man auf NFS-Dateisysteme schon beim Booten zugreifen kann, lassen sich so große Teile des Filesystems von einem fernen Rechner beziehen.

Der Dateizugriff per NFS ist für Clients vollständig transparent und funktioniert mit sehr unterschiedlichen Serverstrukturen: es spielt keine Rolle, ob es sich dabei um Linux- oder Windows-Server handelt. Ein Sicherheitsproblem bei NFS besteht aber darin, dass es keine Anmeldung erwartet und Zugriffsregeln nur über die IP-Adresse ermöglicht.

8.2 NFS-Server installieren und konfigurieren

Wie viele andere Distributionen installiert auch OpenSUSE in der Voreinstellung einen NFS-Server.

Den NFS-Server gibt es prinzipiell in zwei Varianten, einmal als Kernel-NFS, andererseits als Userspace-NFS:

Das Kernel-NFS ist direkt im Betriebssystemkern verankert und damit deutlich performanter, setzt aber einen entsprechend kompilierten Kernel voraus. Da OpenSUSE die Standard-Kernel mit Kernel-NFS konfiguriert hat, installiert YaST standardmäßig kein Userspace-NFS.

Das Userspace-NFS erfordert keinerlei Veränderungen am Kernel, es lässt sich also leicht auch nachträglich installieren.

Der Funktionsumfang beider Versionen ist identisch. Sie können sogar beide Versionen nebeneinander installieren; welche Version Sie dann starten, legen Sie über Variablen in der Konfigurationsdatei von YaST fest.

8.2.1 Kernel NFS

Falls Sie auf Ihrem System bisher keinerlei NFS-Server installiert haben, sollten Sie nun das Paket `nfs-kernel-server` einspielen, welches Sie in der Paketgruppe `Netzwerk` finden.

Wenn Sie Ihre Programme gern mit YaST konfigurieren, dann sollten Sie auch dessen NFS-Server-Modul installieren, welches Sie im Paket `yast2-nfs-server` in der Paketgruppe `System` finden.

Sollten Sie einen eigenen Kernel erstellen, aktivieren Sie in der Konfigurationsdatei für den Kernel die folgenden Schalter:

- `CONFIG_NFS_FS` und
- `CONFIG_NFSD`.

8.2.2 Userspace-NFS

Sollten Sie aus irgendeinem Grund doch das Userspace-NFS nutzen wollen, so müssen Sie das Paket `nfs-server` nachinstallieren. Die Autoren haben die Erfahrung gemacht, dass festplattenlose Linux-Clients, die ihr Dateisystem per NFS beziehen, nicht einwandfrei mit Servern mit Kernel-NFS zusammenarbeiten. In solch einem Fall ist die Umstellung sinnvoll, auch wenn Kernel-NFS als performanter gilt.

Der weitere Teil dieses Kapitels bezieht sich auf Kernel-NFS, das keine weiteren Installationsschritte erfordert.

8.2.3 Der Portmapper

Um NFS nutzen zu können, benötigt man einen Dämon als Servicevermittler für Client/Server Dienste, die mit *Remote Procedure Calls* (Fernaufrufe für Prozeduren) arbeiten, den *RPC-Portmapper*.

Bei einem derartigen Dienst kann ein Client über ein Serverprogramm Prozeduren auf dem Server ausführen. Zu jeder der Prozeduren gehört eine eindeutige Programmnummer. Der Portmapper ordnet diesen Programmnummern Ports zu. Wenn Sie die aktuelle Zuordnung bei laufendem NFS-Server mit dem Befehl

```
rpcinfo -p
```

abrufen, erhalten Sie eine Tabelle mit folgendem Aufbau:

```
boss:~ # rpcinfo -p
   Program Vers Proto   Port
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100003    2   udp   2049  nfs
    100003    3   udp   2049  nfs
```

```
    100021    1   udp   1026  nlockmgr
    100021    3   udp   1026  nlockmgr
    100021    4   udp   1026  nlockmgr
    100005    1   udp   1027  mountd
    100005    1   tcp   1025  mountd
    100005    2   udp   1027  mountd
    100005    2   tcp   1025  mountd
    100005    3   udp   1027  mountd
    100005    3   tcp   1025  mountd
…
```

In der ersten Spalte dieser Tabelle sehen Sie jeweils die Programm-Nummern für die RPC-Calls, in der vierten Spalte die zugeordneten Ports. Die fünfte Spalte beschreibt die zugeordnete Funktion.

8.2.4 Start des NFS-Servers

Den ersten Start des NFS-Servers können Sie von der Konsole aus vornehmen, dauerhaft aktivieren Sie ihn am einfachsten mit YaST.

Um den NFS-Server einzuschalten, muss man den Portmapper und dann den Server in dieser Reihenfolge starten:

Falls der Portmapper nicht ohnehin läuft, ruft man ihn über das Startskript

```
rcportmap  start
```

auf und danach den eigentlichen Server mit

```
rcnfsserver  start
```

Das Boot-Skript aktiviert diese beiden Programme automatisch, wenn im Runlevel-Editor des YaST-Kontrollzentrums die Häkchen für die Level 3 und 5 gesetzt sind.

Damit ist der NFS-Server einsatzbereit, auch wenn er bisher noch keine Verzeichnisse exportiert.

Im nächsten Schritt müssen Sie dem Server mitteilen, welche Verzeichnisse er an welche Clients exportieren soll.

