Setting up the NFS Maestro Server under Windows NT

Here is a detailed description of how to setup the NFS Server from scratch. If you require any further explanation of this procedure, please do not hesitate to call Hummingbird directly.

THE EXPORTS EDITOR SECTION

Right after you install the NFS Server as "Administrator", the first thing you want to do is go to the Exports Editor. Here is where you will ADD a directory that will be exported to your NFS Client(s). On Windows NT, there are two different types of Filesystems, FAT and NTFS. NTFS is a more secure Filesystem that has the concept of ownership and groups. The other filesystem (FAT) does not have a concept of security.. thus allowing anyone access to the files.

In the export editor, click on the PLUS icon in the tool bar. You will be prompted with an "Exported Filesystem Maintenance" Box. There are two fields present in this box. The first field is used to enter the Filesystem to be exported. There are two ways to do this.

D:\NFS\Work\Users (dos style)
/d/NFS/WORK/USERS (UNIX style)

** **IMPORTANT NOTE**: Please keep in mind that the Export Editor checks for case sensitive filenames. Don't use the Windows Explorer to check for case sensitivity, as it usually only capitalizes the first letter of the filename regardless of it's "true" case. To check for the true case of the directory, go to the MS-DOS prompt and use the DIR command.

In the second field of the "Exported Filesystem Maintenance" box, is where you specify the parameters. There are five options that can be used to specify how you would like to restrict access to the filesystem. Access can be restricted on a "per host" basis, not a "per user" basis. For example, we will use the host "Fred" & "Barney" which are UNIX based operating Systems. Here are the 5 options that you can use:

rw	gives read/write access to the clients listed and "read only" to everyone else
ro	gives read only access to the clients listed
access	same as rw, but denies access to everyone else
root	needed to give "root" user access to the filesystem
lower	export the filesystem forcing all lowercase filenames

You can mix these options, for more than one host. The initial option must ALWAYS be preceded by a dash (-). Multiple hosts should be separated by a colon (:), and multiple options should be separated by commas.

WARNING!! Do not put spaces in the parameters!!

Here are a few examples of parameter entries:

-rw=barney,access=fred,root=fred:barney
-access=barney,root=barney
-ro=fred,lower
-lower
-rw
-ro

NOTE: You may want to put the Host name and the Fully Qualified Domain Name as well..

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

(Ex. -rw=barney:barney.domain.com,access=fred:fred.domain.com)

REASON: The Server resolves the client name according to what it receives from the client. Sometimes the client will send over the Fully Qualified Domain name instead of just "barney".

Question: How Do I Know what the server sends?

Answer: The way to find out if the server recognizes the client is to export the filesystem as -access to only that machine name (barney). As you know, the "access" option gives the client read/wrtie ability and gives NO ACCESS to any other host. Let's say that the host Barney tries to mount the server (with the -access=barney), and gets ACCESS DENIED..

Even though you specified -Access=Barney, the client maybe sending over "Barney.Domain.Com". The server doesn't believe that "Barney" and "Barney.Domain.Com" are the same hosts.

BASIC RULE: If the server doesn't match the hosts exactly... ACCESS DENIED.

NOTE: If you encounter this problem and the fully qualified domain name does not resolve the issue, please call Hummingbird directly.

RELOADING THE EXPORTS

Every time you make a change to the exports editor, you ALWAYS have to reload the exports to be available to the NFS Clients.

1. Goto Control Panel
2. Double Click on "NFS Server"
3. Click on "Server Status" button
4. Cleck on "Exports" button
5. Click on "Reload" button

If everything is correct, you should be able to see the exported directory in the list.

IMPORTANT NOTE: The root & lower options are NEVER displayed in the exports.

NOTE: If you don't see the exported directory, the case of the directory you specified in the Exports Editor is most likely incorrect. (Check MSDOS to verify)

MOUNTING THE FILESYSTEM

Now that the filesystem is exported from the NFS Server, you should be able to mount the directory on your NFS Client. The mount command varies depending on the operating system. If the NFS Client is UNIX based, you should have a filesystem that will become the "mount point". It shouldn't contain any important files (as it becomes unavailable when mounted). It should be preferably empty. If you're not sure where to mount, create a temporary directory to be the mount point. For this example, lets say the directory is /ntfs/mount on your UNIX Client. To mount the filesystem, you need to be logged in as root. Only root has the ability to mount and unmount filesystems.

