

0. INTRODUCTION.

This HOWTO describes:

1. How to setup a NIS client-only system
2. How to specify what resources to use for NIS
3. How to setup a NIS master server
4. Shadow passwords
5. How to setup a NIS slave server

1. HOW TO SETUP A LOCAL NIS CLIENT

- 1.1 Install the netbase, netstd and nis packages
- 1.2 The installation procedure will ask for your NIS domainname. This is just a name which describes the group of systems that use NIS, it is not a hostname. It is stored in the config file **/etc/defaultdomain**, in case you ever want to change it.
- 1.3 If your NIS server is not "local" to your network, you will have to do some finetuning. The ypbind process has a configuration file called **/etc/yp.conf**. You can hardcode a NIS server there - for more info see the manual page for ypbind(8).
- 1.4 Set the **NISSERVER** variable to **false** in **/etc/default/nis** file (if needed).
- 1.5 Specify what resources to use for NIS, as described in paragraph 2 below.
- 1.6 Start NIS client by typing:

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

2. HOW TO SPECIFY WHAT RESOURCES TO USE FOR NIS

2.1 FOR LIBC6:

Check your **/etc/nsswitch.conf** file and make sure that the entries for passwd, group, shadow and netgroup look like this:

```
passwd: compat  
group: compat  
shadow: compat  
netgroup: nis files
```

2.2 USERS: Add the following line to **/etc/passwd** file on your NIS clients:

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

You can also use the + and - characters to include/exclude or change users.

If you want to exclude the user guest just add **-guest** to your **/etc/passwd** file.

You want to use a different shell (e.g. ksh) for the user "linux"?

No problem, just add "+linux::::::/bin/ksh" (without the quotes) to your **/etc/passwd**.

Fields that you don't want to change have to be left empty.

For example, to only allow login-access to miquels,dth and ed, but to have the account data of all other users available:

```
+miquels::::::  
+ed::::::  
+dth::::::  
+:*::::::/etc/NoShell
```

Note that in Linux you can also override the password field, as we did in this example.

Add the following line to **/etc/shadow** file if it is used:

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

2.3 GROUPS: Add the following line to **/etc/group** file:

```
+:::
```

2.4 HOSTS:

Normally you should not do host lookups through NIS, **use DNS for that**.

If you really must use NIS for it, here's how to do it.

2.4.1 For libc5 applications:

If you want to use the NIS host maps of your NIS server you have to change your **/etc/host.conf** file.

You just have to add the word "nis" to the "order" line.

A sample **host.conf** file could look like this:

```
order host,bind,nis  
multi on
```

2.4.2 For libc6 applications:

Edit **/etc/nsswitch.conf** and change the hosts entry:

```
hosts: nis files
```

3. HOW TO SETUP A MASTER NIS SERVER

3.1 Install the nis package. You'll also need the netbase and netstd packages installed for the RPC daemons (rpc.portmap).

3.2 Make sure that *_all_* systems you are going to use as a NIS server, master or slave, have all other systems in the */etc/hosts* file.

The FQDN (Fully Qualified Domain Name) should be the first hostname after every IP address, followed by just the hostname (without the domainname).

For example:

```
192.168.88.10  troi.cistron.nl  troi
```

On the NIS server the NIS hosts file (which is usually */etc/hosts*) should also have this setup. This is because NIS does not use DNS.

3.3 Set your NIS domain in the */etc/defaultdomain* file.

The NIS domain is just a name which describes the group of systems that use NIS, it is not a hostname.

Then type:

```
domainname my_domain_name
```

3.4 Set your NIS server to master (**NISSERVER=master**) in */etc/default/nis* file.

3.5 Add your local network number to the file */etc/ypserv.securenets*.

You don't *_have_* to do this: the default is to allow everybody to access your NIS server. It is however **strongly** recommended.

For added security you might also want to edit the */etc/ypserv.conf* file to mangle the password file for some clients (do not do this if you have non-debian slave servers in your network!)

3.6 Edit the */var/yp/Makefile* file and set the **NOPUSH**, **MINUID**, **MINGID** and **MERGE_PASSWD** variable to appropriate values.

Alternatively, you could edit the */var/yp/Makefile* file and set NIS to use another source password file.

On big systems, the NIS password, shadow and group files are usually stored in */var/yp/ypfiles/*.

If you do this the normal tools to administrate the password file such as "**passwd**", "**chfn**", "**adduser**" will not work anymore and you will need special homemade tools for this.

