**Common Linux (Unix) Commands**

|  |  |  |  |
| --- | --- | --- | --- |
| **Command with Brief Description** | **Pg** | **Command with Brief Description** | **Pg** |
| **alias** Create an alias for a command  **apropos** Performs a man keyword search  **cal** Displays a calendar  **cat** Displays contents of a file or files  **cd** Change the current Directory **chgrp** Change group ownership of files **chmod** Change file/directory access rights **chown** Change file owner and group  **clear** Clear terminal screen  **cp** Copy file/s to another location  **crontab** Task scheduler  **date** Display or set time and date **dd** Low level file duplication  **df** Report disk space usage  **dhclient (pump)** Request DHCP configuration **dmesg** Display system messages  **du** Estimate file space usage  **e2fsck** Check & Repair Linux File Systems  **env** Display Environmental variables  **ethtool** Change Ethernet Card Settings  **fdisk** Linux Partition table manipulator  **file** Display file types  **find** Search for files that meet a criteria  **ftp** Transfer files between hosts  **grep** Search Files for specific text  **groupadd** Add a new user group  **groupdel** Delete an existing user group  **groupmod** Modify user group properties  **groups** Display group names a user is in  **grub-install** Restore Grub Loader to MBR  **hostname** Display or set system name  **id** Display user and group ids  **ifconfig** Configure Network Settings  **info** Help info for commands  **init** Change current Run level  **ipcalc** Calc. Info from IP Address/Mask  **kill** Kill a process by PID  **killall** Kill processes by name  **kudzu** Check for new/changed hardware  **less** Display output one screen at a time  **ln** Make links between files  **loadlin.exe** Boot into Linux from MSDOS  **locate**  Find location of files & directories  **lpc** Line printer control program  **lpr** Off line print  **lprm** Remove jobs from the print queue  **ls** List information about file(s)  **man** See Info page 20 | **3**  **3**  **3**  **3**  **3**  **4**  **4**  **5**  **6**  **6**  **6**  **7**  **9**  **10**  **10**  **13**  **13**  **13**  **14**  **14**  **15**  **15**  **16**  **18**  **20**  **21**  **21**  **21**  **21**  **22**  **22**  **22**  **23**  **24**  **24**  **25**  **25**  **26**  **26**  **27**  **28**  **29**  **29**  **29**  **31**  **32**  **32**  **..** | **minicom** CLI Serial Communications  **mkbootdisk** Create Linux Boot Floppy  **mkdir** Create new folder(s)  **mkfs** Make a File System (format)  **more** Display output one screen at a time  **mount** Mount a file system  **mv** Move or rename files or directories  **netstat** Display TCP/IP connection stats  **nice** Exec. a command with different priority  **nl** Number lines and write files  **ntsysv** CLI RunLevel Services configuration  **passwd** Modify a user password  **pico**  Command Line text editor  **ping** Check network connectivity  **pr** Convert text files for printing  **ps** Display active processes  **pwd** Print Working Directory  **rawrite** MSDOS, create boot image floppy  **redhat-config-\*** Redhat Configuration Tools  **renice** Change priority of running process  **rm** Remove files  **rmdir** Remove folder(s)  **route** Add network routes to interface  **rpm** Remote Package Manager  **service** Start, Stop, Restart services  **serviceconfig** GUI Level 5 Services Config  **shutdown** Shutdown or restart Linux  **sleep** Delay for a specified time  **sort** Sort text files  **su** Run command as different user/group  **tar** Tape ARchiver  **top** Show active CPU-intensive processes  **touch** Creates empty file  **tr** Text translation utility.  **traceroute**  Trace path taken to host address  **umount** Unmount a device  **uname** Display OS and Hardware info  **updatedb** Update slocate database (see locate)  **umask** Set default permissions on new files  **unalias** Remove an command alias  **useradd**  Add a new user  **userdel** Delete an existing user  **usermod** Modify user properties  **users** Display logged in user names  **wc** Displays byte, word, and line counts  **whereis** Find location of executable file  **which** Display full path of commands  **who whoami** Print all usernames currently logged in | **33**  **33**  **34**  **34**  **35**  **36**  **37**  **37**  **38**  **38**  **39**  **40**  **40**  **41**  **41**  **43**  **45**  **45**  **45**  **46**  **47**  **47**  **47**  **48**  **51**  **..**  **52**  **52**  **53**  **53**  **54**  **55**  **56**  **57**  **58**  **58**  **59**  **..**  **59**  **59**  **60**  **60**  **61**  **61**  **61**  **62**  **63**  **64** |

**Common Linux (UNIX) Commands**

Common Command Line Hotkeys

**<Ctrl><Alt> F1..F6** Switch to Command Line Session TTY0..5

**<Ctrl><Alt> F7** Switch to X-Windows Session

**<Ctrl><Alt> <Bksp>** Restart X-Windows (within an X-Window session).

**<Ctrl><Alt><Delete>** Restart Computer (same as shutdown -r now)

**<Ctrl> a** Move cursor to beginning of command line

**<Ctrl> c** Cancel Current Command

**<Ctrl> e**  Move cursor to end of command line

**<Ctrl> n** Next Command in command history

**<Ctrl> p** Previous Command in command history

**q** Exit More or Man pages

**up/down Arrow Keys** Command History Navigation (same as ^p and ^n)

**Command Resources**

The command documentation contained on the following pages was primarily compiled from the --help and man pages in Redhat 9.0 Linux.

