SettingUpNFSHowTo

Sommaire

NFS Server

Pre-Installation Setup

None of the following pre-installation steps are strictly necessary.

User Permissions

NFS user permissions are based on user ID (UID). UIDs of any users on the client must match those on the server in order for the users to have access. The typical ways of doing this are:

- Manual password file synchronization
- Use of LDAP
- Use of NIS

It's also important to note that you have to be careful on systems where the main user has root access - that user can change UID's on the system to allow themselved access to anyone's files. This page assumes that the administrative team is the only group with root access and that they are all trusted. Anything else represents a more advanced configuration, and will not be addressed here.

Group Permissions

With NFS, a user's access to files is determined by his/her membership of groups on the client, not on the server. However, there is an important limitation: a maximum of 16 groups are passed from the client to the server, and, if a user is member of more than 16 groups on the client,

some files or directories might be unexpectedly inaccessible.

Host Names

optional if using DNS

Add any client name and IP addresses to /etc/hosts. The *real* (not 127.0.0.1) IP address of the server should already be here. This ensures that NFS will still work even if DNS goes down. You could rely on DNS if you wanted, it's up to you.

NIS

optional - perform steps only if using NIS

Note: This **only** works if using NIS. Otherwise, you can't use netgroups, and should specify individual IP's or hostnames in `/etc/exports`. Read the **BUGS** section in `man netgroup`.

Edit /etc/netgroup and add a line to classify your clients. (This step is not necessary, but is for convenience).

```
[illegible]
```

Obviously, more clients can be added. `myclients` can be anything you like; this is a *netgroup name*.

Run this command to rebuild the YP database:

```
[illegible]
```

Portmap Lockdown

optional

Add the following line to /etc/hosts.deny:

```
[illegible]
```

By blocking all clients first, only clients in /etc/hosts.allow below will be allowed to access the server.

Now add the following line to /etc/hosts.allow:

```
SRUWP DS P RXQWG QIVG VWDWG ORFNG UTXRWDG OLVW RI,3 DGGUHVVHV
```

Where the "list of IP addresses" string is, you need to make a list of IP addresses that consists of the server and all clients. These have to be IP addresses because of a limitation in portmap (it doesn't like hostnames). Note that if you have NIS set up, just add these to the same line.

Installation and Configuration

Install NFS Server

```
VXGR DSWJHW LQVWDOO SRUWPDS QIVNHUQHOVHUYHU
```

Shares

Edit /etc/exports and add the shares:

```
KRPH  P\FOLHQWV UZ V\QF QRBVXEWUHHBFKHFN
KVU ORFDO  P\FOLHQWV UZ V\QF QRBVXEWUHHBFKHFN
```

The above shares /home and /usr/local to all clients in the myclients netgroup.

```
KRPH  UZ V\QF QRBVXEWUHHBFKHFN  UZ V\QF QRBVXEWUHHBFKHFN
KVU ORFDO  UZ V\QF QRBVXEWUHHBFKHFN
```

The above shares /home and /usr/local to two clients with fixed ip addresses. Best used only with machines that have static ip addresses.

```
KRPH  UZ V\QF QRBVXEWUHHBFKHFN
KVU ORFDO  UZ V\QF QRBVXEWUHHBFKHFN
```

The above shares /home and /usr/local to all clients in the private network falling within the designated ip address range.

`rw` makes the share read/write, and `sync` requires the server to only reply to requests once any changes have been flushed to disk. This is the safest option (`async` is faster, but dangerous. It is strongly recommended that you read `man exports`.

After setting up /etc/exports, export the shares:

```
VXGR H[SRUWIV UD
```

You'll want to do this command whenever `/etc/exports` is modified.

Restart Services

If /etc/default/portmap was changed, portmap will need to be restarted:

```
/etc/init.d/portmap restart
```

The NFS kernel server will also require a restart:

```
/etc/init.d/nfs-kernel-server restart
```

Security Note

Aside from the UID issues discussed above, it should be noted that an attacker could potentially masquerade as a machine that is allowed to map the share, which allows them to create arbitrary UIDs to access your files. One potential solution to this is IPSec, see also the NFS and IPSec section below. You can set up all your domain members to talk only to each other over IPSec, which will effectively authenticate that your client is who it says it is.

