The Linux NIS(YP)/NIS+/NYS HOWTO

Andrea Dell'Amico, Mitchum DSouza, Erwin Embsen

Version 0.3, Last Modified on June 16, 1994

Contents

1 Glossary of Terms **2**

1.1 Some General Information . . . 3

2 Introduction **3**

2.1 New versions of this document . . . 3

2.2 Disclaimer . . . 3

2.3 Feedback . . . 4

2.4 Acknowledgements . . . 4

3 NIS or NIS+ (aka. NYS) ? **4**

4 How it works **4**

5 What you need to set up NIS? **5**

5.1 The RPC Portmapper . . . 5

5.2 Determine whether you are a Server, Slave or Client. . . . 5

5.3 The Software . . . 6

5.4 Setting up a NIS Client . . . 6

6 What you need to set up NYS? **8**

6.1 Determine whether you are a Server, Slave or Client. . . . 8

6.2 The Software . . . 8

6.3 Setting up a NYS Client . . . 9

6.4 The nsswitch.conf File . . . 9

7 Setting up a NYS Server **10**

7.1 The Server Program ypserv . . . 10

7.2 The Program yppasswdd . . . 11

8 Verifying the NIS/NYS Installation **11**

9 Common Problems and Troubleshooting NIS **12**

10 Frequently Asked Questions **12**

1 Glossary of Terms

In this document a lot of acronyms are used. Here are the most important acronyms and a brief explanation:

DBM

DataBase Management, a library of functions which maintain key-content pairs in a data base.

DLL

Dynamically Linked Library, a library linked to an executable program at run-time.

FTP

File Transfer Protocol, a protocol used to transfer files between two computers.

NIS

Network Information Service, a service that provides information, that has to be known throughout the network, to all machines on the network.

nsl

network services library, a library of network service calls (socket, bind, etc...) on SYSV Unixes.

NYS

This is a mutation of the word NIS and represents the NIS+ services library written by Peter Eriksson (pen@lysator.liu.se).

RPC

Remote Procedure Call. RPC routines allow C programs to make procedure calls on other machines across the network.

YP

Yellow Pages(tm), a registered trademark in the UK of British Telecom plc.

TCP-IP

Transmission Control Protocol/Internet Protocol. It's a data communication protocol often used on Unix machines.

1.1 Some General Information

The next three lines are quoted from the Sun(tm) System & Network Administration Manual:

```
"NIS was formerly known as Sun Yellow Pages (YP) but
 the name Yellow Pages(tm) is a registered trademark
 in the United Kingdom of British Telecom plc and may
 not be used without permission."
```

NIS stands for Network Information Service. It's purpose is to provide information, that has to be known throughout the network, to all machines on the network. Information likely to be distributed by NIS is:

- login names/passwords/home directories (/etc/passwd)
- group information (/etc/group)

So, for example, if your password entry is recorded in the NIS passwd database, you will be able to login on all machines on the net which have the NIS client programs running.

Sun is a trademark of Sun Microsystems, Inc. licensed to SunSoft, Inc.

2 Introduction

More and more, Linux machines are installed as part of a network of computers. To simplify network administration, most networks (mostly Sun-based networks) run the Network Information Service. Linux machines can take full advantage of existing NIS service or provide NIS service themselves.

This document tries to answer questions about setting up NIS(YP)/NIS+/NYS on your Linux machine. Don't forget to read section 4.1, The RPC Portmapper.

2.1 New versions of this document

New versions of this document will be posted periodically (about every month) to the newsgroups comp.os.linux.announce and comp.os.linux.misc. The document is archived on a number of Linux FTP sites, including `sunsite.unc.edu` in `/pub/Linux/docs/HOWTO`.

2.2 Disclaimer

Although this document has been put together to the best of our knowledge it may, and probably does contain errors. Please read any README files that are bundled with any of the various pieces of software described in this document for more detailed and accurate information. We will attempt to keep this document as error free as possible.