The websites: [**http://www.onlamp.com/linux/cmd/**](http://www.onlamp.com/linux/cmd/%20) and [**http://www.ss64.com/bash/index.html**](http://www.ss64.com/bash/index.html) provided additional useful information.

The --help and --version switches are not shown to conserve space. It should be assumed that they are available for most commands. --help will show syntax and command options and --version will indicate the version number of the command being used. Some obsolete or infrequently used switches are also omitted (refer to the man pages for unabridged help for all commands)

**alias** Create an alias; aliases allow a string to be substituted for a word when it is used as the first word

of a simple command.

SYNTAX

**alias [-p] [name[=value] ...]**

**unalias [-a] [name ... ]**

If arguments are supplied, an alias is defined for each name whose value is given. If no value is given, alias will print the current value of the alias. Without arguments or with the -p option, alias prints the list of aliases

on the standard output in a form that allows them to be reused as input. unalias will remove each name from

the list of aliases. If -a is supplied, all aliases are removed.

**apropos** *string* *...*

Search the short manual page descriptions in the **whatis** database for occurrences of each *string* and display the result on the standard output. Like **whatis**, except that it searches for strings instead of words. Equivalent to **man -k**.

**cal** Displays a calendar

SYNTAX

**cal [-my] [month [year]]**

Cal displays a simple calendar. If arguments are not specified, the cur­rent month is displayed. The options are as follows:

**-m** Display Monday as the first day of the week.

**-y** Display a calendar for the current year.

**cat** Displays the contents of a file (concatenate)

SYNTAX

**cat [FILE]...**

Concatenate FILE(s), or standard input, to standard output.

**cd** Change Directory - change the current working directory to a specific Folder.

SYNTAX

**cd [directory]**

**.** (current directory) and **..** (parent directory) may be used to specify a directory path.

**chgrp** chgrp changes the group ownership of each given FILE to GROUP (which can be either a group name or a numeric group id) or to the group of an existing reference file.

SYNTAX

**chgrp [OPTION]... {GROUP |** - -**reference=FILE} FILE...**

KEY

**-c** Verbosely describe the action for each FILE whose group actually changes.

- -**reference=FILE**

Use the group of the reference FILE instead of an explicit GROUP.

**-R** Recursively change the group ownership of directories and their contents.

Example  
 Make Oracle the owner of the database directory  
 **$chgrp oracle /usr/database**

**chmod** Change access permissions for file/s or directories

SYNTAX*:*

**chmod [OPTION]... MODE[,MODE]... FILE...**

**chmod [OPTION]... NUMERIC\_MODE FILE...**

OPTIONS

**-v, - -verbose** output a diagnostic for every file processed

**-c, - -changes**  like verbose but report only when a change is made

**-R, - -recursive**  change files and directories recursively

**Numeric mode:**

From one to four octal digits, Any omitted digits are assumed to be leading zeros.   
 The first digit = selects attributes for the set user ID (4) and set group ID (2) and save text image (1)S  
 The second digit = permissions for the user who owns the file: read (4), write (2), and execute (1)  
 The third digit = permissions for other users in the files group: read (4), write (2), and execute (1)  
 The fourth digit = permissions for other users NOT in the files group: read (4), write (2), and execute (1)

**Examples**  
 Allow read permission to everyone: **chmod 444 *file***  
 Make a file readable and writable by the group and others: **chmod 066 *file***   
 Allow everyone to read, write, and execute the file: **chmod 777 *file*Symbolic Mode**

The format of a symbolic mode is [**ugoa**...][[**+-=**][**rwxXs­t**...]...][,...].   
  
 Multiple symbolic operations can be given, separated by commas. A combination of the letters ugoa controls which **users** access to the file will be changed: The user who owns it (u), users in the files group (g), Other users not in the files group (o), All users (a) If none of these are given, the effect is as if a were

given, but bits that are set in the umask are not affected. all users (a) is effectively user + group + others  
  
 The operator + causes the permissions selected to be added to the existing permissions of each file; - causes them to be removed; and = causes them to be the only permissions that the file has.  
  
 The letters **rwxXstugo** select the new **permissions** for the affected users: Read (r), Write (w), Execute (or access for directories) (x), Execute only if the file is a directory or already has execute permission for some user (X), Set user or group ID on execution (s), Save program text on swap device (t), The permissions that the user who owns the file currently has for it (u), The permissions that other users in the files group have for it (g),Permissions that other users not in the files group have for it (o).