For this example

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

NT machine name Hawk
Exported Directory D:\NFS\Work\Users
Mount Point /ntfs/mount

Typical mount commands are:

mount hawk:/d/NFS/Work/Users /ntfs/mount
mount -F nfs hawk:/d/NFS/Work/Users /ntfs/mount
mount -T nfs hawk:/d/NFS/Work/Users /ntfs/mount

NOTE: To see more options and the format of the mount command specific to your machine, type the following command:

man mount

If the mount command is sucessful, it will return back to the prompt.

Goto the mount point by typing:

cd /ntfs/mount

CHECKING THE PERNISSIONS

NOTE: THE REST OF THIS DOCUMENT ONLY APPLIES TO FILESYSTEMS ON AN NTFS EXPORTED FILESYSTEM ONLY. IF YOU ARE MOUNTING A FAT FILESYSTEM, THERE IS NO SECURITY ON THE FILES..

cd /ntfs/mount

Once in the mount point, check the permissions on the files by typing:

ls -la

You will see a listing similar to this one:

```
d---------   2 65533     65533        4096  May 28 14:47 .
drwxr-xr-x  210 bin       bin         8192  May 25 12:50 ..
----------   1 65533     65533         395  Jun 10  1997 apple
----------   1 65533     65533         395  Jun 10  1997 orange
----------   1 65533     65533         395  Jun 10  1997 grapes
----------   1 65533     65533         395  Jun 10  1997 pears
----------   1 65533     65533         395  Jun 10  1997 fruits
----------   1 65533     65533         395  Jun 10  1997 peaches
----------   1 65533     65533         395  Jun 10  1997 plums
----------   1 65533     65533         395  Jun 10  1997 tangerine
```

Before we go any further, you will need to understand a slight concept between the two different operating systems!!

NT uses the concept of Serurity Identifiers associated with it's files and directories. Along with these SID's are the Access Control List (ACL) and Access Control Entries (ACE). On UNIX, ownership is defined as

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

a specific number (like SID's) called UID's.. and groups are associated to a specific group number called a GID.

Because UNIX and NT are quite different in the way they store ownership information, that is why the username and group in the above listing are "65533". This is because UNIX expects to see a UID & GID attached to the file, and it doesn't understand NTFS security. So, it replaces the files ownership with 65533, a NOBODY account. This will cause permission problems, as NOBODY has limited access.

HOW TO RESOLVE 65533 (NOBODY) ENTRIES

If you noticed at the top of the directory, there is a dot (.) and dot-dot (..) directory. The dot (.) directory is the current directory (which represents the exported directory)

```
d---------   2 65533     65533        4096  May 28 14:47 .
drwxr-xr-x  210 bin       bin          8192  May 25 12:50 ..
```

We want to associate the user root as the owner of this mountpoint. We need to know the UID of the root user account. From the command line, type the following:

```
% id root
```

You will see something like this:

```
uid=0(root) gid=3(sys) groups=10(staff)
```

We need to associate the UID of zero (0) to the NT User who owns the exported directory. So, when the UNIX machine checks to see who the owner is, the NT machine will convert the NT Username to UID 0 (which UNIX will then understand).

Question: How do you know who the User and Group are of the NT directory or file?

Answer: SHOWPROT.EXE

NFS Maestro Server comes with an MSDOS command line utility called SHOWPROT which is located by default in your Program Files directory, under the nfssrvr.nt sub-folder. The Syntax for the command is as follows:

```
showprot D:\NFS\Work\Users | more
```

NOTE: Sometimes the listing is longer than a page.. that is why it is piped to more.

This command can be used on files or directories to see the security. It can only be run by an administrative user. The output is similar to this:

```
Full lookup of file d:\nfs\work\users (#links = 1)
Owner - BUILTIN\Administrators (an alias)
Group - BUILTIN\Administrators (an alias)
DACL - \Everyone (an Access-Allowed-Ace-Type)
  This ACE is inherited by contained non-directories
  This ACE only applies to contained files, not to the file itself
  Access_Mask = 0x10000000
  Generic-All set
Equals first DACL
```

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

DACL - \Everyone (an Access-Allowed-Ace-Type)
This ACE has no flags.
Access_Mask = 0x001f01ff
Access = File-All-Access

The two important lines to be concerned with are:

Owner - BUILTIN\Administrators (an alias)
Group - BUILTIN\Administrators (an alias)

NOTE: Notice how it shows "Administrators" (plural, not singular)

According to NT, the owner of this directory is Administrators, but it can be any valid NT Username. In this example, the group is associated with Administrators.