2.3 Feedback

If you have any comments, questions or suggestions please email them to `erwin@nioz.nl`. Definitely contact him if you find errors or obvious omissions.

2.4 Acknowledgements

We would like to thank all the people who have contributed (directly or indirectly) to this document. In alphabetical order:

```
Andrea Dell'Amico (adellam@di.unipi.it)
Mitchum DSouza    (m.dsouza@mrc-apu.cam.ac.uk)
Erwin Embsen      (erwin@nioz.nl)
Byron A Jeff      (byron@cc.gatech.edu)
```


3 NIS or NIS+ (aka. NYS) ?

The choice between NIS and NIS+ is a question of flexibility vs. maturity. Generally NIS has been around longer and sometimes suffers from its age, inflexibility, and administration difficulties. NIS+ addresses all these problems, provides more flexibility, is easier to set up and administer, and is backward compatable with NIS servers and thus recommended for first time users setting up Linux machines within a NIS enviroment. However, unlike NIS which is integrated into the standard C library, NIS+ requires you to relink all deamons and clients wishing to access these services with the NYS library libnsl.a (similarly to SYSV - or the shared library libnsl.so). However, once clients and deamons are compiled with NYS, they bypass the /etc/host.conf mechanism totally (for host lookups) and determine their resolving mechanism via the /etc/nsswitch.conf file.

4 How it works

Within a network there must be at least one machine acting as a NIS server. If there are more machines in the network providing NIS, then one machine is said to be the master NIS server and all the other NIS servers are so-called slave NIS servers. Slave servers only have copies of the NIS databases and receive these copies from the master NIS server whenever changes are made to the master's databases. Depending on the number of machines in your network and the reliability of your network, you might decide to install one or more slave servers. Whenever a NIS server goes down or is too slow in responding to requests, a NIS client connected to that server will try to find one that is up or quicker.

With NYS, however, the /etc/nsswitch.conf file determines which host you want to go to for a particular service (as there is no binding deamon 'ypbind') if your primary NIS server is not running or is too slow to respond. This adds more flexibility as you are able to choose on a per-service basis which host should answer what query. For example, you may specify that if the master server 'A' is down then, for host queries, go to server 'B' and for password queries go to server 'C' etc..

NIS databases are in so-called DBM format, derived from ASCII databases. For example, the files /etc/passwd and /etc/group can be directly converted to DBM format using ASCII-to-DBM translation software ('dbload', it's included with the server software). The master NIS server should have both, the ASCII databases and the DBM databases. Slave servers or client machines will be notified of any change to the NIS maps, (via the "ypbind" deamon), and automatically retrieve the necessary changes in order to synchronize their databases. The author of the YP clients for linux has informed us that the newest ypbind (from yp-clients.tar.gz) is able to get the server from a configuration file - thus not need to broadcast.

5 What you need to set up NIS?

5.1 The RPC Portmapper

To run any of the software mentioned below you will need to run the program /usr/sbin/rpc.portmap. Some Linux distributions already have the code in /etc/rc.d/rc.inet2 to start up this daemon. All you have to do is comment it out and reboot your Linux machine to activate it.

The RPC portmapper (portmap(8c)) is a server that converts TCP/IP protocol port numbers into RPC program numbers. It must be running in order to make RPC calls (which is what the NIS client software does). When an RPC server is started (like a NIS server program), it will tell portmap what port number it is listening to, and what RPC program numbers it is prepared to serve. When a client wishes to make an RPC call to a given program number, it will first contact portmap on the server machine to determine the port number where RPC packets should be sent.

Normally, standard RPC servers are started by inetd(8C), so portmap must be started before inetd is invoked.

5.2 Determine whether you are a Server, Slave or Client.

To answer this question you have to consider two cases:

1. Your machine is going to be part of a network with existing NIS servers
2. You do not have any NIS servers in the network yet

In the first case, you only need the client programs (ypbind, ypwhich, ypcat, yppoll, ypmatch). The most important program is ypbind. This program must be running at all times, that is, it should always appear in the list of processes. It's a so-called daemon process and needs to be started from the system's startup file (eg. /etc/rc.local). As soon as ypbind is running, your system has become a NIS client.