**Examples** Deny execute permission to everyone: **chmod a-x *file***  
 Allow read permission to everyone: **chmod a+r *file***  
 Make a file readable/writable by the group/others: **chmod go+rw *file***  
  
 Allow everyone to read, write, and execute the file and turn on the set group-ID: **chmod =rwx,g+s *file***   
  
 Notes: When chmod is applied to a directory: read = list files in the directory  
 write = add new files to the directory, execute = access files in the directory

**chown** chown changes the user and/or group ownership of each given FILE.

SYNTAX

**chown [OPTION]... [USER] [ [:.] [GROUP] ] FILE...**

**chown [OPTION]... :GROUP FILE...**

If used, **NEW\_OWNER** specifies the new owner and/or group as follows:  **[USER] [ [:.] [GROUP] ]**

Some examples of how the owner/group can be specified: If only USER, user is made owner of file/s, and the files group is not changed. If USER with colon or dot and a GROUP, with no spaces between them, the group ownership changed as well. If a colon or dot but no group name follows USER, that user is made the owner of the files and the group of the files is changed to USER’s login group. If :GROUP (with USER omitted), only the group of the files is changed;

OPTIONS:

**-c** Verbosely describe the action for each FILE whose ownership actually changes.

**-R** - -**recursive** Recursively change ownership of directories and their contents.

**-v - -verbose** Verbosely describe the action (or non-action) taken for every FILE.

**clear** Clear terminal screen

**cp** Copy one or more files to another location, Copy SOURCE to DEST, or multiple SOURCE(s)

to DIRECTORY.

SYNTAX

**cp [OPTION]... SOURCE DEST**

**cp [OPTION]... SOURCE... DIRECTORY**

KEY

**-f, - -force** remove existing destinations, never prompt

-i, --interactive prompt before overwrite

**-l, - -link** link files instead of copying

-R, **- -**recursive copy directories recursively

-s, **- -**symbolic-link make symbolic links instead of copying

-u, **- -**update copy only when the SOURCE file is newer than the destination file or when the destination file is missing

-v, **- -**verbose explain what is being done

**Example** - copy floppy to home directory

**$ cp -f /mnt/floppy/\* /home/simon**

**crontab** [*options*] [*file*]

View, install, or uninstall your current *crontab* file. A privileged user can run **crontab** for another user by supplying **-u** *user*. A *crontab* file is a list of commands, one per line that will execute automatically at a given time. Numbers are supplied before each command to specify the execution time. The numbers appear in five fields, as follows:

*Minute*  0-59

*Hour*  0-23

*Day of month*  1-31

*Month*  1-12 Jan, Feb, Mar, ...

0-6, with 0 = Sunday Sun, Mon, Tue, ...

Use a comma between multiple values, a hyphen to indicate a range, and an asterisk to indicate all possible values. For example, assuming these *crontab* entries:

**59 3 \* \* 5 find / -print | *backup\_program***

**0 0 1,15 \* \* echo "Timesheets due" | mail *user***

The first command backs up the system files every Friday at 3:59 a.m., and the second command mails a reminder on the 1st and 15th of each month. The superuser can always issue the **crontab** command. Other users must be listed in the file ***/etc/cron.allow*** if it exists; otherwise, they must not be listed in ***/etc/cron.deny***. If neither file exists, only the superuser can issue the command.

Options

The **-e**, **-l**, and **-r** options are not valid if any *files* are specified.

**-e** Edit the users current *crontab* file (or create one).

**-l** Display the users *crontab* file on standard output.

**-r** Delete the users *crontab* file.

**-u** *user* Indicates which *user*s *crontab* file will be acted upon.

**date** [*options*] [**+***format*] [*date*]

DESCRIPTION

Print the current date and time. You may specify a display *format*. *format* can consist of literal text strings (blanks must be quoted) as well as field descriptors, whose values will appear as described in the following entries (the listing shows some logical groupings). A privileged user can change the systems date and time.

OPTIONS

**+***format*

Display current date in a nonstandard format. For example:

**% date +"%A %j %n%k %p"** Tuesday 248 15 PM

The default is %a %b %e %T %Z %YÂ—e.g., Tue Sep 5 14:59:37 EDT 2000.

**-d** *date*, **- -date** *date*

Display *date*, which should be in quotes and may be in the format *d* **days** or *m* **months** *d* **days** to print a date in the future. Specify **ago** to print a date in the past. You may include formatting (see the "Format" section that follows).

**-f** *datefile*, **--file=***datefile*

Like **-d** but printed once for each line of *datefile*.

**-I** [*timespec*], **- -iso-8601**[**=***timespec*]

Display in ISO-8601 format. If specified, *timespec* can have one of the values **date** (for date only), **hours**, **minutes**, or **seconds** to get the indicated precision.