THE ACTUAL MAPPING OF AN NT ACCOUNT TO A UID

Goto the control panel, and double click on NFS Server. This is your UID/GID Mappings screen. Type in "Administrators" (remember plural, not singular) under NT Username. Put in the associated UID of zero (0) into the UID field directly under it To make it valid. Click on "ADD" (it will move over to the right). To Save this and make the changes permanent, click on the Save button

The moment you click on SAVE, go back to the UNIX Client and do another "ls -la" in the mount point and you will notice a change!!

```
d---------   2 root      65533        4096  May 28 14:47 .
drwxr-xr-x  210 bin       bin          8192  May 25 12:50 ..
----------   1 root      65533         395  Jun 10  1997 apple
----------   1 root      65533         395  Jun 10  1997 orange
----------   1 root      65533         395  Jun 10  1997 grapes
----------   1 root      65533         395  Jun 10  1997 pears
----------   1 root      65533         395  Jun 10  1997 fruits
----------   1 root      65533         395  Jun 10  1997 peaches
----------   1 root      65533         395  Jun 10  1997 plums
----------   1 65533     65533         395  Jun 10  1997 tangerine
```

Notice how some of the files including the dot (.) directory has changed to the ownership of "root". That is because the files that have changed are owned by the "Administrators" account on NT, and therefor are mapped to UID 0, which is recognized by UNIX as "root".

Question: Although, why is the file "tangerine" still owned by 65533?
Answer: UNIX doesn't recognize the owner!! -----> time for SHOWPROT again!!

This time, we are going to use showprot on a file "tangerine":

showprot D:\NFS\Work\Users\tangerine | more

Full lookup of file d:\nfs\work\users\tangerine (#links = 1)
Owner - Bedrock\Alwyn (an user)
Group - Bedrock\Domain Users (a group)
DACL - \Everyone (an Access-Allowed-Ace-Type)
This ACE is inherited by contained non-directories

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

```
   This ACE only applies to contained files, not to the file itself
   Access_Mask = 0x10000000
   Generic-All set
Equals first DACL
DACL - \Everyone (an Access-Allowed-Ace-Type)
   This ACE has no flags.
   Access_Mask = 0x001f01ff
   Access = File-All-Access
...
...
```

Now we know why the file "tangerine" is mapped as nobody (65533).. because UNIX doesn't know who ALWYN is. Although, Alwyn has an account on UNIX called dx144a32. Just like the root account, we need to know the UID and GID information for dx144a32. So we need to use the id command once again -- this time for the UNIX user dx144a32:

```
$ id dx144a32
uid=900(dx144a21) gid=10(staff)
```

Now that we know the NT Username associated with "tangerine" and the UID and GID to associate with the file.. we can go over to the Control Panel and go into the NFS Server icon. Under NT Username, put in EXACTLY what you saw in the showprot utility which is "Alwyn" and map that NT User to the effective UID for dx144a32 (which is 900). While we're in the mapping screen, we are also going to map the group "Domain Users" to GID 10, and put in "Administrators" (under NT Group) and map it to GID 3 (the default GID for root). Remember that "Administrators" (plural) is an alias, not the actual user Administrator (singular).

Click on "add" for each entry and then click on Save. Once this is done, go back to the UNIX client and do an "ls -la" in the mount point again. You should see something like this:

```
d------rwx   2 root      root       4096  May 28 14:47 .
drwxr-xr-x  210 bin       bin        8192  May 25 12:50 ..
-------rwx   1 root      root        395  Jun 10  1997 apple
-------rwx   1 root      root        395  Jun 10  1997 orange
-------rwx   1 root      root        395  Jun 10  1997 grapes
-------rwx   1 root      root        395  Jun 10  1997 pears
-------rwx   1 root      root        395  Jun 10  1997 fruits
-------rwx   1 root      root        395  Jun 10  1997 peaches
-------rwx   1 root      root        395  Jun 10  1997 plums
-------rwx   1 dx144a32   staff       395  Jun 10  1997 tangerine
```

Starting to look better, but what about the permissions?

Question: Why are the permissions blank?
Answer: Simple.. There are none..

If you bring up file manager/explorer under NT.. right click on the exported directory and goto security/permissions. Most likely, the default NT group of "everyone" is in the Members of box with "Full Control". The group "Everyone" on NT is directly associated with the Other permissions on UNIX (last 3 bits). Although, this document doesn't go into much detail about UNIX permissions.

In this case, click on Add and select Administrators (you will only see this, if the computer selected at the top of the screen is the local NT machine and not the domain). Also select Domain Users and Creator Owner, and give them "full contrrol". Click on OKAY.