In the second case, if you don't have NIS servers, then you will also need a NIS server program (usually called ypserv). Section 6 describes how to set up a NIS server on your Linux machine using the 'ypserv' implementation by Peter Eriksson (`pen@lysator.liu.se`). Note that this implementation does NOT support the master-slave concept talked about in section 3. Using this software, all your NIS servers will be master servers.

5.3 The Software

The system library "/usr/lib/libc.a" (version 4.4.2 and better) or the shared library "/usr/lib/libc.sa" and its related DLL contain all necessary system calls to succesfully compile the NIS client and server software. Some people reported that NIS only works with "/usr/lib/libc.a" version 4.5.21 and better so if you want to play it safe don't user older libc's. The NIS client software can be obtained from:

```
Site                  Directory                     File Name

ftp.uni-paderborn.de  /pcsoft2/linux/local/yp       yp-clients.tar.gz
nic.funet.fi          /pub/OS/Linux/BETA/NYS        yp-clients.tar.gz
sunsite.unc.edu       /pub/Linux/system/Network/admin yp-clients.tar.gz
```

Once you obtained the software, please follow the instructions which come with the software.

5.4 Setting up a NIS Client

Assuming you have succesfully compiled the software you are now ready to install the software. A suitable place for the ypbind daemon is the directory /usr/sbin.

You'll need to do this as root of course. The other binaries (ypwhich, ypcat, yppoll, ypmatch) should go in a directory accessible by all users, for example /usr/etc or /usr/local/bin. It might be a good idea to test ypbind before incorporating it in /etc/rc.d/rc.inet2.

To test ypbind do the following:

- Make sure you have your domain name set. If it is not set then issue the command:

  ```
  /bin/domainname my.domain
  ```

  where `my.domain` is the domain your machine is part of.

- Start up "/usr/sbin/rpc.portmap" if it is not already running.
- Create the directory "/var/yp" if it does not exist.
- Start up "/usr/sbin/ypbind"
- Use the command "rpcinfo -p localhost" to check if ypbind was able to register its service with the portmapper. The rpcinfo should produce something like:

  ```
  program vers proto   port
   100000    2   tcp    111  portmapper
   100000    2   udp    111  portmapper
   100007    2   udp    637  ypbind
   100007    2   tcp    639  ypbind
   300019    1   udp    660
  ```

- You may also run "rpcinfo -u localhost ypbind". This command should produce something like:

```
program 100007 version 2 ready and waiting
```

Finally, do not forget that for host lookups you must set (or add) "nis" to the lookup order line in your /etc/host.conf file. Please read the manpage 'resolv+.8' for more details.

At this point you should be able to use NIS client programs like ypcat, etc... For example, "ypcat passwd" will give you the entire NIS password database.

IMPORTANT: If you skipped the test procedure then make sure you have set the domain name, and created the directory:

```
/var/yp
```

This directory MUST exist for ypbind to start up succesfully.

If the test worked you may now want to change the files /etc/rc.d/rc.M and /etc/rc.d/rc.inet2 on your system so that ypbind will be started up at boot time and your system will act as a NIS client. Edit the file /etc/rc.d/rc.M and look for the commands which set the domain name. Change the domain name into the name of your domain. Also, edit the file /etc/rc.d/rc.inet2, comment out the lines which start up the rpc.portmap daemon, and add the following lines just after the place where rpc.portmap is started:

```
#
# Start the ypbind daemon
#
if [ -f ${NET}/ypbind -a -d /var/yp ]; then
    echo -n " ypbind"
    ${NET}/ypbind
fi
```

Unlike Sun's implementation of NIS you do not need to edit /etc/passwd and /etc/group to take advantage of NIS. Sun's implementation needs a line "+:*:0:0:::" in /etc/passwd and a line "+:*:0:" in /etc/group to tell NIS to search the NIS password and group databases.

Well, that's it. Reboot the machine and watch the boot messages to see if ypbind is actually started.