**-r** *file*, **--reference=***file*

Display the time *file* was last modified.

**-s** *date*, **- -set** *date*

Set the date.

**-u**, **--universal**

Set the date to Greenwich Mean Time, not local time.

**Format**

**%** Literal **%**.

**-** Do not pad fields (default: pad fields with zeros).

**\_** Pad fields with space (default: zeros).

**%a** Abbreviated weekday.

**%b** Abbreviated month name.

**%d** Day of month (01-31).

**%h** Same as **%b**.

**%k** Hour in 24-hour format, without leading zeros (0-23).

**%l** Hour in 12-hour format, without leading zeros (1-12).

**%m** Month of year (01-12).

**%n** Insert a new line.

**%p** String to indicate AM or PM.

**%r** Time in **%I:%M:%S %p** (12-hour) format.

**%s** Seconds since "The Epoch," 1970-01-01 00:00:00 UTC (a nonstandard extension).

**%t** Insert a tab.

**%w** Day of week (Sunday = 0).

**%y** Last two digits of year (00-99).

**%A** Full weekday.

**%B** Full month name.

**%D** Date in **%m/%d/%y** format.

**%H** Hour in 24-hour format (00-23).

**%I** Hour in 12-hour format (01-12).

**%M** Minutes (00-59).

**%S** Seconds (00-59).

**%T** Time in **%H:%M:%S** format.

**%U** Week number in year (00-53); start week on Sunday.

**%V** Week number in year (01-52); start week on Monday.

**%W** Week number in year (00-53); start week on Monday.

**%Y** Four-digit year (e.g., 1996).

**%Z** Time zone name.

**Strings for setting date**

Strings for setting the date may be numeric or nonnumeric. Numeric strings consist of time, day, and year in the format *MMDDhhmm[[CC]YY][.ss]*. Nonnumeric strings may include month strings, time zones, a.m., and p.m.

*time* A two-digit hour and two-digit minute (*hhmm*); *hh* uses 24-hour format.

*day* A two-digit month and two-digit day of month (*MMDD*); default is current day and month.

*year* The year specified as either the full four-digit century and year or just the two-digit year;

the default is the current year.

**Examples**

Set the date to July 1 (**0701**), 4 a.m. (**0400**), 1995 (**95**):  **date 0701040095**

The command: **date +"Hello%t Date is %D %n%t Time is %T"** produces a formatted date as follows:

Hello Date is 05/09/93

Time is 17:53:39**dd** *options*

Make a copy of an input file (**if**) using the specified conditions, and send the results to the output file (or standard output if **of** is not specified). Any number of options can be supplied, although **if** and **of** are the most common and are usually specified first. Because **dd** can handle arbitrary block sizes, it is useful when converting between raw physical devices.

**Options**

**bs=***n* Set input and output block size to *n* bytes; this option overrides **ibs** and **obs**.

**cbs=***n* Set the size of the conversion buffer (logical record length) to *n* bytes. Use only if the conversion *flag* is **ascii**, **ebcdic**, **ibm**, **block**, or **unblock**.

**conv=***flags* Convert the input according to one or more (comma-separated) *flags* listed next. The first five *flags* are mutually exclusive.

**ascii** EBCDIC to ASCII.

**ebcdic** ASCII to EBCDIC.

**ibm** ASCII to EBCDIC with IBM conventions.

**block** Variable-length records (i.e., those terminated by a newline) to fixed-length records.

**unblock** Fixed-length records to variable-length.

**lcase** Uppercase to lowercase.

**ucase** Lowercase to uppercase.

**noerror** Continue processing after read errors.

**notrunc** Don’t truncate output file.

**swab** Swap each pair of input bytes.

**sync** Pad input blocks to **ibs** with trailing zeros.

**count=***n C*opy only *n* input blocks.

**ibs=***n* Set input block size to *n* bytes (default is 512).

**if=***file* Read input from *file* (default is standard input).

**obs=***n* Set output block size to *n* bytes (default is 512).

**of=***file* Write output to *file* (default is standard output).

**seek=***n* Skip *n* output-sized blocks from start of output file.

**skip=***n* Skip *n* input-sized blocks from start of input file.

You can multiply size values (*n*) by a factor of 1024, 512, or 2 by appending the letter **k**, **b**, or **w**, respectively. You can use the letter **x** as a multiplication operator between two numbers.

**Examples**

Convert an input file to all lowercase:

**dd if=caps\_file of=small\_file conv=lcase**

Retrieve variable-length data; write it as fixed-length to **out**:

*data\_retrieval\_cmd* **| dd of=out conv=sync,blockÂ df** [*options*] [*name*]

Report the amount of free disk space available on all mounted filesystems or on the given *name*. (**df** cannot report on unmounted filesystems.) Disk space is shown in 1KB blocks (default) or 512-byte blocks (if the environment variable POSIXLY\_CORRECT is set). *name* can be a device name (e.g., */dev/hd\**), the directory name of a mounting point (e.g., */usr*), or a directory name (in which case **df** reports on the entire filesystem in which that directory is mounted).