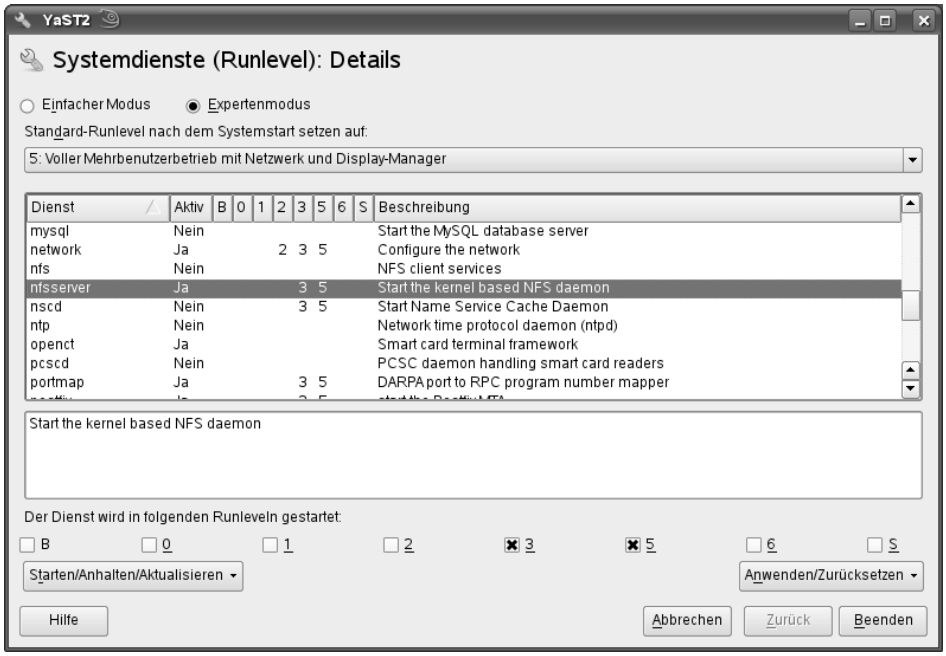

Abbildung 8.1: Aktivieren von NFS-Server und Portmap

8.3 Verzeichnisse exportieren

Wenn Sie einen funktionsfähigen NFS-Server eingerichtet haben, müssen Sie noch Verzeichnisse freigeben.

Verzeichnisse des NFS-Servers können Sie exportieren, indem Sie

- diese in einem Dialog von YaST erfassen oder
- per Editor direkt in die Datei `/etc/exports` eintragen.

Welche der Möglichkeiten Sie wählen, spielt keine Rolle – letztlich landen die Informationen immer in der Datei `/etc/exports`.

8.3.1 Verzeichnisse per YaST-Dialog exportieren

Gehen Sie über das YaST-Kontrollzentrum und die Menüfolge *Netzwerkdienste* • *NFS-Server*.

Hier erwartet zuerst ein kleiner Dialog *NFS-Server* von Ihnen die Entscheidung, ob Sie den NFS-Server starten wollen oder nicht. Hier können Sie auch den zugehörigen Port in der SUSE-Firewall freigeben, wenn entfernte Rechner auf die Freigaben zugreifen sollen.

Nach einem Klick auf *Weiter* öffnet YaST für Sie ein Dialogfenster mit zwei Teilfenstern:

Im ersten Fenster tragen Sie die Verzeichnisse ein, die Sie exportieren wollen und im zweiten die Rechner, die diese Verzeichnisse nutzen dürfen.

Abbildung 8.2: Bearbeiten der Exports-Datei in YaST

Wenn Sie auf *Beenden* klicken, übernimmt YaST Ihre Eingaben in die Konfigurationsdatei, wobei es eigenmächtig einige weitere Optionen setzt.

8.3.2 Verzeichnisse manuell per Editor exportieren

Statt mit dem Dialog von YaST können Sie die Datei `/etc/exports` direkt bearbeiten.

Diese nach der Standardinstallation leere Datei können Sie z. B. folgendermaßen tabellarisch einrichten:

```
# Beispieldatei  /etc/exports
# Zeilen, die mit dem Zeichen # beginnen, werden ignoriert
#
/home            *.lokales-netz.de(rw,no_subtree_check)
/windows         www.linuxbu.ch(ro,no_subtree_check)
/media/cdrom     *(ro,no_subtree_check)
```

Diese tabellenartige Darstellung in der Form

```
/pfad/zum/verzeichnis   Rechnername(n)(option1,option2,...)
```

gibt drei Daten an:

- Pfad zum Verzeichnis (siehe 8.3.1),
- Rechner, die zugreifen dürfen (siehe 8.3.2) und
- Optionen (siehe 8.3.5)

Für jedes Verzeichnis können Sie mehrere Rechner beziehungsweise Domains mit den zugehörigen Optionen angeben. Im vorliegenden Beispiel dürfen alle Rechner der Domain `lokales-netz.de` lesend und schreibend (`rw`) auf `/home` zugreifen, der Rechner `www.linuxbu.ch` nur lesend (`ro`) auf `/windows`.

Wenn Sie die `/etc/exports` direkt verändert haben, müssen Sie den NFS-Server neu starten, damit er diese Veränderungen registriert. Dazu geben Sie ein:

```
rcnfsserver  restart
```

8.3.3 Verzeichnisse

Die Angaben des obigen Beispiels exportieren drei Verzeichnisse: Das gesamte Home-Verzeichnis mit den Benutzerdaten, ein Windows-Verzeichnis und das CD-ROM-Laufwerk.

Die Pfadangabe dürfen Sie nicht weglassen, da sonst die Freigabe sinnlos ist. Alle weiteren Angaben dürfen entfallen.

8.3.4 Welche Rechner dürfen zugreifen?

Die zweite Angabe hinter dem Verzeichnisnamen beschränkt die Rechner, die auf diese Freigabe zugreifen dürfen.

