

## Tutorials - NFS + NIS

## Tutorials - NFS + NIS

## NFS + NIS

Dieser Artikel wurde in einem Wikiweb System geschrieben, das heißt, Sie können die Artikel jederzeit editieren, wenn Sie einen Fehler gefunden haben, oder etwas hinzufügen wollen.

**Keywords:** nfs nis linux howto zentrale userverwaltung datenspeicherung

**Quelle:** [Online-Tutorial.net](https://www.online-tutorial.net)

Abbildung

## Inhaltsverzeichnis

- [Vorwort](#)
- [Überblick](#)
- [Software](#)
- [Installation der Pakete](#)
- [Serverkonfiguration](#)
- [Starten der Server](#)
- [Beibehalten bestehender Benutzerverzeichnisse](#)
- [Clientkonfiguration](#)

# Linux

- [Schlusswort](#)
- [Der Autor](#)

## Vorwort [Top](#)

Den meisten dürfte der Komfort einer zentralen Benutzer- und Datenverwaltung von größeren Windowsnetzwerken bekannt sein. Nur durch diese beiden Technologien ist es möglich, sich überall mit einem einheitlichen Benutzernamen anmelden zu können und alle seine gespeicherten Daten und Einstellungen vorzufinden.

Dies ist zwar seit langem auch für Linux/Unix verfügbar, doch damit ist häufig längeres Suchen nach Informationen im Internet verbunden.

Dieser Artikel soll einen kurzen Überblick über das Thema geben und dabei helfen, schnell eine einfache Anmeldung mit dazugehörigem Home-Verzeichnis bereitzustellen. Auf Kriterien wie Sicherheit und Performance wird nicht ausdrücklich geachtet.

Als Dienste dienen NFS (Network File System) sowie NIS (Network Information System), auch yellow pages (yp) genannt.

## Überblick [Top](#)

Wir setzen an dieser Stelle voraus, dass sich bereits ein funktionsfähiger Server sowie ein oder mehrere installierte Linux-Clients im Netzwerk befinden.

Zuerst werden die benötigten Pakete installiert, danach folgt das Einrichten des NFS- und NIS-Servers. Wenn diese beiden Dienste soweit funktionieren werden wir beim Client das derzeitige /home-Verzeichnis auf den Server kopieren, dort an die richtige Stelle verschieben, es dann über NFS auf dem Client verfügbar machen und gleichzeitig die User-Tabelle über NIS ins System laden. Nach diesen paar Handgriffen ist das System vollkommen einsatzbereit.

## Software [Top](#)

Zum Einsatz kommen für diese Aufgabe auf Server-Seite der NFS-Kernel-Daemon (entsprechende Kernelunterstützung erforderlich, praktisch immer gegeben) und der NIS-Daemon (auch ypsserv genannt). Zusätzlich wird ein Portmapper benötigt sowie make.

Für den Client werden normalerweise keine zusätzlichen Module für NFS benötigt sondern lediglich der NIS-Client (ypbind) und ebenfalls Portmap.

Auf dem Server lief bei diesem Versuch ein aktuelles Debian Sarge und die Clients waren mit Ubuntu 5.10 preview ausgestattet.

## Installation der Pakete [Top](#)

Die hier beschriebene Installation der Pakete bezieht sich auf Debian-basierende Distributionen. In anderen (z.B. Suse) sollten aber ähnliche Pakete zu finden sein, bei welchen die Installation ähnlich einfach ablaufen sollte. Desweiteren sollte es selbstverständlich sein, dass diese Befehle nur mit Superuserrechten (root) ausgeführt werden können.

### Server:

Auf dem Server genügt der Befehl

### Code:

```
apt-get install nfs-kernel-server nis
```

# Linux

Alle weiteren benötigten Pakete werden automatisch mitinstalliert.

## Client:

Der Client gab sich sogar mit folgender Zeile zufrieden:

## Code:

```
apt-get install nis
```

## Serverkonfiguration [Top](#)

Auf dem Server finden wir im Verzeichnis `/etc/default` die Datei `nis`, welche noch angepasst werden muss. Hierzu ändern wir in der NIS-Konfigurationsdatei folgende Zeilen:

### **/etc/default/nis:**

```
NISSERVER=true  
NISCLIENT=false
```

Jetzt ist es an der Zeit um das Verzeichnis festzulegen, welches später exportiert werden soll. Da wir in diesem Fall jedem Benutzer ein überall verfügbares `/home`-Verzeichnis bieten möchten, wird folgende Zeile in die `export`-Datei eingetragen:

### **/etc/exports:**

```
/home 192.168.0.0/24(rw,async)
```

Das erste Argument gibt das Quellverzeichnis an (hier: `/home`).