**Options**

**-a**, **- -all** Include empty filesystems (those with 0 blocks).

**- -block-size=***n*  Show space as *n*-byte blocks.

**-h**, **- -human-readable** Print sizes in a format friendly to human readers (e.g., 1.9G instead of 1967156).

**-H**, **- -si** Like **-h**, but show as power of 1000 rather than 1024.

**-i**, **- -inodes** Report free, used, and percent-used inodes.

**-k**, **- -kilobytes** Print sizes in kilobytes.

**-l**, **- -local** Show local filesystems only.

**-m**, **- -megabytes** Print sizes in megabytes.

**- -no-sync** Show results without invoking **sync** first (i.e., without flushing the buffers). This is the default.

**-P**, **- -portability** Use POSIX output format (i.e., print information about each filesystem on exactly

one line).

**- -sync** Invoke **sync** (flush buffers) before getting and showing sizes.

**-t** *type*, **- -type=***type* Show only *type* filesystems.

**-T**, **- -print-type** Print the type of each filesystem in addition to the sizes.

**-x** *type*, **- -exclude-type=***type*

Show only filesystems that are not of type *type*.

**dhclient**  - Dynamic Host Configuration Protocol Client

SYNTAX

**dhclient [ -p port ] [ -d ] [ -q ] [ -1 ] [ -r ] [ -lf lease-file ] [ -pf pid-file ] [ -cf config-file ]**

**[ -sf script-file ] [ -s server ] [ -g relay ] [ -n ] [ -nw ] [ -w ] [ if0 [ ...ifN ] ]**

DESCRIPTION

The Internet Software Consortium DHCP Client, dhclient, provides a means for configuring one or more network interfaces using the Dynamic Host Configuration Protocol, BOOTP protocol, or if these protocols fail, by statically assigning an address.

OPERATION

The DHCP protocol allows a host to contact a central server which maintains a list of IP addresses which

may be assigned on one or more subnets. A DHCP client may request an address from this pool, and then use it on a temporary basis for communication on network. The DHCP protocol also provides a

mechanism whereby a client can learn important details about the network to which it is attached, such as the location of a default router, the location of a name server, and so on.

On startup, dhclient reads the dhclient.conf for configuration instructions. It then gets a list of all the network interfaces that are configured in the current system. For each interface, it attempts to configure the interface using the DHCP protocol.

In order to keep track of leases across system reboots and server restarts, dhclient keeps a list of leases it has been assigned in the dhclient.leases file. On startup, after reading the dhclient.conf file, dhclient reads the dhclient.leases file to refresh its memory about what leases it has been assigned.

When a new lease is acquired, it is appended to the end of the dhclient.leases file. In order to prevent the file from becoming arbitrarily large, from time to time dhclient creates a new dhclient.leases file from its in-core lease database. The old version of the dhclient.leases file is retained under the name dhclient.leases~ until the next time dhclient rewrites the database.

Old leases are kept around in case the DHCP server is unavailable when dhclient is first invoked

(generally during the initial system boot process). In that event, old leases from the dhclient.leases file

which have not yet expired are tested, and if they are determined to be valid, they are used until either they expire or the DHCP server becomes available.

A mobile host which may sometimes need to access a network on which no DHCP server exists may be

preloaded with a lease for a fixed address on that network. When all attempts to contact a DHCP server

have failed, dhclient will try to validate the static lease, and if it succeeds, will use that lease until it is

restarted.

COMMAND LINE

The names of the network interfaces that dhclient should attempt to configure may be specified on the

command line. If no interface names are specified on the command line dhclient will normally identify all network interfaces, eliminating non-broadcast interfaces if possible, and attempt to configure each interface.

It is also possible to specify interfaces by name in the dhclient.conf file. If interfaces are specified in this way, then the client will only configure interfaces that are either specified in the configuration file or on the command line, and will ignore all other interfaces.

If the DHCP client should listen and transmit on a port other than the standard (port 68), the -p flag may

used. It should be followed by the udp port number that dhclient should use. This is mostly useful for

debugging purposes. If a different port is specified for the client to listen on and transmit on, the client will also use a different destination port - one greater than the specified destination port.

The DHCP client normally transmits any protocol messages it sends before acquiring an IP address to, 255.255.255.255, the IP limited broadcast address. For debugging purposes, it may be useful to have the server transmit these messages to some other address. This can be specified with the -s flag, followed by the IP address or domain name of the destination.

The DHCP client will normally run in the foreground until it has configured an interface, and then will

revert to running in the background. To run force dhclient to always run as a foreground process, the -d

flag should be specified. This is useful when running the client under a debugger, or when running it out of inittab on System V systems.