Auf das Home-Verzeichnis sollen nur Rechner aus dem lokalen Netz zugreifen dürfen. Da die entsprechende Angabe für das CD-ROM-Laufwerk fehlt, dürfen hier alle Rechner, also auch beliebige Rechner aus dem Internet, zugreifen.

Die Rechner, die auf das Verzeichnis zugreifen dürfen, können Sie auf folgende Arten angeben:

1. Einem einzelnen Rechner erlauben Sie den Zugriff, indem Sie seinen Namen oder seine IP angeben.
2. Einer Gruppe von Rechnern können Sie den Zugriff erlauben, indem Sie Rechnernamen angeben, welche die Joker (Wildcards) "`*`" oder "`?`" enthalten. Im Beispiel erlauben Sie u. a. dem Rechner `rosine.lokales-netz.de` den Zugriff, da dieser Name der Angabe `*.lokales-netz.de` entspricht. Das Wildcardzeichen "`*`" steht für eine beliebige Zeichenfolge, also auch für `rosine`.
3. Sie können einen IP-Bereich angeben, indem Sie eine IP-Adresse und eine zugehörige Netzwerkmaske angeben. Mit `192.168.1.0/255.255.255.0` (oder auch `192.168.1.0/24`) erlauben Sie allen Rechnern, deren IP in den ersten drei Gruppen `192.168.1.x` lautet, den Zugriff.

4. Sie erlauben allen Rechnern den Zugriff, indem Sie in dieser Spalte keine Angabe machen, oder ein "*" als Jokerzeichen eintragen.

8.3.5 Optionen

Die dritte Angabe nennt Optionen, hier im Beispiel für Zugriffsrechte.

Die wichtigsten Optionen sind:

Befehl	*Erläuterung*
`rw`	*Read-Write* gibt den Clients Lese- und Schreibrechte für das Verzeichnis.
`ro`	*Read-Only* ist die Voreinstellung, bei der Clients nicht in das Verzeichnis hineinschreiben dürfen.
`root_squash`	Voreinstellung, die privilegierte Zugriffe des Super-Users *root*
	unterbindet. *Root*-Zugriffe führt der Server nur mit den Rechten des Benutzers `nobody` und der Gruppe `nogroup` aus.
`no_root_squash`	Das Gegenteil zu obiger Option. Der Super-User *root* kann vom
	Client aus mit seinen vollen Rechten auf die Dateien auf dem Server zugreifen.
`all_squash`	Der Server führt alle Zugriffe vom Client nur mit den Rechten des Users `nobody` aus.
`noaccess`	Verbietet den Clients den Zugriff auf Unterverzeichnisse; damit kann man einzelne Unterverzeichnisse eines freigegebenen Verzeichnisses sperren.
`async`	Diese Option erlaubt es dem NFS-Server, Anfragen zu beantworten, bevor die zugehörigen Änderungen auf dem Datenträger erledigt sind. Das steigert die Performance, erhöht aber auch etwas das Risiko von Datenverlust.
`sync`	Der NFS-Server beantwortet Anfragen erst dann, wenn die zugehörigen Änderungen auf dem Datenträger erledigt sind.
`secure`	`Lässt Zugriffe nur über Portnummern kleiner 1024 zu (Vorgabe)`
`insecure`	`Lässt NFS-Zugriffe auf allen Ports zu.`

Tabelle 8.1: Wichtige Optionen für Zugriffssteuerung

Eine vollständige Liste aller Optionen finden Sie in der Manpage von `exports`.

Die Optionen notiert man innerhalb runder Klammern. Mehrere Optionen trennt man durch Kommata ohne Leerzeichen. Zulässig wäre z. B. die Angabe

```
/media/cdrom          *.lokales-netz.de(ro,no_root_squash)
```

Hier darf der Superuser mit seinen Rechten nur lesend auf das CD-ROM-Laufwerk zugreifen.

8.4 Netzwerkverzeichnisse einbinden

Netzwerkverzeichnisse, die auf irgendeinem Rechner freigegeben sind, können Anwender

- mit YaST nach der Befehlsfolge *YaST Kontrollzentrum* • *Netzwerkdienste* • *NFS-Client* im Dialogfenster *Konfiguration des NFS-Clients* bequem menügestützt einbinden oder
- genauso wie CD-ROM-Laufwerke mit dem Befehl `mount` in ihr lokales Dateisystem einbinden (mounten), wenn sie über die notwendigen Zugriffsrechte verfügen.

8.4.1 NFS-Zugriff auf linuxbu.ch

Um Ihnen das Testen zu erleichtern, haben die Autoren ein Verzeichnis auf `www.linuxbu.ch` exportiert und für alle Rechner freigegeben; die zugehörige Datei `/etc/exports` hat folgenden Inhalt:

```
# See exports(5) for a description.
# This file contains a list of all directories
# exported to other computers.
# It is used by rpc.nfsd and rpc.mountd.
/srv/ftp      *(ro,async,insecure)
```

Auf dieses Verzeichnis können Sie auch mit anonymem FTP (Kapitel 5) zugreifen.

Wenn Sie mit dem Internet verbunden sind, können Sie dieses Verzeichnis in Ihr lokales Filesystem mit YaST oder per `mount`-Befehlt einbinden. Im YaST-Dialog *Konfiguration des NFS-Clients* klicken Sie auf die Schaltfläche *Hinzufügen*, tragen dann in einem weiteren Fenster den

- Hostnamen des NFS-Servers,
- das entfernte Dateisystem und den
- Mountpoint (Einhänge-Ordner) ein und
- bestätigen mit der Schaltfläche OK

Abbildung 8.3: Yast Dialog NFS-Client

Alternativ können Sie als *root* folgenden Befehl eingeben:

```
mount   -t  nfs  www.linuxbu.ch:/srv/ftp   /mnt
```

Unabhängig davon, welchen Weg Sie beschritten haben, können Sie anschließend mit den üblichen Linux-Befehlen zum Anzeigen von Inhaltsverzeichnissen bzw. zum Kopieren von Dateien auf das Verzeichnis `/mnt` zugreifen. Alle Zugriffe auf das Verzeichnis `/mnt` gehen dann auf den Server zu diesem Buch.

