

CHAPTER 12

Network Related Commands

12.1 THE PING COMMAND

This is used to check whether a given host is reachable currently from our machine or not. It can take host IP address or name. The following shows output of two ping sessions. We may get information such as host is reachable or not, delay packet loss, etc.

ping mail.yahoo.com

Pinging in-login.lgg1.b.yahoo.com [202.86.7.110] with 32 bytes of data:

Reply from 202.86.7.110: bytes=32 time=115ms TTL=53

Reply from 202.86.7.110: bytes=32 time=137ms TTL=53

Reply from 202.86.7.110: bytes=32 time=63ms TTL=53

Reply from 202.86.7.110: bytes=32 time=63ms TTL=53

Ping statistics for 202.86.7.110:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 63ms, Maximum = 137ms, Average = 94ms

ping 202.86.7.110

Pinging 202.86.7.110 with 32 bytes of data:

Reply from 202.86.7.110: bytes=32 time=64ms TTL=53

Reply from 202.86.7.110: bytes=32 time=63ms TTL=53

Reply from 202.86.7.110: bytes=32 time=63ms TTL=53

Reply from 202.86.7.110: bytes=32 time=63ms TTL=53

Ping statistics for 202.86.7.110:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 63ms, Maximum = 64ms, Average = 63ms

12.2 THE FINGER COMMAND

This command is used to know the information about the user such as when he has logged into machine, when did he/she see their email, etc in addition to content of .plan (and other files) of his home directory.

Example:

```
finger root gave the following result.
Login: root                      Name: root
Directory: /root                 Shell: /bin/bash
On since Tue Feb 12 09:55 (IST) on :0 (messages off)
On since Tue Feb 12 09:59 (IST) on pts/0 from :0.0
New mail received Tue Feb 12 10:01 2002 (IST)
  Unread since Sun Feb 10 00:03 2002 (IST)
Plan :
I have class at 9.00AM
10.00AM
4.30 to 9.00PM
```

If any user wants to convey any thing to the people who fingers his account he can write in his .plan file. Best thing we can write is our schedule today. Such that, other people can see and accordingly they can start interactive sessions such as talk, chat, or calling by phone etc.

If this command is executed without any arguments then displays details of all currently logged in users (similar to who command) such as :

| Login | Name | Tty | Idle Login Time | Office | Office Phone |
|-------|------|-------|-----------------|-------------|--------------|
| rao | | pts/2 | Feb 12 10:37 | (localhost) | |
| root | root | *:0 | Feb 12 09:55 | | |
| root | root | pts/0 | Feb 12 10:33 | (:0.0) | |
| root | root | pts/1 | Feb 12 10:36 | | |

Remote Login Services

12.3 RLOGIN

With the help of this command it is possible to login to remote machine if we happened to have legal username and password on that machine. When we do so, the current machine becomes terminal to that remote machines. After that whatever file we create it will be stored in that remote machine. When we run a command, that remote machines processor and RAM is used for running the same.

Example :

rlogin IPaddress_or_machinename -l username

It will prompt the password and after entering valid password we will see that remote machines prompt. For proper functioning TERM environment variable on local machine should be set appropriately such that it matches with that remote machine. Usually, rlogin is used for GUI based remote login service unlike telnet service.

12.4 TELNET COMMAND

telnet IPaddress_or_machinename

The following output appear on the screen

```
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
Red Hat Linux release 9 (Shrike)
Kernel 2.4.20-8 on an i686
login :
Login incorrect
login :
```

When we enter legal username and password then we will be logging into that remote machine and we will see it's prompt. Here also the local machine becomes dumb terminal for the remote machine.

Unlike rlogin service this supports only character based remote login service.

12.5 THE FTP COMMAND

This command is used to transfer files from one machine to another machine.

ftp Ipaddress_or_machiename

This command gives a prompt namingly ftp> after we enter legal user name and password of that remote machine. Once we have logged in, we can download files with command **get filename**. We can use commands such as ls, cd etc on remote machine directory while mls, mcd commands can be used on local machine. We can set transfer as ASCII or binary by simply typing ASCII or binary commands at ftp> prompt.

We can put files of the local directory using **put filename** command. On some ftp servers we can download files using mget and upload files using mput command.

If we execute ftp command without any argument then we will see ftp> prompt. Using **open Ipaddressormachinename** we can connect to remote machine for file transfer. By type ! symbol at the ftp> prompt we can exit from the ftp program.

There are many ftp servers are available in the internet for free download. While logging into those servers we can login with anonymous as username and our email address as password. Thus these servers are often called as anonymous servers. From these servers we can download only. If we have some SW is available and wants to be available freely to others we have to contact these servers' administrators who can give permission temporarily to upload our SW into their servers.

