Chapter 13. Network Commands

Table of Contents

The network commands chapter explains various tools which can be useful when networking with other computers both within the network and accross the internet, obtaining more information about other computers. This chapter also includes information on tools for network configuration, file transfer and working with remote machines.

netstat

> Displays contents of /proc/net files. It works with the Linux Network Subsystem, it will tell you what the status of ports are ie. open, closed, waiting, masquerade connections. It will also display various other things. It has many different options.

tcpdump

> This is a sniffer, a program that captures packets off a network interface and interprets them for you. It understands all basic internet protocols, and can be used to save entire packets for later inspection.

ping

> The ping command (named after the sound of an active sonar system) sends echo requests to the host you specify on the command line, and lists the responses received their round trip time.
>
> You simply use ping as:

```
ping ip_or_host_name
```

> Note to stop ping (otherwise it goes forever) use **CTRL-C** (break).
>
>> **Please note:** Using ping/smbmount/ssh or other UNIX system programs with a computer name rather than IP address will only work if you have the computer listed in your /etc/hosts file. Here is an example:

```
192.168.1.100 new
```

>> This line says that their is a computer called “new” with IP address 192.168.1.100. Now that it exists in the /etc/hosts file I don't have to type the IP address anymore, just the name “new”.

hostname

> Tells the user the host name of the computer they are logged into. Note: may be called *host*.

traceroute

traceroute will show the route of a packet. It attempts to list the series of hosts through which your packets travel on their way to a given destination. Also have a look at *xtraceroute* (one of several graphical equivalents of this program).

Command syntax:

```
traceroute machine_name_or_ip
```

tracepath

tracepath performs a very simlar function to *traceroute* the main difference is that *tracepath* doesn't take complicated options.

Command syntax:

```
tracepath machine_name_or_ip
```

findsmb

findsmb is used to list info about machines that respond to SMB name queries (for example windows based machines sharing their hard disk's).

Command syntax:

```
findsmb
```

This would find all machines possible, you may need to specify a particular subnet to query those machines only...

nmap

“ network exploration tool and security scanner”. *nmap* is a very advanced network tool used to query machines (local or remote) as to whether they are up and what ports are open on these machines.

A simple usage example:

```
nmap machine_name
```

This would query your own machine as to what ports it keeps open. *nmap* is a very powerful tool, documentation is available on the nmap site as well as the information in the manual page.

Network Configuration

ifconfig

This command is used to configure network interfaces, or to display their current configuration. In addition to activating and deactivating interfaces with the “up” and “down” settings, this command is necessary for setting an interface's address information if you don't have the *ifcfg* script.

Use *ifconfig* as either:

```
ifconfig
```

This will simply list all information on all network devices currently up.

```
ifconfig eth0 down
```

This will take eth0 (assuming the device exists) down, it won't be able to receive or send anything until you put the device back “up” again.

Clearly there are a lot more options for this tool, you will need to read the manual/info page to learn more about them.

ifup

Use *ifup device-name* to bring an interface up by following a script (which will contain your default networking settings). Simply type *ifup* and you will get help on using the script.

For example typing:

```
ifup eth0
```

Will bring eth0 up if it is currently down.

ifdown

Use *ifdown device-name* to bring an interface down using a script (which will contain your default network settings). Simply type *ifdown* and you will get help on using the script.

For example typing:

```
ifdown eth0
```

Will bring eth0 down if it is currently up.

ifcfg

Use *ifcfg* to configure a particular interface. Simply type ifcfg to get help on using this script.

For example, to change eth0 from 192.168.0.1 to 192.168.0.2 you could do:

```
ifcfg eth0 del 192.168.0.1
ifcfg eth0 add 192.168.0.2
```

The first command takes eth0 down and removes that stored IP address and the second one brings it back up with the new address.

route

The *route* command is the tool used to display or modify the routing table. To add a gateway as the default you would type:

```
route add default gw some_computer
```

Prev Home Next
Mathematical tools Internet Specific Commands