Wollen Sie das Verzeichnis nach Ihren Experimenten wieder freigeben, bevor Sie die Internet-Verbindung abbauen, geben Sie ein:

```
umount   /mnt
```

Mit dem Befehl `showmount` kann man abfragen, welche Verzeichnisse ein Rechner per NFS anbietet. Dazu gibt man ein:

```
/usr/sbin/showmount   -e   www.linuxbu.ch
```

Der Rechner gibt dann Folgendes aus:

```
root@boss:~ > showmount   -e   www.linuxbu.ch
Export list for www.linuxbu.ch:
/srv/ftp *
```

Auf das Verzeichnis `/srv/ftp` können Sie also von jedem Rechner aus zugreifen, was `showmount` mit dem * kennzeichnet.

8.4.2 Der Befehl mount

Ein NFS-Client muss wissen, welches Dateisystem er beziehen möchte und an welcher Stelle er es in sein lokales Dateisystem einbinden will. Für diese Festlegungen dient der Befehl `mount`.

Sie kennen aus dem vorangegangenen Abschnitt

```
mount  -t  nfs   www.linuxbu.ch:/srv/ftp    /mnt
```

und vom Einhängen eines CD-ROM-Laufwerks:

```
mount  -t   iso9660   /dev/cdrom   /media/cdrom
```

Der `mount`-Befehl erwartet also Quelle, Ziel und den Typ des Dateisystems (Parameter `-t`):

Der erste Parameter nennt die Quelle, also was in das Dateisystem eingebunden werden soll; in den Beispielen ist dies ein Verzeichnis eines anderen Rechners oder ein CD-ROM-Laufwerk. Zwischen dem Rechnernamen und dem Verzeichnis steht immer ein Doppelpunkt; beim CD-ROM-Laufwerk auf dem gleichen Linux-System geben Sie ein Gerät, hier `/dev/cdrom` an, bei einem CD-ROM-Laufwerk auf einem anderen Linux-System den Rechnernamen und die Gerätebezeichnung, hier `www.linuxbu.ch:/dev/cdrom`.

Der zweite Parameter gibt an, über welches Verzeichnis die Ressource eingebunden werden soll, den so genannten Einhänge-Ordner oder *Mountpoint*. Die Angabe ist beliebig, das Verzeichnis muss nur existieren und leer sein. Die OpenSUSE-Distribution legt standardmäßig für diesen Zweck die Verzeichnisse `/media/cdrom` und `/mnt` an. Nach erfolgreichem Mounten finden Sie die eingebundenen Daten in dem vorher leeren Verzeichnis.

Mit dem Parameter `-t` (Typ) können Sie u. a. die folgenden Dateisysteme angeben (der Kernel muss das jeweilige Dateisystem unterstützen, was bei OpenSUSE-Kerneln der Fall ist):

Typ des Dateisystems	*Bedeutung*
`nfs`	Network File System
`iso9660`	Dateisystem auf CD-ROM
`vfat`	Windows-Dateisystem
`ext2`	Linux-Dateisystem
`reiserfs`	Reiser-Dateisystem
`proc`	Pseudo-Dateisystem
`smbfs/cifs`	Samba File System

Tabelle 8.2: Dateisysteme

8.4.3 Verzeichnisse permanent in das System einhängen

Nach den bisherigen Beschreibungen darf nur der Super-User *root* irgendwelche Ressourcen mounten. Praktikabler ist, allen Benutzern das Einhängen (Mounten) von CDs und Disketten zu erlauben. Andere Ressourcen will man schon beim Booten ohne manuellen Eingriff ins System einbinden.

Für dieses permanente Einbinden von Dateisystemen ist die Datei `/etc/fstab` zuständig, über die man auch Festplattenpartitionen einbindet. Bei einer Standardinstallation erzeugt YaST eine Datei der folgenden Art:

```
/dev/hda2   /              ext3       acl,user_xattr      1 1
/dev/hda1   swap           swap       defaults            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
```

Die Spalten entsprechen den Parametern des `mount`-Befehls.

- In der ersten Spalte steht die Datenquelle bzw. das jeweilige Gerät. Eine Angabe wie `/dev/hda2` bezeichnet die Partition *Zwei* der ersten IDE-Festplatte (siehe Abschnitt 2.3, »Festplatten vorbereiten«).

- In der zweiten Spalte stehen die Einhänge-Ordner (Mountpoints), über die Sie die jeweiligen Geräte im System ansprechen können.
- Die dritte Spalte gibt die Dateisysteme an. Neu gegenüber dem `mount`-Befehl ist hier die Angabe `subfs`. Bei Einträgen mit diesem Dateityp versucht das System selbst, das Vorhandensein und das Dateisystem zu ermitteln. Das ist bei Wechseldatenträgern wie Disketten und CDs sinnvoll: sobald ein Datenträger eingelegt ist, mountet ihn das System automatisch. In der vierten Spalte folgen die Optionen, wieder durch Kommata getrennt ohne Leerzeichen. Interessant sind hier die Optionen `noauto` und `user`. Mit der Option `noauto` verhindern Sie, dass die entsprechende Zeile schon beim Hochfahren des Systems aktiviert wird. Das wäre für Wechselmedien nicht sinnvoll. Mit der Option `user` erlauben Sie allen Benutzern, dieses Dateisystem zu mounten. Die Option `exec` erlaubt zusätzlich das Ausführen von Programmen im Dateisystem. In der oben dargestellten Konfiguration können Sie also keine Programme von einer Diskette aus starten.
- Die Spalten fünf und sechs steuern das Sichern bzw. Überprüfen von Dateisystemen.
- Bei `ext2/ext3/reiserfs`-Partitionen sollte in der fünften Spalte eine `1` stehen, ansonsten eine `0`. Die `0` gibt an, dass der Dämon das entsprechende Verzeichnis beim Mounten nicht testen soll. Wenn in der fünften Spalte eine `1` steht, dann sollte in der sechsten Spalte eine `2` stehen, außer beim Wurzelverzeichnis, das kennzeichnen Sie mit einer `1`. Das Wurzelverzeichnis testet er vorrangig, alle anderen Verzeichnisse später.