IMPORTANT: Note that the netgroup feature is not yet implemented. Netgroups allow access control for every machine and every user in the NIS domain, and they require an entry like:

```
+@this_machine_users
```

in /etc/passwd. At this moment, every user in the NIS password database can access your linux machine if you run "ypbind".

6 What you need to set up NYS?

6.1 Determine whether you are a Server, Slave or Client.

To answer this question you have to consider two cases:

- Your machine is going to be part of a network with existing NIS servers
- You do not have any NIS servers in the network yet

In the first case, you only need to relink all client and deamon programs with the NYS library libnsl.so (or statically link them with libnsl.a). This means adding the line

```
LIBS=-lnsl
```

to your Makefile signifing you want to link the Network Services Library.

Basically all network deamons and the 'login' program need to be recompiled. However an alternate strategy is now available. See section 5.5 below on how to make your binaries NYS aware.

Similarly like in the case of the standard NIS, if you don't have NIS servers, then you will also need a NIS server program (usually called ypserv) and you have to designate one of the machines in your network as a master NIS server. Again, you might want to set up at least one slave server as well (Note: not applicable to ypserv on Linux).

6.2 The Software

You need to retrieve and compile the NYS services library libnsl.so. If you don't have the DLL tools installed you may retrieve a precompiled shared, static and stub library from the same site mentioned below.

The NYS library (source and precompiled version) can be obtained from:

```
Site            Directory                   File Name

nic.funet.fi    /pub/OS/Linux/BETA/NYS      nys-0.27.1.tar.gz
nic.funet.fi    /pub/OS/Linux/BETA/NYS/lib  libnsl.so.1.0.a26
```

Precompiled 'login' and 'su' programs may also be fetched from

```
Site            Directory                   File Name

nic.funet.fi    /pub/OS/Linux/BETA/NYS/bin  login
nic.funet.fi    /pub/OS/Linux/BETA/NYS/bin  su
```

Similarly, example configuration files may be retrieved from

Site	Directory	File Name
nic.funet.fi	/pub/OS/Linux/BETA/NYS/etc	*conf

For compilation of the nsl library, please follow the instructions which come with the software. If you wish to compile the shared DLL library you must have the DLL tools installed in the standard place (/usr/dll). The DLL tools (the package tools-2.11.tar.gz) can be obtained from many sites.

6.3 Setting up a NYS Client

Unlike NIS, there is no setting up required for a NYS client. All that is required is that the configuration files (/etc/yp.conf, /etc/nis.conf) point to the correct servers for their information. See the examples provided with the source code.

6.4 The nsswitch.conf File

The Network Services switch file /etc/nsswitch.conf determines the order of lookups performed when a certain piece of information is requested, just like the /etc/host.conf file which determines the way host lookups are performed. Again, look at at the example file provided in the source distribution. For example, the line

```
hosts: files nis dns
```

specifies that host lookup functions should first look in the local /etc/hosts file, followed by a NIS lookup and finally thru the domain name service (/etc/resolv.conf and named), at which point if no match is found an error is returned.

5.5. Making your binaries NYS aware

Instead of relinking each binary with the NYS library (libnsl.so), a cleaner solution has been achieved by providing the user with the ability to build a NYS aware libc. This means all you need to do is recompile a new libc and replace your existing /lib/libc.so.x.y.z for all (non-static compiled) programs to be NYS aware.

This merge also gives you the advantage over the "standard" YP implementation in the linux libc in that it allows transparent shadow passwords support (via the /etc/nisswitch.conf file).

Follow these simple steps to rebuild a NYS aware libc.

- Make sure you have the latest DLL tools installed. Refer to the the GCC-FAQ for more info on where to get this.
- Get the latest libc sources. (again see GCC-FAQ)
- Get the latest nys sources from

  ```
  ftp.lysator.liu.se:/pub/NYS
  ```

and extract it under this libc-linux source directory. The current NYS distribution is "nys-0.27.1.tar.gz".