IPSec works by encrypting traffic to the server with the server's key, and the server sends back all replies encrypted with the client's key. The traffic is decrypted with the respective keys. If the client doesn't have the keys that the client is supposed to have, it can't send or receive data.

An alternative to IPSec is physically separate networks. This requires a separate network switch and separate ethernet cards, and physical security of that network.

NFS Client

Installation

```
sudo apt-get install portmap nfs-common
```

Portmap Lockdown

optional

Add the following line to /etc/hosts.deny:

```
portmap: ALL
```

By blocking all clients first, only clients in /etc/hosts.allow below will be allowed to access

the server.

Now add the following line to /etc/hosts.allow:

```
portmap: 192.168.0.3
```

Where "NFS server IP address" is the IP address of the server. **This must be numeric!** It's the way portmap works.

Host Names

optional if using DNS

Add the server name to /etc/hosts. This ensures the NFS mounts will still work even if DNS goes down. You could rely on DNS if you wanted, it's up to you.

Mounts

Check to see if everything works

You should try and mount it now. The basic template you will use is:

```
mount -t nfs -o options servername:/path/to/share /local/path
```

so for example:

```
mount -t nfs -o proto=tcp,port=2049 192.168.0.2:/mnt/sharename /home/username/sharename
```

Mount at startup

NFS mounts can either be automatically mounted when accessed using autofs or can be setup with static mounts using entries in /etc/fstab.

Automounter

Install autofs:

```
sudo apt-get install autofs
```

The following configuration example sets up home directories to automount off an NFS server upon logging in. Other directories can be setup to automount upon access as well.

Add the following line to the end of /etc/auto.master:

```
[illegible]
```

Now create /etc/auto.home and insert the following:

```
[illegible]
```

The above line automatically mounts any directory accessed at /home/[username] on the client machine from either solarisbox1.company.com.au:/export/home/[username] or solarisbox2.company.com.au:/export/home/[username].

Restart autofs to enable the configuration:

```
[illegible]
```

Static Mounts

Prior to setting up the mounts, make sure the directories that will act as mountpoints are already created.

In /etc/fstab, add lines for shares such as:

```
[illegible]
```

The `rw` mounts it read/write. Obviously, if the server is sharing it read only, the client won't be able to mount it as anything more than that. The `hard` mounts the share such that if the server becomes unavailable, the program will wait until it is available. The alternative is `soft`. `intr` allows you to interrupt/kill the process. Otherwise, it will ignore you. Documentation for these can be found in the `Mount options for nfs` section of `man mount`.

The filesystems can now be mounted with `mount /mountpoint`, or `mount -a` to mount everything that should be mounted at boot.

Notes

Minimalistic NFS Set Up

The steps above are very comprehensive. The minimum number of steps required to set up NFS are listed here:

http://czarism.com/easy-peasy-ubuntu-linux-nfs-file-sharing

Using Groups with NFS Shares

When using groups on NFS shares (NFSv2 or NFSv3), keep in mind that this might not work if a user is a member of more than 16 groups. This is due to limitations in the NFS protocol. You can find more information on Launchpad ("Permission denied when user belongs to group that owns group writable or setgid directories mounted via nfs") and in this article: "What's the deal on the 16 group id limitation in NFS?"

IPSec Notes

If you're using IPSec, the default shutdown order in Breezy/Dapper causes the client to hang as it's being shut down because IPSec goes down before NFS does. To fix it, do:

```
[illegible]
[illegible]
```

A bug has been filed here: https://launchpad.net/distros/ubuntu/+source/ipsec-tools/+bug/37536

Credits

- MatthewCaron - NFS Server, NFS Client, IPSec Notes
- NaamanCampbell - NFS Client - Automount

CategoryDocumentation

SettingUpNFSHowTo (dernière édition le 2008-06-27 18:30:48 par https://login.launchpad.net/+id/BBb7MRb @ cpe-76-171-7-75.socal.res.rr.com[76.171.7.75]:rocket2dmn)