

**NAME**

exportfs - maintain table of exported NFS file systems

**SYNOPSIS**

```
/usr/sbin/exportfs [-avi] [-o options,..] [client:/path ..]
/usr/sbin/exportfs -r [-v]
/usr/sbin/exportfs [-av] -u [client:/path ..]
/usr/sbin/exportfs [-v]
/usr/sbin/exportfs -f
```

**DESCRIPTION**

An NFS server maintains a table of local physical file systems that are accessible to NFS clients. Each file system in this table is referred to as an *exported file system*, or *export*, for short.

The **exportfs** command maintains the current table of exports for the NFS server. The master export table is kept in a file named */var/lib/nfs/etab*. This file is read by **rpc.mountd** when a client sends an NFS MOUNT request.

Normally the master export table is initialized with the contents of */etc/exports* and files under */etc/exports.d* by invoking **exportfs -a**. However, a system administrator can choose to add or delete exports without modifying */etc/exports* or files under */etc/exports.d* by using the **exportfs** command.

**exportfs** and its partner program **rpc.mountd** work in one of two modes: a legacy mode which applies to 2.4 and earlier versions of the Linux kernel, and a new mode which applies to 2.6 and later versions, providing the **nfsd** virtual filesystem has been mounted at */proc/fs/nfsd* or */proc/fs/nfs*. On 2.6 kernels, if this filesystem is not mounted, the legacy mode is used.

In the new mode, **exportfs** does not give any information to the kernel, but provides it only to **rpc.mountd** through the */var/lib/nfs/etab* file. **rpc.mountd** then manages kernel requests for information about exports, as needed.

In the legacy mode, exports which identify a specific host, rather than a subnet or netgroup, are entered directly into the kernel's export table, as well as being written to */var/lib/nfs/etab*. Further, exports listed in */var/lib/nfs/rmtab* which match a non host-specific export request will cause an appropriate export entry for the host given in *rmtab* to be added to the kernel's export table.

**OPTIONS**

- a** Export or unexport all directories.
- o options,...**  
Specify a list of export options in the same manner as in [exports\(5\)](#).
- i** Ignore the */etc/exports* file and files under */etc/exports.d* directory. Only default options and options given on the command line are used.
- r** Reexport all directories, synchronizing */var/lib/nfs/etab* with */etc/exports* and files under */etc/exports.d*. This option removes entries in */var/lib/nfs/etab* which have been deleted from */etc/exports* or files under */etc/exports.d*, and removes any entries from the kernel export table which are no longer valid.
- u** Unexport one or more directories.
- f** If */proc/fs/nfsd* or */proc/fs/nfs* is mounted, flush everything out of the kernel's export table. Fresh entries for active clients are added to the kernel's export table by **rpc.mountd** when they make their next NFS mount request.
- v** Be verbose. When exporting or unexporting, show what's going on. When displaying the current export list, also display the list of export options.

## DISCUSSION

### Exporting Directories

The first synopsis shows how to invoke **exportfs** when adding new entries to the export table. When using **exportfs -a**, all exports listed in */etc/exports* and files under */etc/exports.d* are added to */var/lib/nfs/etab*. The kernel's export table is also updated as needed.

The *host:/path* argument specifies a local directory to export, along with the client or clients who are permitted to access it. See [exports\(5\)](#) for a description of supported options and access list formats. To export a directory to the world, simply specify *:/path*.

The export options for a particular host/directory pair derive from several sources. The default export options are **sync,ro,root\_squash,wdelay**. These can be overridden by entries in */etc/exports* or files under */etc/exports.d*.

A system administrator may override options from these sources using the **-o** command-line option on **exportfs**. This option takes a comma-separated list of options in the same fashion as one would specify them in */etc/exports*. In this way **exportfs** can be used to modify the export options of an already exported directory.

### Unexporting Directories

The third synopsis shows how to unexport a currently exported directory. When using **exportfs -ua**, all entries listed in */var/lib/nfs/etab* are removed from the kernel export tables, and the file is cleared. This effectively shuts down all NFS activity.

To remove an export, specify a *host:/path* pair. This deletes the specified entry from */var/lib/nfs/etab* and removes the corresponding kernel entry (if any).

### Dumping the Export Table

Invoking **exportfs** without options shows the current list of exported file systems. Adding the **-v** option causes **exportfs** to display the export options for each export.

## EXAMPLES

The following adds all directories listed in */etc/exports* and files under */etc/exports.d* to */var/lib/nfs/etab* and pushes the resulting export entries into the kernel:

```
# exportfs -a
```

To export the */usr/tmp* directory to host **django**, allowing insecure file locking requests from clients:

```
# exportfs -o insecure_locks django:/usr/tmp
```

To unexport the */usr/tmp* directory:

```
# exportfs -u django:/usr/tmp
```

To unexport all exports listed in */etc/exports* and files under */etc/exports.d*:

```
# exportfs -au
```

## USAGE NOTES

Exporting to IP networks or DNS and NIS domains does not enable clients from these groups to access NFS immediately. Rather, these sorts of exports are hints to [rpc.mountd\(8\)](#) to grant any mount requests from these clients. This is usually not a problem, because any existing mounts are preserved in *rmtab* across reboots.

When unexporting a network or domain entry, any current exports to members of this group will be checked against the remaining valid exports and if they themselves are no longer valid they will be removed.

## FILES

*/etc/exports*

input file listing exports, export options, and access control lists

<i>/etc/exports.d</i>	directory where extra input files are stored. <b>Note:</b> only files that end with <i>.exports</i> are used.
<i>/var/lib/nfs/etab</i>	master table of exports
<i>/var/lib/nfs/rmtab</i>	table of clients accessing server's exports

**SEE ALSO**

[exports\(5\)](#), [rpc.mountd\(8\)](#), [netgroup\(5\)](#)

**AUTHORS**

Olaf Kirch <okir@monad.swb.de>

Neil Brown <neilb@cse.unsw.edu.au>