The client normally prints a startup message and displays the protocol sequence to the standard error

descriptor until it has acquired an address, and then only logs messages using the syslog facility.

The -q flag prevents any messages other than errors from being printed to the standard error descriptor.

The client normally doesn’t release the current lease as it is not required by the DHCP protocol. Some

cable ISPs require their clients to notify the server if they wish to release an assigned IP address. The -r flag explicitly releases the current lease, and once the lease has been released, the client exits.

The -1 flag causes dhclient to try once to get a lease. If it fails, dhclient exits with exit code two.

The DHCP client normally gets its configuration information from /etc/dhclient.conf, its lease database from /var/lib/dhcp/dhclient.leases, stores its process ID in a file called /var/run/dhclient.pid, and configures the network interface using /sbin/dhclient-script. To specify different names and/or locations for these files, use the -cf, -lf, -pf and -sf flags, respectively, followed by the name of the file. This can be particularly useful if, for example, /var/lib/dhcp or /var/run has not yet been mounted when the DHCP client is started.

The DHCP client can be directed not to attempt to configure any interfaces using the -n flag. This is most likely to be useful in combination with the -w flag.

The client can also be instructed to become a daemon immediately, rather than waiting until it has

acquired an IP address. This can be done by supplying the --nw flag.

THE CONTROL OBJECT

The control object allows you to shut the client down, releasing all leases that it holds and deleting any DNS records it may have added. It also allows you to pause the client - this unconfigures any interfaces the client is using. You can then restart it, which causes it to reconfigure those interfaces. You would normally pause the client prior to going into hibernation or sleep on a laptop computer. You would then resume it after the power comes back. This allows PC cards to be shut down while the computer is hibernating or sleeping, and then reinitialized to their previous state once the computer comes out of hibernation or sleep.

The control object has one attribute - the state attribute. To shut the client down, set its state attribute to 2. It will automatically do a DHCPRELEASE. To pause it, set its state attribute to 3. To resume it, set its state attribute to 4.

FILES

/sbin/dhclient-script, /etc/dhclient.conf,

/var/lib/dhcp/dhclient.leases, /var/run/dhclient.pid,

/var/lib/dhcp/dhclient.leases~.

On some Linux Distributions such as Debian Linux, the **pump** command is used to request configuration information from a DHCP server.**dmesg [*options*]**

System administration command. Display the system control messages from the kernel ring buffer. This buffer stores all messages since the last system boot or the most recent ones, if the buffer has been filled.

**Options**

**-c** Clear buffer after printing messages.

**-n** *level* Set the level of system message that will display on console.

**du**  Estimates file space usage, reports the amount of disk space used by the specified files and for each subdirectory.

SYNTAX

**du [OPTION]... [FILE]...**

DESCRIPTION

With no arguments, du reports the disk space for the current directory. Normally the disk space is printed in units of 1024 bytes, but this can be overridden

OPTIONS

-a **- -**all Show counts for all files, not just directories.

-b - -bytes Print sizes in bytes, overriding the default block size (\*note Block size::).

-h **- -**human-readable

Append a size letter such as ‘K’ or M. Powers of 1024 are used, not 1000.

-k **- -**kilobytes Print sizes in 1024-byte blocks, overriding the default block size

-m - -megabytes Print sizes in megabyte (that is, 1,048,576-byte) blocks.

**- -**separate-dirs Report the size of each directory separately, not including the sizes of

subdirectories.

**e2fsck** [options] device

DESCRIPTION

**CAUTION: The filesystem being checked by e2fsck must be unmounted. If this is not possible, boot to a Linux rescue disk or bootable CDROM Linux such as Knoppix first.** e2fsck is used to check a Linux second extended file system (ext2fs). E2fsck also supports ext2 filesystems containing a journal, which are also sometimes known as ext3 filesystems, by first applying the journal to the filesystem before continuing with normal e2fsck processing. After the journal has been applied, a filesystem will normally be marked as clean. Hence, for ext3 filesystems, e2fsck will normally run the journal and exit, unless its superblock indicates that further checking is required. device is the device file where the filesystem is stored (e.g. /dev/hdc1).

OPTIONS:

-p Automatic repair (no questions)

-n Make no changes to the filesystem

-y Assume "yes" to all questions

-c Check for bad blocks

-f Force checking even if filesystem is marked clean

-v Be verbose

-b superblock Use alternative superblock

-B blocksize Force blocksize when looking for superblock

-j external-journal Set location of the external journal

-l bad\_blocks\_file Add to badblocks list

-L bad\_blocks\_file Set badblocks list

**env** [*option*] [*variable***=***value ...* ] [*command*]

DESCRIPTION

Display the current environment (environmental variables) or, if an environment *variable* is specified, set it to a new *value* and display the modified environment. If *command* is specified, execute it under the modified environment.