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

Before you click on Okay under permissions, you might want to replace permissions on Subdirectories as well as exsisting files. This way, the permissions will take effect on all the sub directories and not just the main level.

Once, and only once this operation has been completed.. check the permissions on UNIX by doing another "ls -la" in the mountpoint. You should see something like this:

```
drwxrwxrwx   2 root      root        4096  May 28 14:47 .
drwxr-xr-x  210 bin       bin         8192  May 25 12:50 ..
-rwxrwxrwx   1 root      root         395  Jun 10  1997 apple
-rwxrwxrwx   1 root      root         395  Jun 10  1997 orange
-rwxrwxrwx   1 root      root         395  Jun 10  1997 grapes
-rwxrwxrwx   1 root      root         395  Jun 10  1997 pears
-rwxrwxrwx   1 root      root         395  Jun 10  1997 fruits
-rwxrwxrwx   1 root      root         395  Jun 10  1997 peaches
-rwxrwxrwx   1 root      root         395  Jun 10  1997 plums
-rwxrwxrwx   1 dx144a32  staff        395  Jun 10  1997 tangerine
```

There is no password needed to access the NFS Server, as authentication is handled on the client when the user logs into the system.

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

hclnfsd Overview

Article ID : MS5000008
Last reviewed: Jun 26, 1997
Product : NFS Maestro Version 5
OS : UNIX

DISCUSSION

hclnfsd Overview: What HCLNFSD does and how to run it.

RESOLUTION

Authentication by an NFS Server is performed by HCLNFSD

Authentication is a means of validating the username and password entered by a user. File security and access to resources are controlled based upon the privileges that a particular user is assigned during authentication. Authentication is one of the functions that the pcnfsd or hclfsd server daemon provides.

A UNIX username/password is required when nfs linking a specific file, or by using the nfs register command. The Authentication process is invoked when an nfs link command (or its equivalent Connect As in the File Manager) is attempted. A username/password (either specifically entered or the username/password stored with the nfs register command) is encrypted and a call is made to the pcnfsd or bwnfsd server daemon. If this succeeds, a user ID (UID) and group ID (GID) are returned to the NFS Client which are used for all subsequent NFS calls. If this authentication call fails, an appropriate error message is displayed indicating that the username/password is invalid.

On subsequent NFS calls from the client to the server, the server checks the file access rights to ensure that the user specified UID and GID has the appropriate permissions to server files. The UID/GID are also used to assign file privileges when creating new files on the NFS Server.

HCLNFSD Syntax

Hclnfsd is the authentication, print, and lock daemon for Maestro, and is executed on the NFS File Server machine. Refer to the Integrating hclnfsd into the network topic for information about implementing hclnfsd on your network. Before hclnfsd can be run on the server it must be compiled.

This daemon should generally be run by a ROOT user on the Unix machine and not just any logged in user. It would be best if the user was logged in as ROOT and did not su to ROOT or 'su -' to ROOT

It is very important that hclnfsd is not stopped with a "kill -9 xxxx" from the UNIX console. Hclnfsd should only be stopped with "kill xxxx", where xxxx is the PID of the process.

SYNTAX

hclnfsd [-A] [-d [-d]] [-l log] [-s server_file] < -X | /staging_directory>

The case and the order is very imporatant. If either the wrong case or the wrong order is used the daemon will not function as designed and most often show you the syntax line.

-A

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

The HCLNFSD daemon will authenticate a username and password request from any client that calls it. (these can be NFS from HCL, NFS from Novell, NFS from 3com, NFS for DOS from Chameleon, NFS from Intercon for MacIntosh etc.) It will also authenticate the username and password if the request is sent from another server. ie. If you try to mount a drive on server A and server A sends the request to server B that is running the HCLNFSD daemon, server B has the option to authenticate for server A.
The -A option tells the daemon to authenticate for any machine that sends the request. The -A switch is the most commonly used one. Unless this is a very secure environment you can leave this one on all of the time. Without the -A or -s options, hclnfsd only authorizes access to the machine it is executing on. The -A option tells the daemon to serve all hosts.

-d

This starts the daemon in debug mode. This option is optional and is not necessary if using the -l option. This will start the daemon in a logging method that will display all HCLNFSD daemon activity on the consol. This is useful and can be piped to a file to be sent to support to debug printing, and linking problems. To stop the daemon when running like this you will need to issue a CTRL+C on the server or start another telnet/consol session, find the PID for the daemon and kill it. (don't use the -9 option.). You can start the dameon in debug mode and have it running in the background if at the end of the command you use the & sign.