Um ein Verzeichnis per NFS automatisch zu beziehen, können Sie in die Datei `/etc/fstab` eine weitere Zeile aufnehmen:

```
/dev/hda2   /                 ext3    acl,user_xattr      1 1
/dev/hda1   swap              swap    defaults            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
www.linuxbu.ch:/srv/ftp  /mnt nfs     ro                  0 0
```

Auch für das Bearbeiten der `/etc/fstab` gibt es im YaST-Kontrollzentrum eine Funktion, die Ihnen die Arbeit erleichtert.

8.5 NFS-Probleme aufspüren und beheben

Sind auf einem Server notwendige Dämonen nicht aktiviert oder fehlen gewünschte Freigaben, erleben Anwender dies als Fehler beim Mounten von Verzeichnissen. Wenn Sie auf dem FTP-Server Root-Rechte besitzen, können Sie den Status der Server-Programme überprüfen.

```
rcportmap status
```

Sie sollten ein einfaches `OK` als Antwort erhalten.

Testen Sie danach, ob auch der NFS-Server läuft, mit

```
rcnfsserver status
```

Sie sollten hier die Meldung `NFS server up` erhalten.

Sollte einer der Dienste nicht aktiv sein, so prüfen Sie die Einstellungen in YaST und starten die Dienste neu.

Sollte bis hierher alles korrekt aussehen, so fehlt es an der Freigabe – eventuell wurde der NFS-Server nach Änderungen nicht neu gestartet. Ob eine Freigabe auf Ihrem Rechner aktiv ist, können Sie jederzeit testen mit

```
/usr/sbin/showmount   -e
```

Wollen Sie einen fremden Rechner untersuchen, so hängen Sie wie oben beschrieben den Rechnernamen als Parameter an den Befehl an:

```
/usr/sbin/showmount   -e  www.linuxbu.ch
```

Falls die Freigabe nur für bestimmte Rechner gilt, lohnt sich auch ein Blick in die Datei `/var/log/messages` des freigebenden Rechners. Diese protokolliert alle Mount-Versuche und auch den Grund für eine eventuelle Ablehnung.

Hier hat der Client versucht, auf eine nicht vorhandene Freigabe zuzugreifen.

```
Dec 22 13:46:19 boss rpc.mountd: refused mount request from
192.168.1.1 for /home/debacher (/): no export entry
```

Ein erfolgreicher Mountversuch, hier für das CD-Laufwerk, hinterlässt ebenfalls einen Eintrag.

```
Dec 22 13:48:09 boss rpc.mountd: authenticated mount request from
192.168.1.1:842 for /media/cdrom (/media/cdrom)
```

8.6 NIS

Dieses Buch beschreibt an mehreren Stellen alternative Wege zur Benutzerverwaltung. Hier geht es um den älteren *Network Information Service* NIS. Im Kapitel 3.5 finden Sie dagegen Informationen zu LDAP.

Der *Network Information Service* NIS benötigt einen NIS-Server, der die Benutzerdaten für seine NIS-Domain verwaltet. Zu dieser Domain können beliebig viele NIS-Clients gehören. In größeren Domänen kann es sinnvoll sein, zusätzlich Slave-Server einzusetzen, die beim Ausfall des Hauptservers dessen Aufgabe übernehmen können. Auf Slave-Server geht dieses Buch nicht ein.

In den Beispieldateien dieses Kapitels heißt die NIS-Domain `lokales-netz`. Die Bezeichnung können Sie recht frei wählen, es muss keine offizielle DNS-Domain sein.

Es gibt zwei Implementierungen von NIS. Das ursprüngliche NIS überträgt Benutzerdaten unverschlüsselt übers Netz. Die aktuellere Implementierung NIS+ überträgt die Benutzerdaten dagegen verschlüsselt. NIS+ ist sicherer, dafür ist die manuelle Konfiguration aufwändiger. Der folgende Text beschreibt NIS.

8.7 NIS Server-Installation

Auf dem Anmeldeserver müssen die Pakete `ypserv`, `ypbind` und `yp-tools` installiert sein, bei der Standardinstallation fehlt davon nur `ypserv`. Sie finden das Paket in der Paketgruppe *Netzwerk*. Für die NIS-Konfiguration mittels YaST installieren Sie zusätzlich das Paket `yast2-nis-server` aus der Paketgruppe `system`.

Nach der Installation der Pakete müssen Sie im YaST-Kontrollzentrum unter *Netzwerkdienste* • *NIS-Server* einige Parameter einstellen, wobei Sie mit YaST sogar beide Schritte kombinieren können.

