

8 Network Filesystem einrichten

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

Im weiteren Verlauf dieses Buchs lernen Sie im Kapitel 10, stabile und kostengünstige Linux-Arbeitsplätze statt absturzgefährdeter und teurer Windows-PCs 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 Kapitel 10 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.

NFS-Server und -Clients für Windows-Abarten sind den Autoren nicht als Open Source-Software bekannt. Daher ist NFS hier nur für Linux-Server und Linux-Clients beschrieben. Stabile lizenzpflichtige NFS-Server und -Clients für Windows gibt es z.B. von Hummingbird.

Um NFS im Linux-Umfeld benutzen zu können, muss man den Linux-Server und den Linux-Client ein bisschen 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 in den lokalen Verzeichnisbaum einhängen.

8.1 Wann brauchen Sie 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 auf Samba (siehe Kapitel 9) zurückgreifen, generell ist aber der Zugriff per NFS deutlich stabiler 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 den Client vollständig transparent und funktioniert mit sehr unterschiedlichen Serverstrukturen.

8.2 NFS-Server installieren und konfigurieren

Wie viele andere Distributionen auch, installiert SuSE 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 Betriebssystem-Kern verankert und damit deutlich performanter, setzt aber einen entsprechend kompilierten Kernel voraus. Da SuSE die Standard-Kernel entsprechend konfiguriert hat, installiert sie standardmäßig das Userspace-NFS nicht.

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

Vom Funktionsumfang her sind beide Versionen identisch. Sie können sogar beide Versionen nebeneinander installieren; welche Version Sie dann starten, legen Sie über die entsprechende Variable in der Konfigurationsdatei von YaST fest.

8.2.1 Kernel NFS

Falls Sie auf Ihrem System bisher keinerlei NFS-Server installiert haben, so sollten Sie nun das Paket `knfsd` der Serie `n` installieren oder die Datei `knfsd.rpm` aus dem Verzeichnis `n1` laden.

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

- `CONFIG_NFS_FS` und
- `CONFIG_NFSD`.

8.2.2 User Space NFS

Sollten Sie aus irgendeinem Grund doch das Userspace-NFS nutzen wollen, so müssen Sie das Paket `nfsserv` aus der Serie `n` bzw. die Datei `nfsservev.rpm` aus dem Verzeichnis `n1` nachinstallieren.

Der weitere Teil dieses Kapitels bezieht sich auf das Kernel-NFS, für das keine weiteren Installationen notwendig sind.

8.2.3 Der Portmapper

Um NFS nutzen zu können, benötigt man einen Dämon als Service-Vermittler für Client/Server Dienste, die mit *Remote Procedure Calls* (RPC) arbeiten, den RPC-Portmapper. Bei einem derartigen Dienst kann ein Client über ein zugehöriges Serverprogramm Prozeduren auf dem Server ausführen. Zu jeder der Prozeduren gehört eine eindeutige Programm-Nummer. Der Portmapper ordnet diesen Programmnummern Ports zu. Wenn Sie die aktuelle Zuordnung mit dem Befehl

```
rpcinfo -p
```

abrufen, erhalten Sie eine Tabelle mit folgendem Aufbau:

```
root@boss:~ > rpcinfo -p
  Program Vers  Proto  Port
  100000     2   tcp    111  portmapper
  100000     2   udp    111  portmapper
  100021     1   udp   1024  nlockmgr
  100021     3   udp   1024  nlockmgr
  100024     1   udp    895  status
  100024     1   tcp    897  status
  100005     1   udp    898  mountd
  100005     1   tcp    900  mountd
  100005     2   udp    903  mountd
  100005     2   tcp    905  mountd
  100003     2   udp   2049  nfs
```

In der ersten Spalte dieser Tabelle sehen Sie jeweils die Programmnummern 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

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

Zuerst den Portmapper über das Startscript

```
/sbin/init.d/portmap start
```

und danach den eigentlichen Server mit

```
/sbin/init.t/nfsserver start
```

Das Bootscript aktiviert diese beiden Programme automatisch, wenn in YaST unter *Administration des Systems • Konfigurationsdatei verändern* die folgenden Einstellungen vorhanden sind:

- *START_PORTMAP = yes*
- *NFS_SERVER = yes*
- *USE_KERNEL_NFSD_NUMBER = 4*

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

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

8.3 Verzeichnisse exportieren

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

Damit der Server weiß, welche Verzeichnisse er exportieren soll, braucht man diese nur in die Datei `/etc/exports` einzutragen. 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)
/cdrom (ro)
```

Diese tabellenartige Darstellung in der Form

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

gibt drei Daten an:

- Pfad zum Verzeichnis,
- Rechner, die zugreifen dürfen,
- Optionen.

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

```
/sbin/init.d/nfsserver restart
```

8.3.1 Pfad zum Verzeichnis

Die Angaben des obigen Beispiels exportieren zwei Verzeichnisse, das gesamte Homeverzeichnis mit den Benutzerdaten und das CD-ROM-Laufwerk.

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

8.3.2 Welche Rechner dürfen zugreifen?

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

Auf das Homeverzeichnis 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, die die 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 Werten `192.168.1` lautet, den Zugriff.
4. Sie erlauben allen Rechnern den Zugriff, indem Sie in dieser Spalte keine Angabe machen.

8.3.3 Optionen

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

Die wichtigsten Optionen sind:

<i>Befehl</i>	<i>Erläuterung</i>
rw	<i>Read-Write</i> gibt den Clients Lese- und Schreibrechte für das Verzeichnis.
ro	<i>Read-Only</i> ist die Voreinstellung, bei der Clients nicht in das Verzeichnis hinein schreiben dürfen.
root_squash	Voreinstellung, die privilegierte Zugriffe des Super-Users <i>root</i> unterbindet. <i>Root-</i> Zugriffe führt der Server nur mit den Rechten des Benutzers <i>nobody</i> aus.
no_root_squash	Das Gegenteil zu obiger Option. Der Super-User <i>root</i> 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 <i>nobody</i> aus.
noaccess	Verbietet den Clients den Zugriff auf Unterverzeichnisse; damit kann man einzelne Unterverzeichnisse eines freigegebenen Verzeichnisses sperren.

Tabelle 8.1: 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