-l log

This will enable Logging in the hclnfsd daemon and put the output to a file. Generally it is good to use this only if support tells you to and to put the file to /dev/null. When the daemon starts it makes calls to the OS on the UNIX machine asking it do things. When it makes these calls it will wait for the server to answer (with either an ack or a result) and then continue with the next call or request for the OS or a process on the OS. Some UNIX OS's don't want anything waiting around for a return of information. (Solaris 2.5+/and Alpha 3.0+ servers are 2 of the most common0. The -l option will make the call and not wait to return, but will disconnect and send the output to a file. Most commonly this is needed if the user runs the daemon and the daemon appears to CORE DUMP or hang. Have the user start the daemon in debug mode (-d) and see if this helps. if the daemon no longer hangs or CORE DUMPS, kill it and restart it with the -l option. This will mean that you have found a server that does not like a daemon to start a process and wait for the result.

-s

The -s servers_file option allows the daemon to authenticate more than one file server but not all servers (like when using the -A option). The entries in the specified file will indicate which additional machines are to be served by the daemon, and also which other server machines have such daemons. The format of this file contains any combination of the following lines:

+Address/Name
-Address/Name
=Address/Name Address/Name

where Address/Name refers to either an IP address such as 123.231.73.19, an Internet name such as maccs.mcmaster.ca or a Network_Address/Name which matches all IP addresses on the specified network. The "+" field means that the daemon is to service requests for the named host/network.

The "+" field means that the daemon is to authenticate for the named host/network.
The "-" field means that the daemon is not to service requests for the named host/network.
The "=" field means that the daemon is to receive requests for the first named host/network and pass them to the second named host where there is another such daemon running.

Hosts with more than one IP address must use either the -A option or the -s option with all of the hosts' IP addresses in the servers_file. For example:

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

The line in the servers_file file on host "A" might read:

C D

The line in the servers_file file on host "D" would read:

+C

With the above parameters entered, the following sequence would take place when PC "B" wants to link a file system on host "C".

1. PC "B" attempts to authorize on host "C".
2. Host "C" returns "no server" error.
3. PC "B" tries default server "A" for host "C".
4. Host "A" responds with a redirect to "D".
5. PC "B" attempts to authorize on host "D" for "C".
6. Host "D" responds to the authorization.

Note that the IP address 0.0.0.0 is a wild-card match. Further note that the daemon searches for the first matching entry in the file, so if the first line of the file is:

+0.0.0.0

any following lines will be completely ignored.

-X

When starting the HCLNFSD daemon you must specify a staging directory that will be used as a temp directory. Most commonly this is used for printing. If the user does not want to do printing there is no need for the daemon to spool anything and the staging directory is not needed. The -X option will allow the user to start the daemon and not specify a directory.

[staging_directory]

If the -X option is used this can be left out. When starting the HCLNFSD daemon you will need to have a directory with 777 permissions that everyone can write to. This directory must be exported and be mountable by ROOT and by the Unix machine. This directory is used for spooling print jobs. There are some sites that do not want a directory on the server that has 777 permissions (because anyone can write things into it), and others that will not be using printing. If either of the above is true have the user use the -X option and not use the directory.

DESCRIPTION

The hclnfsd command (executed on the NFS File Server computer) creates a process which provides authentication of users for NFS activities and also provides the capability to print files on the Server machine to networked printers.

When an NFS client prints a file and the client uses HCLNFSD it will send the print job in a raw binary format to the HCLNFSD daemon. HCLNFSD will take the job spool it with a number and queue the job. It will then call LPD with the compiled command in the daemon and send the file to the server LPD daemon. The directory where the file is queued is the staging directory.

The print server feature uses the file /etc/printcap to determine the names of the available printers. This file is also required to be present on SYS V- based UNIX systems which do not have /etc/printcap files normally. See the topic on Printing for more information. The specified directory must be a world-writable directory where files to be printed are temporarily stored before printing.

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.

The directory must be included in the /etc/exports file and available for mounting by the server host as well as all PCs running HCLNFS. The authentication function allows UID and GID mapping between PC users and the NFS Server machine based upon username and password authentication. Up to 16 GIDs associated with the UID are passed to HCLNFS for subsequent access to the file systems. The locking function allows HCLNFS to implement DOS 3.1 record locking and file sharing functions.

Hclnfsd requires other daemons such as portmapper, nfsd, and the mount daemon to be active on the server.

Hclnfsd must be started by user root or included at the end of the system start-up file (typically rc.local). It should not be started by a Super User

©1997 Hummingbird Communications LTD. All rights reserved. Legal Notices.