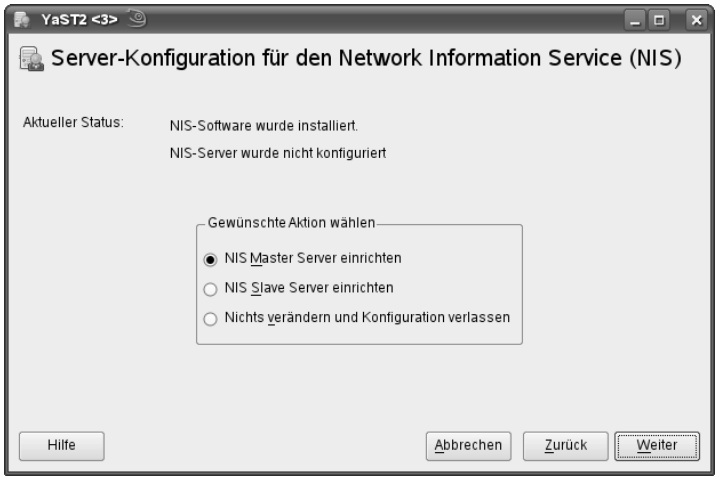

Abbildung 8.4: NIS Server – Master Server

Im ersten Formular müssen Sie nur auswählen, dass Sie einen *NIS Master Server* einrichten wollen. Über einen Klick auf *Weiter* kommen Sie zum nächsten Formular.

Abbildung 8.5: NIS Server – Domainname

Hier geben Sie einen Domainnamen an, den Sie später auch bei auf den Clients verwenden. Weiter müssen Sie verhindern, dass YaST die Konfigurationsdatei für Clients erzeugt. Dazu darf die Option *Dieser Rechner ist zugleich NIS-Client* nicht aktiviert sein.

Das Ändern der Passwörter sollten Sie hier erst einmal nicht erlauben, Sie finden ausführlichere Informationen dazu im Abschnitt »NIS-Feintuning« (Kap 8.10).

Auf der nächsten Seite legen Sie fest, welche *Maps* der Server weitergeben soll. NIS bezeichnet Datenbanken wie die `passwd` und die `group`-Datei als *Map*. Sie können hier erst einmal alle anwählen, und dann auf *Weiter* klicken.

Im letzten Formular legen Sie fest, welche Rechner auf Ihren NIS-Server zugreifen dürfen.

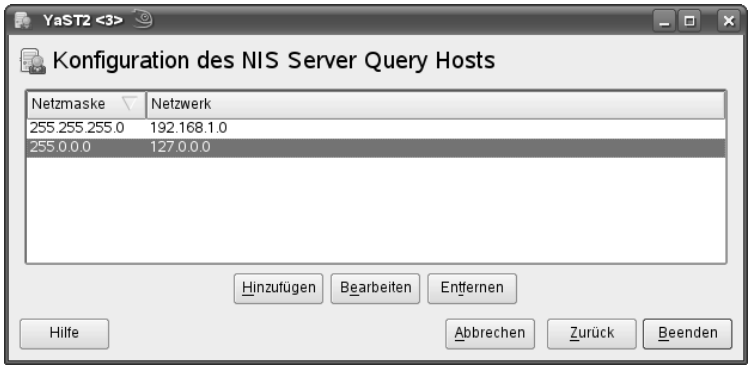

Abbildung 8.6: NIS Server – Query Hosts

YaST überträgt diese Einstellungen in die Datei `/var/yp/securenets`. In der Voreinstellung steht dort am Ende die Zeile

```
255.0.0.0               127.0.0.0
```

und gibt nur dem Rechner selber den Zugriff. Angeben müssen Sie hier als erste Zahl eine Netzmaske und als zweite Zahl eine IP-Adresse.

Mit

```
           192.168.1.0
```

erlauben Sie nur Rechnern aus Ihrem lokalen Netz den Zugriff auf den NIS-Server. Sie müssen dann natürlich darauf achten, dass in der Konfigurationsdatei die ursprüngliche Zeile entfernt ist. Sobald die Client-IP nämlich eine der Regeln erfüllt, darf der Rechner zugreifen.

Aus Sicherheitsgründen sollten Sie nur den Rechnern aus Ihrem lokalen Netz den Zugriff erlauben, niemals beliebigen Computern.

Da YaST den `ypserv` bei der Konfiguration automatisch startet, ist das System fast schon einsatzbereit, er kennt nur den Namen seiner NIS-Domain noch nicht. Sie können jetzt den NIS-Server durch einen Reboot aktivieren, oder an der Konsole eingeben:

```
domainname  lokales-netz
```

Nun müssen Sie noch erreichen, dass der NIS-Server die Daten aus den Benutzerdateien

- `/etc/passwd`
- `/etc/shadow`
- `/etc/group`
- …

bekommt. Dazu dient ein Aufruf des Programmes `make`. Dem Programm geben Sie über den Schalter `-C /var/yp` das Verzeichnis an, mit dem es arbeiten soll. Der Schalter `-s` (Silent) unterbindet Ausgaben.

```
make  -s  -C  /var/yp
```

Dies übersetzt die Benutzerdaten in die Dateien für NIS. Sie finden die erzeugten Dateien im Verzeichnis `/var/yp/lokales-netz/`. Die Dateien liegen in einem speziellen Datenbank-Format vor, das schneller auswertbar ist als eine einfache Textdatei.

Da NIS leider nichts über Änderungen in den Benutzerdateien erfährt, müssen Sie diesen Befehl regelmäßig nach jeder Änderung aufrufen, im einfachsten Fall über einen Cronjob. Ergänzen Sie die Crontab um die folgenden Zeile:

```
*/15  *  *  *  *  make  -s  -C  /var/yp
```

Damit sind neue Benutzer und geänderte Passworte spätestens nach 15 Minuten in der gesamten NIS-Domain bekannt.