Die Adresse `192.168.0.0/24` definiert hierbei, dass nur Clients aus diesem Netz das Verzeichnis einbinden dürfen. Erlaubt sind Hostnamen, IPs, Netze (z.B. wie oben bzw. `192.168.0.0/255.255.255.0` - siehe dazu die `Netmask`-Reference).

Die in Klammern geschriebenen Optionen ermöglichen Lese- und Schreibzugriff sowie asynchronen Datentransfer, welcher erheblich schneller ist als der synchrone.

Für den NIS-Server ist die Datei `/etc/ypserv.conf` zuständig.

### **/etc/ypserv.conf:**

```
* : * : shadow.byname : port  
* : * : passwd.adjunct.byname : port  
* : * : * : none
```

Weiters ist es erforderlich den Domainname für die NIS-Domain in `/etc/defaultdomain` einzutragen.

### **/etc/defaultdomain:**

```
heimnetzwerk
```

# Linux

Nun können noch die benötigten Benutzer auf dem Server mit dem Befehl `adduser` hinzugefügt werden.

Nun sollte der NIS-Server gestartet werden:

## Code:

```
/etc/init.d/nis start
```

Danach muss noch `/usr/lib/yp/ypinit` ausgeführt werden, um die Einstellungen zu übernehmen und die benötigten 'Maps' zu erstellen. Bei Bedarf kann die Datei `/var/yp/Makefile` angepasst werden.

Werden später weitere User hinzugefügt bzw. Einstellungen verändert, muss man `make` in `/var/yp` ausführen, um diese zu übernehmen.

## Starten der Server [Top](#)

Der NFS- sowie der NIS-Server werden durch drei kurze Zeilen aktiviert:

## Code:

```
/etc/init.d/portmap restart
/etc/init.d/nfs restart
(nfs? wir haben doch nfs-kernel-server installiert. Der Befehl lautete bei mir: /etc/init.d/nfs-kernel-server restart)
/etc/init.d/nis restart
```

Da bei den meisten Distributionen die Server bereits nach der Installation durch den Paketmanager gestartet werden, verwenden wir hier den `restart`-Befehl.

Es kann (vor allem bei älteren) Distributionen vorkommen, dass statt `nis` der name `yp` bzw. `yellowpages` verwendet wird.

Ob die Server auch wirklich laufen wird mit folgendem Befehl überprüft:

## Code:

```
rpcinfo -p localhost
```

## Beibehalten bestehender Benutzerverzeichnisse [Top](#)

Um das bereits vorhandene `/home`-Verzeichnis vom Client auf den Server zu bekommen, wird das Verzeichnis des Servers zuerst nach `/mnt` eingebunden.

## mounten von nfs:

```
mount -t nfs -o rsize=8192,wsiz=8192,hard,intr 192.168.0.1:/home /mnt
```

Damit der Client nun das `home`-Verzeichnis des Benutzers auf den Server kopieren kann, wird auf Seite 4 / 7

# Linux

diesem ein Verzeichnis mit Lese- und Schreibrechte für jeden eingerichtet:

## Server:

```
$ mkdir /home/puffer
$ chmod 777 /home/puffer
$ chown nobody:nogroup /home/puffer
```

Jetzt werden alle User-Verzeichnisse vom Client auf den Server kopiert:

## Client:

```
$ cp /home/user1 /mnt/puffer -R
$ cp /home/user2 /mnt/puffer -R
usw.
```

Sind nun alle Verzeichnisse auf den Server kopiert, können sie von dort aus mit root-Rechten an die richtigen Orte verschoben werden.

## Server:

```
$ mv /home/puffer/benutzer1 /home/
$ chown benutzer1:gruppe-benutzer1 /home/benutzer1 -R
```

bzw. bei Problemen:

## Server:

```
$ mv /home/puffer/benutzer1/* /home/benutzer1
$ mv /home/puffer/benutzer1/.*/home/benutzer1
$ chown benutzer1:gruppe-benutzer1 /home/benutzer1 -R
$ rmdir /home/puffer/benutzer1
```

Nach getaner Arbeit wird das NFS-Verzeichnis wieder vom Client mit `umount /mnt` entfernt.