- Do the ./configure as before and first answer 'n' to the question

```
Values correct (y/n) [y] ?
```

Then go thru all the other questions and the last question will now be

```
Build a NYS libc from nys-0.27 (y default) ?
```

answer 'y' to this.

- Then issue the command

```
% make
```

The library generated after compilation is named something like

```
libc.so.4.5.26
```

and placed under the directory jump/libc-nys. To install this library I would avdise copying it to /lib with a name lexiographically greater than the version number it currently has. Just appending the letter "a" should do the trick. For example:

```
% cp jump/libc-nys/libc.so.4.5.26 /lib/libc.so.4.5.26a
```

Alternatively I append "nys" to it so I can quickly identify it. Now run the command

```
% ldconfig
```

which will reset your cache to use the new library. The dynamic linker strategy may be examined with the command 'ldconfig -p'.

That's basically it. All your programs should now be NYS aware. Please note that usually the program "login" is compiled static and thus cannot access the new NYS functions from the NYS aware libc. You must either recompile "login" without the -static flag, or else statically link it to the libnsl.a library.

7 Setting up a NYS Server

7.1 The Server Program ypserv

The NIS/NYS server software can be found on:

```
Site                 Directory                          File Name

ftp.lysator.liu.se   /pub/NYS                           ypserv-0.11.tar.gz
ftp.univie.ac.at     /unix/system/linux/funet/BETA/NYS  ypserv-0.11.tar.gz
mcsun.eu.net         /os/linux/BETA/NYS                 ypserv-0.11.tar.gz
```

The server setup is the same for both NIS and NYS.

Compile the software to generate the 'ypserv', 'dbcat' and 'dbload' programs. Firstly, determine what files you require to be available via NIS/NYS and then add or remove the appropriate entries to the ypMakefile. Now build the DBM files by

```
% make -f ypMakefile
```

Make sure the portmapper (rpc.portmap) is running, and start the server 'ypserv'. The command

```
% rpcinfo -u localhost ypserv
```

should output something like

```
program 100004 version 2 ready and waiting
```

That's it, your server is up and running.

7.2 The Program yppasswdd

Whenever users change their passwords, the NIS password database and probably other NIS databases, which depend on the NIS password database, should be updated. The program "yppasswdd" is a server that handles password changes and makes sure that the NIS information will be updated accordingly. The software for "yppasswdd" can be found on:

```
Site                  Directory       File Name

ftp.lysator.liu.se    /pub/NYS        yppasswdd-0.1.tar.gz
```

Once you obtained the software, please follow the instructions which come with the software.

8 Verifying the NIS/NYS Installation

If everything is fine (as it should be), you should be able to verify your installation with a few simple commands. Assuming, for example, your passwd file is being supplied by NIS/NYS, the command

```
% ypcat passwd
```

should give you the contents of your NIS passwd file. The command

```
% ypmatch userid passwd
```

(where userid is the login name of an arbitrary user) should give you the user's entry in the NIS passwd file. The 'ypcat' and 'ypmatch' programs should be included with your distribution of NIS or NYS.

9 Common Problems and Troubleshooting NIS

Here are some common problems reported by various users:

1. The libraries for 4.5.19 are broken. NIS won't work with it.
2. If you upgrade the libraries from 4.5.19 to 4.5.24 then the su command breaks. You need to get the su command from the slackware 1.2.0 distribution. Incidentally that's where you can get the updated libraries.
3. You could run into trouble with NIS and DNS on the same machine. My DNS server occasionally will not bring up NIS. Haven't yet tracked down why.
4. When a NIS server goes down and comes up again ypbind starts complaining with messages like:

   ```
   yp_match: clnt_call:
                   RPC: Unable to receive; errno = Connection refused
   ```

   and logins are refused for those who are registered in the NIS database. Try to login as root and if you succeed, then kill ypbind and start it up again.

10 Frequently Asked Questions

Most of your questions should be answered by now. If there are still questions unanswered you might want to post a message to

```
comp.os.linux.help
```

or contact one of the authors of this HOWTO.