Welche Daten der NIS-Server verteilen darf, legen Sie mit der Datei `/var/yp/Makefile` fest, die vom `make`-Aufruf ausgewertet wird. Sie können hier mit

```
MINUID=1000
MINGID=1000
```

festlegen, dass er nur Benutzer bzw. Gruppen ab der genannten ID exportiert.

Die Hauptrisiken von NIS ergeben sich aus den Zeilen

```
MERGE_PASSWD=true
MERGE_GROUP=true
```

NIS kann nämlich nicht mit Shadow-Passwörtern umgehen und fügt daher die Daten aus den Dateien `/etc/passwd` und `/etc/shadow` wieder zu einer Datei zusammen, zumindest für den Export. Da dadurch alle lokalen Benutzer die (verschlüsselten) Passworte lesen können, erleichtert dies Angriffe. Weitere Informationen zum Knacken verschlüsselter Passwörter finden Sie im Kapitel 3 im Abschnitt über das Erkennen schwacher Passwörter mit `john`.

Welche Exportdateien NIS anlegt, bestimmt die Zeile

```
all: group netid  passwd rpc  services
```

8.8 NIS-Client-Installation

Für den Client müssen Sie die Pakete `ypbind` und `yp-tools` installiert haben, was normalerweise der Fall ist.

Danach stellen Sie im YaST-Kontrollzentrum unter *Netzwerkdienste* • *NIS-Client* einige Parameter ein.

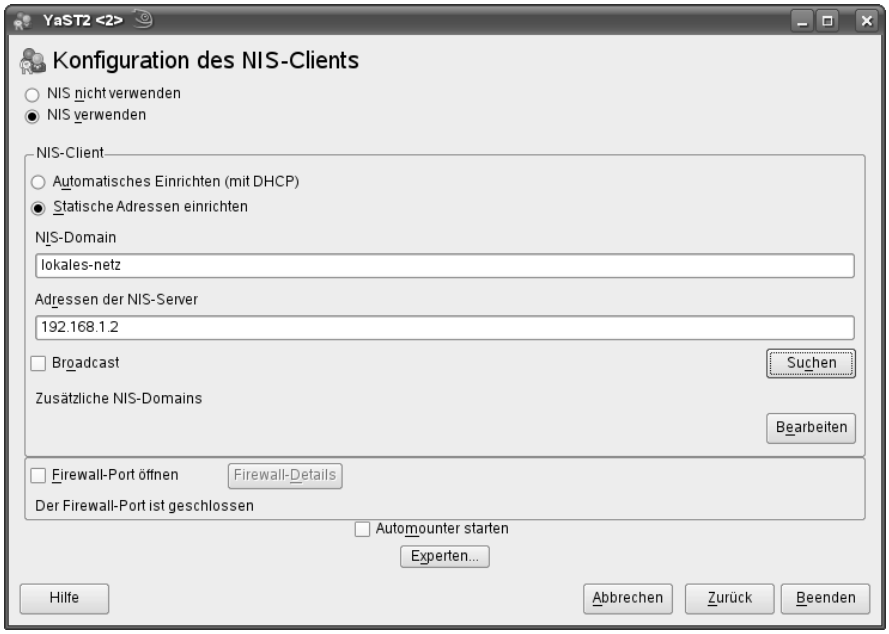

Abbildung 8.7: NIS-Client

Hier müssen Sie den gleichen Domainnamen angeben wie beim NIS-Server sowie die IP-Adresse Ihres NIS-Servers. Sie könnten hier auch den DNS-Namen des Servers eintragen, die IP-Adresse ist aber am sichersten.

Beim Beenden von YaST verändert SUSEconfig auf dem Client die Dateien `/etc/passwd` und `/etc/group`, indem es eine Zeile

```
+::::::
```

an die Datei anhängt. Die Datei `/etc/passwd` sieht dann z. B. folgendermaßen aus (Auszug, Ende der Datei).

```
.....
sshd:x:71:65:SSH daemon:/var/lib/sshd:/bin/false
ntp:x:74:65534:NTP daemon:/var/lib/ntp:/bin/false
nobody:x:65534:65533:nobody:/var/lib/nobody:/bin/bash
debacher:x:1000:100:Uwe Debacher:/home/debacher:/bin/bash
named:x:44:44:Name server daemon:/var/lib/named:/bin/false
+::::::
```

Sie sehen hier in der Beispieldatei die von OpenSUSE vorgegebenen Systembenutzer wie `sshd` und `nobody` sowie einen lokalen Benutzer `debacher`. Die Daten aller weiteren Benutzer bekommt der Rechner über den NIS-Server.

Nach dem Beenden von YaST aktivieren Sie den Client, indem Sie entweder den Rechner rebooten, oder an der Konsole eingeben

```
domainname  lokales-netz
```

Beim Start versucht das NIS-Client-Programm, Kontakt zu einem NIS-Server zu bekommen, und gibt eine entsprechende Meldung aus.

8.9 Die Home-Verzeichnisse

Im Prinzip kann sich nach dem Start des NIS-Servers ein Benutzer auf jedem Rechner anmelden, auf dem der NIS-Client läuft.

Wenn Sie das gleich ausprobieren, werden Sie feststellen, dass ein Login mit falschen Daten abgelehnt wird, Sie aber mit richtigen Eingaben sofort wieder im Anmeldebildschirm landen, da die Benutzer auf dem Client bisher keine Home-Verzeichnisse besitzen. Statt auf jedem Client für jeden Benutzer ein Home-Verzeichnis anzulegen, mounten Sie besser die Home-Verzeichnisse vom Anmeldeserver.

Im einfachsten Fall exportieren Sie auf dem Anmeldeserver das komplette Home-Verzeichnis und mounten dies dann auf den Client-Rechnern entsprechend.