1. OPTIONS

**-, -i, - -ignore-environment** Ignore current environment entirely.

**-u** *name*, **- -unset** *name* Unset the specified variable.

**ethtool** Display or change ethernet card settings

SYNTAX

**ethtool [options..] *ethX***

* + - 1. DESCRIPTION

ethtool is used for querying settings of an ethernet device and changing them. *ethX* is the name of the ethernet device to work on.

1. OPTIONS

ethtool with a single argument specifying the device name prints current setting of the specified device.

-h shows a short help message.

-a queries the specified ethernet device for pause parameter information.

autoneg on|off Specify if autonegotiation is enabled.

rx on|off Specify if RX is enabled.

tx on|off Specify if TX is enabled.

-i queries the specified Ethernet device for associated driver information.

-d retrieves and prints a register dump for the specified Ethernet device.

-e retrieves and prints an EEPROM dump for the specified Ethernet device.

-r restarts auto-negotiation on the specified Ethernet device, if auto-negotiation is

enabled.

-S queries the specified ethernet device for NIC- and driver-specific statistics.

-s option allows changing some or all settings of the specified Ethernet device. All following options only apply if -s was specified.

**speed 10|100|1000** Set speed in Mb/s. ethtool with single argument will show you the supported device speeds.

**duplex half|full** Set full or half duplex mode.

**port tp|aui|bnc|mii** Select device port.

**autoneg on|off** Specify if autonegotiation is enabled. In the usual case it is, but

might cause some problems with some network devices, so you can turn it off.

-t executes adapter self test on the specified ethernet device. Possible test modes are: offline|online, defines test type: offline (default) means to perform full set of tests possibly causing normal operation interruption during the tests, online means to perform limited set of tests do not interrupt normal adapter operation.

**fdisk** Partition table manipulator for Linux. Hard disks can be divided into one or more logical disks called

partitions. This division is described in the partition table found in sector 0 of the disk.

SYNTAX

**fdisk** device

DESCRIPTION

Linux needs at least one partition, namely for its root file system. It can use swap files and/or swap partitions, but the latter are more efficient. So, usually one will want a second Linux partition dedicated as swap partition. On Intel hardware, the BIOS can often only access the first 1024 cylinders of the disk. For this reason people with large disks often create a third partition, just a few MB large, typically mounted on /boot, to store the kernel image and a few auxiliary files needed at boot time, so as to make sure that this stuff is accessible to the BIOS. There may be reasons of security, ease of administration and backup, or testing, to use more than the minimum number of partitions.

**fdisk device**: a menu driven program for creation and manipulation of partition tables.

The device is usually one of the following:

**/dev/had /dev/hdb /dev/sda /dev/sdb**

(/dev/hd[a-h] for IDE disks, /dev/sd[a-p] for SCSI disks, device name refers to the entire disk.

The partition is a device name followed by a partition number. For example, /dev/hda1 is the first partition on the first IDE hard disk in the system. IDE disks can have up to 63 partitions, SCSI disks up to 15. A DOS type partition table can describe an unlimited number of partitions. In sector 0 there is room for the description of 4 partitions (called primary). One of these may be an extended partition; this is a box holding logical partitions with descriptors found in a linked list of sectors, each preceding the corresponding logical partitions. The four primary partitions, present or not, get numbers 1-4. Logical partitions start numbering from 5.

**file [*options*] *files*** Classify the named *files* according to the type of data they contain. **file** checks the magic file (usually */usr/share/magic*) to identify some file types.

**Options**

**-b** Brief mode; do not prepend filenames to output lines.

**-c** Check the format of the magic file (*files* argument is invalid with **-c**). Usually used with **-m**.

**-f** *file* Read the names of files to be checked from *file*.

**-L** Follow symbolic links. By default, symbolic links are not followed.

**-m** *file* Search for file types in *file* instead of */usr/share/magic*.

**-n** Flush standard output after checking a file.

**-s** Check files that are block or character special files in addition to checking ordinary files.

**-v** Print the version.

**-z** Attempt checking of compressed files.

Many file types are understood. Output lists each filename, followed by a brief classification such as:

ascii text c program text c-shell commands data

empty iAPX 386 executable directory [nt]roff, tbl

shell cmds symbolic link

**Example**

List all files that are deemed to be troff/nroff input:

**file \* | grep roff**

**find** [*pathnames*] [*conditions*]

An extremely useful command for finding particular groups of files (numerous examples follow this description). **find** descends the directory tree beginning at each *pathname* and locates files that meet the specified *conditions*. The default pathname is the current directory. The most useful conditions include **-print** (which is the default if no other expression is given), **-name** and **-type** (for general use), **-exec** and **-size** (for advanced users), and **-mtime** and **-user** (for administrators).