```
/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 Netzwerk-Verzeichnisse einbinden

Ein Netzwerkverzeichnis, das auf irgendeinem Rechner freigegeben ist, können Anwender 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 *www.linuxbu.ch*

Um Ihnen das Testen zu erleichtern, haben die Autoren auf *www.linuxbu.ch* ein Verzeichnis 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.
/usr/local/ftp/pub      (ro)
```

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 einbinden, indem Sie als *root* folgenden Befehl eingeben:

```
mount -t nfs www.linuxbu.ch:/usr/local/ftp/pub /mnt
```

Anschließend können Sie 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 *kshowmount* (*k* steht für Kernel-NFS) kann man abfragen, welche Verzeichnisse ein Rechner per NFS anbietet. Dazu gibt man ein:

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

Der Rechner *www.linuxbu.ch* gibt dann Folgendes aus:

```
root@boss:~ > kshowmount -e www.linuxbu.ch
Export list for www.linuxbu.ch:
/usr/local/ftp/pub (everyone)
```

Auf das Verzeichnis */usr/local/ftp/pub* können Sie also von jedem Rechner aus zugreifen.

8.4.2 Der Befehl *mount*

Ein NFS-Klient 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:/usr/local/ftp/pub /mnt
```

und vom Einhängen eines CD-ROM-Laufwerks:

```
mount -t iso9660 /dev/cdrom /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 ein Verzeichnis eines anderen Rechners oder ein CD-ROM-Laufwerk. Zwischen dem Rechnernamen und dem Verzeichnis steht immer ein Doppelpunkt; beim CD-ROM-Laufwerk geben Sie ein Gerät, hier */dev/cdrom* an.

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

Mit dem Parameter *-t* können Sie u.a. die folgenden Dateisysteme angeben:

<i>Typ des Dateisystems</i>	<i>Bedeutung</i>
nfs	Network File System
iso9660	Dateisystem auf CD-ROM
vfat	Windows-Dateisystem
ext2	Linux-Dateisystem
proc	Pseudo-Dateisystem

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 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 in der folgenden Art:

```

/dev/hda7    swap      swap      defaults      0    0
/dev/hda6    /         ext2      defaults      1    1
/dev/hda5    /boot    ext2      defaults      1    2

/dev/hdb     /cdrom   auto      ro,noauto,user,exec 0    0
/dev/fd0     /floppy  auto      noauto,user   0    0
none        /proc    proc      defaults      0    0
# End of YaST-generated fstab lines

```

Die Spalten entsprechen den Parametern des Mount-Befehls.

- In der ersten Spalte steht die Datenquelle bzw. das jeweilige Gerät. Eine Angabe wie `/dev/hda5` bezeichnet die Partition Fünf der ersten IDE Festplatte (siehe Kapitel 2, Festplatten vorbereiten). Das Gerät `/dev/hdb` bezeichnet hier ein IDE CD-ROM-Laufwerk und `/dev/fd0` das erste Diskettenlaufwerk.
- In der zweiten Spalte stehen die 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 `auto`. Bei Einträgen mit diesem Dateityp versucht das System, das Dateisystem zu ermitteln. Das ist bei Wechsel-Datenträgern wie Disketten und CDs sinnvoll. In der vierten Spalte folgen die Optionen, wieder durch Kommata getrennt ohne Leerzeichen. Interessant sind hier die Optionen `noauto` und `user`. Mit der ersten Option 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 Usern, dieses Dateisystem zu mounten. Die Option `exec` erlaubt zusätzlich das Ausführen von Programmen im Dateisystem. In der 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`-Partitionen sollte in der fünften Spalte eine `1` stehen, ansonsten eine `0`. 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. Die 0 gibt an, dass der Dämon das entsprechende Verzeichnis beim Mounten nicht testen soll. 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/hda7      swap    swap    defaults      0 0
/dev/hda6      /       ext2    defaults      1 1
/dev/hda5      /boot   ext2    defaults      1 2

/dev/hdb       /cdrom  auto    ro,noauto,user,exec 0 0
/dev/fd0       /floppy auto    noauto,user   0 0
none          /proc   proc    defaults      0 0
boss.lokales-netz.de:/cdrom
                ↪ /mnt    nfs     ro            0 0
```

8.5 Was tun bei Problemen?

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 Server Root-Rechte besitzen, können Sie den Status der Server-Programme überprüfen.

```
/sbin/init.d/portmap status
```

Sie sollten ein einfaches OK als Antwort erhalten.

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

```
/sbin/init.d/nfsserver status
```

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

Sollte einer der Dienste nicht aktiv sein, so überprüfen Sie die Einstellungen in YaST und starten die Dienste per Hand.

Sollte bis hierher alles korrekt aussehen, so fehlt es an der Freigabe, eventuell haben Sie auch nur vergessen, den NFS-Server nach Änderungen neu zu starten. Ob eine Freigabe auf Ihrem Rechner aktiv ist, können Sie jederzeit testen mit

```
/usr/sbin/kshowmount -e
```

Wollen Sie einen fremden Rechner untersuchen, so hängen Sie wie oben beschreiben den Rechnernamen mit an:

```
/usr/sbin/kshowmount -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.