## Clientkonfiguration [Top](#)

Endlich kann damit begonnen werden, die Konfigurationsdateien des Clients anzupassen. Als erstes wird in der Datei `/etc/defaultdomain` die selbe Domain wie auf dem Server eingetragen.

Für den NIS-Client ist die `yp.conf` sowie `nsswitch.conf` zuständig:

### **/etc/nsswitch.conf:**

```
passwd:      compat
group:       compat
shadow:     compat

netgroup:   nis
```

# Linux

Diese Zeilen sind üblicherweise bereits vorhanden, müssen aber in Ausnahmefällen angepasst werden.

## **/etc/yp.conf:**

```
ypserver 192.168.0.1
```

Ist diese Zeile nicht vorhanden, versucht der Client über Broadcast den Server ausfindig zu machen. Erfolgt die DNS/Host-Auflösung über NIS, so darf folglich keine Domain für diesen Eintrag verwendet werden.

Damit das Client-System die User von NIS auch übernimmt, sind noch ein paar Anpassungen erforderlich. Die folgenden Zeilen müssen einfach jeweils als letztes den zugehörigen Dateien hinzugefügt werden.

## **/etc/passwd:**

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

## **/etc/group:**

```
+:::
```

## **/etc/shadow:**

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

## **/etc/gshadow:**

```
+:::
```

Damit der Client das /home-Verzeichnis auch vom Server einbindet, wird die fstab um folgende Zeile ergänzt:

## **/etc/fstab:**

```
192.168.0.1:/home /home nfs rsize=8192,wsiz=8192,hard,intr 0 0
```

Die IP sollte der des NFS-Servers entsprechen, gefolgt von einem Doppelpunkt, nach welchem der [Pfad](#) zum exportierten Verzeichnis angegeben wird.

Das zweite Argument bestimmt den Ort, an welchem das sogenannte 'Share' eingebunden wird.

# Linux

'nfs' ist der Typ des Dateisystem.

Die Optionen 'rsize=8192,wsiz=8192' führen zu einer erheblichen Leistungssteigerung des Datenverkehrs.

Aufgrund von 'hard' wartet das System bei einem Ausfall des Servers, bis dieser wieder erreichbar ist. Danke dem Zusatz 'intr' ist es aber möglich, diesen Vorgang zu unterbrechen.

Befindet sich das /home-Verzeichnis des Client auf einer eigenen Partition, so muss diese Zeile in der fstab mit einer Raute (#) am Zeilenanfang auskommentiert. Ist dies nicht der Fall, so sollte zumindest der Inhalt des lokalen /home's in einen anderen Ordner gesichert werden:

## Code:

```
$ cp /home /home-backup -R
$ rm /home/* -rf
```

Um Probleme zu vermeiden sollte man noch die lokalen Benutzer (nicht aber die System-User!) aus der /etc/passwd, der /etc/shadow sowie die lokalen Benutzer-Gruppen aus der /etc/group entfernen. Diese beginnen üblicherweise mit der UID bzw. GID 1000 und höher.

Abschließend kann das System neu gestartet werden. Die Benutzer müssten nun automatisch über das Netzwerk vom NIS-Server abgefragt und das /home-Verzeichnis vom NFS-Dienst bereitgestellt werden.

## Schlusswort [Top](#)

Zum Abschluss muss noch gesagt werden, dass es sich hierbei keineswegs um eine sichere Lösung handelt. Der Einsatz empfiehlt sich vor allem für linuxbasierende Heimnetzwerke oder Subnetze für kleine Arbeitsgruppen ohne Zugriff von Außen.

Wer eine sichere Umgebung für eine größere Nutzeranzahl benötigt, sollte sich intensiv mit diesem Thema beschäftigen und Authentifizierungsdienste wie Kerberos verwenden.

## Der Autor [Top](#)

Der Autor Julian Golderer <glua\_at\_4-mail.net> hat dieses System für eine kleine Arbeitsgruppe im Produktiveinsatz. Für diesen Einsatzzweck in einem homogenen Linux-Subnet war dies die optimale Lösung.

Sollten Fragen oder Problem auftauchen, können Sie gerne per Mail oder [Jabber](#) (glua@jabber.4-mail.net) Kontakt mit ihm aufnehmen.

Haben Sie einen Fehler gefunden? Dann klicken Sie doch auf Editieren, und beheben den Fehler, keine Angst, Sie können nichts zerstören, das Tutorial kann wiederhergestellt werden

Eindeutige ID: #1006

Verfasser: n/a

Letzte Änderung: 2014-02-08 11:16