12.6 ARP COMMAND

This command is used to manipulate arp cache. That is, we can see the arp cache, remove a hosts entry from the arp cache, etc.,. In a networked system when a packet arrives at router machine (often a UNIX machine) then the IP address to Ethernet address mapping is needed. This is achieved by arp protocol. These mappings are stored in arp cache such that next time another packet arrives with the same IP address then its Ethernet or physical address is calculated by carrying out a lookup operation on this arp cache. With arp command we can modify, view, delete the entries of this cache.

| | |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| arp -a | Displays all entries are displayed |
| arp -a hostname | Displays entry of the given host |
| arp -d hostname | Removes the entry of the specified host |
| arp -s hostname HW_addr | Creates manually ARP entry for the host with the given hardware address (HW address has to be given in hexadecimal separated by colons) |

12.7 THE MAIL COMMAND

Though we have many interactive and GUI based mail readers such as elm, pine, Eudora etc., basic UNIX mail command is very efficient. We shall talk about in the following pages.

Some common Mail commands

The table below lists some of the more common commands to the **mail** program. Please consult your online documentation (**man mail**) to learn any site specific differences between your system and the commands listed here.

| Option | Description | | Command | Description |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|---------|------------------------------------------------------------------------------------------|
| -v | Display the details of the mail delivery on the terminal. | | d | Delete the current message. |
| -s | What follows is the subject of the e-mail. ALWAYS use a subject, it is only polite! | | p | Print the current message |
| -c | Address of a carbon copy, the recipient will see this address. | | + & - | Move up or down one or more lines moving in mail is like moving in the ed editor. |
| -b | Blind copy address. The recipient will NOT see this address. Useful for sending a copy to yourself when you don't want the target of the e-mail to know you kept a copy for reference. | | r | Reply to the current message. |
| | | | u | Undelete a mail message that you deleted. Only works during the current mail session. |
| | | | x or q | Exit or quit from the current mail session. Generally, quitting is cleaner than exiting. |

To check whether we have any mail, we can simply type **mail** at the command prompt. If we have mail, the system will begin showing it to us. If we have no mail, we will be told, rather briskly, "No mail for nbv".

To send mail, all we need do is type the **mail** command followed by the e-mail address of the recipient. For example :

```
mail ritchcen@ritchcenter.com
Subject: UNIX
Dear Fellow,
Do you think I will ever teach you without your real involvement?
NBV
^D
```

In this example, nbv is sending mail to ritchcen. The mail program prompted for the "Subject" and we may type one it. (It is really, really important to always have a subject for your e-mail. Some systems will reject e-mails without a subject, and many users will ignore e-mails without subjects.) After ritchcen typed the subject of this e-mail, and touched <Enter>, the program presented him with an empty line, with no system prompt. From that point on, every thing he typed was added to the mail message. When he had finished his extensive epistle, ritchcen typed a control-D "**^D**" as the first character of a new line. When the mail program sees the "**^D**" as the first character of a new line it interprets it as "End-of-File" and terminates keyboard input. Usually e-mail isn't used for writing the Great American Novel, they are short, to the point, and deal with a single topic.

In addition to the commands and the command line options, there are a series of "tilde commands" you can use within mail to cause different actions. Please consult your local system man pages to determine these options.

There are many ways available to configure the mail program to meet individual needs. These configurations can be made permanent by adding them to a file called **.mailrc** in your login directory. (There are a collection of dotted files whose names end in "rc" which stands for *runnable commands*. These are setup files for the various commands. The **.mailrc** file contains the setup information for the mail program. When it begins executing, mail reads **.mailrc**, if there is one, to see if there are any special directives it should use when initializing.

On some UNIX systems, you will be prompted when a new mail message arrives in your mailbox. The program that does the notification is called "biff" and is named after the dog of one of the early UNIX developers. Her dog used to bark at the mailman, so she always knew when she had mail. When it came time to name the command that would alert users when they had mail...it was a natural to name it after the dog who served the same function. If you are not running "biff", then the mail program will tell you that you have new mail when you log in, or when you run the mail program.

The following sequence of commands and replies shows the user interacting with the mail program and reading new mail. The commands in **bold** are what the user typed, and the unbolded lines are presented by the system. After looking at the command table above, you should be able to decipher what is happening. Take your time and study the example :

```
ritchcen % You have mail
ritchcen % mail
Mail version 2.18 5/19/03. Type ? for help.
"/usr/spool/mail/ritchcen": 1 message 1 new
>N 1 pkrao Fri Jul 19 18:23 13/176 "will
they buy Plan-9?"
>N 2 pmg Sat Jul 20 13:25 13/176 "UNIX
pew!"
& t 2
Message 2:
From pmg@xyz.com Sat Jul 20 13:25:18
1991
Date: Sat, 20 Jul 91 13:25:15 BST
From: pmg@xyz.com
To: ritchcen@ritchcenter.com
Subject: Hai
Status: N
Dear Dr Ritchcen,
Do you think I am mad
PMG
& r
To: pmg@xyz.com
Subject: Re: Hai
Thats a hard one to answer. All I know is that
ritchcenter
& s 2 ritchcen200791
"ritchcen200791" [New file] 11/254
& d 2
& q
```

In addition, the mail program has a multitude of features, like replying to mail sent to you whilst you are on holiday, or if you have accounts on multiple systems it can forward mail to a single system, c.f. % **man mail** (specifically the sections on vacation and forward).

12.8 CONCLUSIONS

In this chapter, we have introduced the useful Internet related commands such as arp, ftp, telnet, rlogin, mail, etc.,. All the commands are explained with live examples. Each example is supplied with output of a live session.