Zum Exportieren müssen Sie auf dem NIS-Server folgende Zeile in Ihre Datei `/etc/exports` aufnehmen, entweder direkt oder über die YaST-Funktion.

```
/home   *.lokales-netz.de(rw,sync)
```

Damit erlauben Sie, dass jeder Rechner aus der Domain `lokales-netz.de` dieses Verzeichnis zum Lesen und Schreiben mounten darf. Falls Sicherheit keine so große Rolle spielt, könnten Sie im einfachsten Fall auch schreiben

```
/home   (rw,sync)
```

Falls Sie höhere Sicherheitsansprüche stellen, können Sie auch gezielt nur einzelnen Rechnern das Mounten erlauben.

```
/home   rosine.lokales-netz.de(rw,sync)
  ➥ zitrone.lokales-netz.de(rw,sync)
```

Damit steht den genannten Client-Rechnern dieses Verzeichnis mit allen darin befindlichen Home-Verzeichnissen zur Verfügung.

Auf den Client-Rechnern können Sie dieses Verzeichnis generell ganz mounten, indem Sie die Datei `/etc/fstab` um eine Zeile erweitern, entweder direkt im Editor oder über das YaST-Kontrollzentrum.

```
192.168.1.2:/home      /home   nfs     defaults 0 0
```

Damit mounten Sie das Verzeichnis `/home` des NIS-Servers in das Verzeichnis `/home` auf dem Client. Da Sie das Verzeichnis des Servers nur in ein leeres Verzeichnis auf dem lokalen Rechner mounten können, dürfen die Home-Verzeichnisse eventueller lokaler

Benutzer nicht in `/home` liegen. Legen Sie für diesen Fall ein Verzeichnis `/localhome` für die Home-Verzeichnisse der lokalen Benutzer an.

Damit sollten sich auch Benutzer, die nur auf dem Server angelegt sind, am Client anmelden und am Client arbeiten können. Viel Spaß bei der Arbeit in der NIS-Domain!

8.10 NIS-Feintuning

Mit den bisherigen Beschreibungen arbeitet das NIS-System bereits einwandfrei. Für die praktische Arbeit und vor allem die System-Sicherheit gibt es aber noch ein paar Optimierungsmöglichkeiten.

8.10.1 Passwortänderungen

Interessant wird es, wenn ein Benutzer beim Arbeiten auf einem Client-Rechner sein zentrales NIS-Passwort ändern möchte. Das dafür übliche Programm `passwd` greift nur auf die lokalen Dateien zu und bricht mit einer Fehlermeldung ab.

Um den Benutzern das Ändern ihres Passworts im gesamten Netzwerk zu ermöglichen, muss ein weiterer Dienst, der Passwortdämon `YPPASSWDD`, gestartet werden. Das zweite `D` im Befehl gibt an, dass es sich um den Dämon handelt. Es ist darauf zu achten, diesen Befehl nicht mit dem `yppasswd` auf dem Client zu verwechseln.

Um diesen Dienst zu aktivieren, starten Sie den Dämon auf dem Server per Hand.

```
rcyppasswdd  start
```

Nun kann ein Benutzer sein Passwort ändern, indem er auf dem Client-Rechner das Programm `yppasswd` aufruft.

Einfacher aktivieren Sie diesen Dienst bei der NIS-Server Konfiguration im YaST-Kontrollzentrum, indem Sie die Checkbox vor *Ändern der Passwörter zulassen* aktivieren.

Wenn Sie ein versehentliches Benutzen des alten Programms `passwd` vermeiden wollen, sollten Sie dieses durch einen Link auf `yppasswd` ersetzen.

```
cd /usr/bin
mv passwd passwd.orig
ln -s yppasswd passwd
```

NIS-Benutzer rufen einfach `passwd` auf, die lokalen Benutzer können dann ihr lokales Passwort immer noch durch einen Aufruf von `passwd.orig` ändern.

8.10.2 Vertrauen in die Benutzer

Durchaus nützliche Tools von NIS-Systemen bergen gewisse Risiken. Mit dem NIS-Programm `ypcat` können Sie bzw. Ihre Benutzer eine *Mapdatei* lesen.

```
ypcat  passwd
```

zeigt Benutzern die komplette Passwort-Datei an. Einen bestimmten Datensatz können Sie dann abrufen.

```
ypcat  passwd  | grep debacher
```

würde also den Datensatz für den Benutzer `debacher` liefern.

Neuere Systeme bieten den Befehl `getent` mit den gleichen Funktionen.

In den Datensätzen tauchen zwar nur die verschlüsselten Passwörter auf, das ist aber trotzdem riskant. Passwortdateien lassen sich mit einer gewissen Wahrscheinlichkeit knacken, indem man ein großes Wörterbuch benutzt, jedes Wort verschlüsselt und dann mit den verschlüsselten Passwörtern vergleicht. Auf nahezu jedem System lässt sich so ein großer Teil der Passwörter knacken. Weitere Informationen zur Passwortsicherheit lesen Sie im Kapitel 3.

Sie sollten allen normalen Benutzern die Zugriffsrechte auf diese Dateien wegnehmen, indem Sie die Dateirechte auf 500 ändern.

```
chmod  500  /usr/bin/ypcat
```

Neben Samba (siehe Kapitel 9) gibt Ihnen NIS die Möglichkeit, auch in größeren und heterogenen Netzen mit nur einem einzigen Anmeldeserver zu arbeiten. Nur auf diesem Server müssen Sie Ihre Benutzerdaten pflegen und verwalten. Dieser Server sollte über genügend Plattenkapazität für die Home-Verzeichnisse der Anwender verfügen.