However **yppasswd**, **ypchsh** and **ypchfn** will work ofcourse.

The Debian version of these utilities - and **yppasswdd** daemon - have a non-standard extension.

"Root" can - using the root password - change other people's passwords, finger info and shell.

3.7 Start the NIS daemon server **ypserv** by typing:

```
ypserv
```

this is necessary elsewhere **ypinit** do not act properly

3.8 Setup the NIS server maps by typing:

```
/usr/lib/yp/ypinit -m
```

3.9 Force the NIS server to update his configuration by typing:

```
/etc/init.d/nis restart
```

This will start the daemon server **ypserv**, the password daemon **rpc.yppasswdd** and the daemon **rpc.ypxfrd** and the client **ypbind**

If you edit the NIS password file manually or you use the standard `/etc/passwd` file, remember that after every change in one of the NIS source files you'll have to run "make" in the `/var/yp` directory to update the NIS maps. It would be reasonable to do this nightly from **cron** to be sure the NIS maps are kept up-to-date.

If you want to restrict access to your NIS server, you'll have to setup the NIS server as a client as well by running **ybind** and adding the + entries to `/etc/passwd` file.

The library functions will ignore all normal entries after the first NIS entry, and will get the rest of the info through NIS. This way the NIS access rules are maintained.

Example:

```
root:x:0:0:root:/root:/bin/bash
daemon:*:1:1:daemon:/usr/sbin:
bin:*:2:2:bin:/bin:
sys:*:3:3:sys:/dev:
sync:*:4:100:sync:/bin:/bin/sync
games:*:5:100:games:/usr/games:
man:*:6:100:man:/var/catman:
lp:*:7:7:lp:/var/spool/lpd:
mail:*:8:8:mail:/var/spool/mail:
news:*:9:9:news:/var/spool/news:
uucp:*:10:50:uucp:/var/spool/uucp:
nobody:*:65534:65534:noone at all,,,,:/dev/null:
+miquels::::::
+:*::::::/etc/NoShell
[ All normal users AFTER this line! ]
tester:*:299:10:Just a test account:/tmp:
miquels:1234567890123:101:10:Miquel van Smoorenburg:/home/miquels:/bin/zsh
```

The user tester will exist, but have a shell of `/etc/NoShell`. miquels will have normal access.

4. SHADOW PASSWORDS

The Linux libc5 does not support shadow NIS maps. If you are dependant on libc5 applications, do not use shadow NIS maps. Instead you can use the method below:

4.1 SHADOW-LIKE SECURITY

You can provide shadow-like security by "mangling" the password for NIS lookups of password-file entries. Read the manpage for "ypserv.conf" and read the comments in the sample `/etc/ypserv.conf`.

4.2 REAL SHADOW SUPPORT

Libc6 has real shadow support for NIS builtin. It works like you would expect; export the shadow map from the NIS server and just use it.

The shadow map should be built with the "-s" (secure) option to `makedbm`.

This is automatic in all modern `/var/yp/Makefile` files.

5. HOW TO SETUP A NIS SLAVE SERVER

5.1 First setup your system as a working NIS client (see 1.)

5.2 Follow the steps 3.1 .. 3.5 as described in above, but at step 3.4 set ypserv to slave (**NISSERVER=slave**) in the **/etc/default/nis** file.

5.3 Start the slave and initialize it by entering:

```
/etc/init.d/nis stop  
/etc/init.d/nis start  
/usr/lib/yp/ypinit -s <name_of_your_master_nis_server_here>
```

5.4 Now tell your master NIS server it has a slave. First you have to adjust the NIS Makefile **on the master** so that all future updates get distributed to the slave(s) automatically. Unset the NOPUSH variable in **/var/yp/Makefile** like this:

```
NOPUSH=""
```

Now tell the master about its slaves by running "**/usr/lib/yp/ypinit -m**".

Enter the names of the slave servers. The maps will get rebuilt and pushed to the slaves.

5.5 You might want to edit root's crontab **on the slave** and add the following lines:

```
20 * * * * /usr/lib/yp/ypxfr_1perhour  
40 6 * * * /usr/lib/yp/ypxfr_1perday  
55 6,18 * * * /usr/lib/yp/ypxfr_2perday
```

This will ensure that most NIS maps are kept up-to-date, even if an update is missed because the slave was down at the time the update was done on the master.