Conditions may be grouped by enclosing them in **\(** **\)** (escaped parentheses), negated with **!** (use **\!** in the C shell), given as alternatives by separating them with **-o**, or repeated (adding restrictions to the match; usually only for **-name**, **-type**, **-perm**). Modification refers to editing of a files contents. Change refers to modification, permission or ownership changes, and so on; therefore, for example, **-ctime** is more inclusive than **-atime** or **-mtime**.

**Conditions and actions**

**-atime** **+***n* | **-***n* | *n*

Find files that were last accessed more than *n* (**+***n*), less than *n* (**-***n*), or exactly *n* days ago. Note that **find** changes the access time of directories supplied as *pathnames*.

**-ctime** **+***n* | **-***n* | *n*

Find files that were changed more than *n* (**+***n*), less than *n* (**-***n*), or exactly *n* days ago. A change is anything that changes the directory entry for the file, such as a **chmod**.

**-exec** *command* **{ } \;**

Run the Linux *command*, from the starting directory on each file matched by **find** (provided *command* executes successfully on that file; i.e., returns a 0 exit status). When *command* runs, the argument **{ }** substitutes the current file. Follow the entire sequence with an escaped semicolon (**\;**).

**-group** *gname*

Find files belonging to group *gname*. *gname* can be a group name or group ID number.

**-inum** *n* Find files whose inode number is *n*.

**-links** *n* Find files having *n* links.

**-mtime** **+***n* | **-***n* | *n*

Find files that were last modified more than *n* (**+***n*), less than *n* (**-***n*), or exactly *n* days ago. A modification is a change to a files data.

**-name** *pattern*

Find files whose names match *pattern*. Filename metacharacters may be used but should be escaped or quoted.

**-perm** *nnn* Find files whose permission flags (e.g., **rwx**) match octal number *nnn* exactly (e.g., 664 matches **-rw-rw-r--**). Use a minus sign before *nnn* to make a "wildcard" match of any unspecified octal digit (e.g., **-perm -600** matches **-rw-\*\*\*\*\*\***, where \* can be any mode).

**-size** *n*[**c**] Find files containing *n* blocks, or if **c** is specified, *n* characters long.

**-type** *c* Find files whose type is *c*. *c* can be **b** (**b**lock special file), **c** (**c**haracter special file), **d** (**d**irectory), **p** (fifo or named **p**ipe), **l** (symbolic **l**ink), **s** (**s**ocket), or **f** (plain **f**ile).

**-user** *user* Find files belonging to *user* (name or ID).

**-maxdepth** *num*

Do not descend more than *num* levels of directories.

**-amin +***n* | **-***n* | *n*

Find files last accessed more than *n* (+*n*), less than *n* (**-***n*), or exactly *n* minutes ago.

**-cmin +***n* | **-***n* | *n*

Find files last changed more than *n* (+*n*), less than *n* (**-***n*), or exactly *n* minutes ago.

**-fstype** *type*

Match files only on *type* filesystems. Acceptable types include **minix**, **ext**, **ext2**, **xia**,

**msdos**, **umsdos**, **vfat**, **proc**, **nfs**, **iso9660**, **hpfs**, **sysv**, **smb**, and **ncpfs**.

**-gid** *num* Find files with numeric group ID of *num*.

**-ilname** *pattern*

A case-insensitive version of **-lname**.

**-iname** *pattern*

A case-insensitive version of **-name**.

**-lname** *pattern*

Search for files that are symbolic links, pointing to files named *pattern*. *pattern* can

include shell metacharacters and does not treat **/** or **.** specially. The match is case-

insensitive.

**-mmin +***n* | **-***n* | *n*

Find files last modified more than *n* (+*n*), less than *n* (**-***n*), or exactly *n* minutes ago.

**-nouser** The files user ID does not correspond to any user.

**-nogroup** The files group ID does not correspond to any group.

**Examples**

List all files (and subdirectories) in your home directory:

**find $HOME**

List all files named *chapter1* in the */work* directory:

**find /work -name chapter1**

List all files beginning with *memo* owned by *ann*:

**find /work -name memo\* -user ann**

Search the filesystem (begin at root) for manpage directories:

**find / -type d -name man\***

Search the current directory, look for filenames that *don’t* begin with a capital letter, and send them to the printer:

**find . \! -name [A-Z]\* -exec lpr \;**

Find and compress files whose names *dont* end with *.gz*:

**gzip find . \! -name \*.gz -print**

Remove all empty files on the system (prompting first):

**find / -size 0 -ok rm \;**

Search the system for files that were modified within the last two days (candidates for backing up):

**find / -mtime -2 -print**

Recursively **grep** for a pattern down a directory tree:

**find /book -print | xargs grep [Nn]utshell**

If the files *kt1* and *kt2* exist in the current directory, their names can be printed with the command:

$ **find . -name kt[0-9]**

./kt1

./kt2

**ftp** [*options*] [*hostname*]

Transfer files to and from remote network site *hostname*. **ftp** prompts the user for a command. The commands are listed after the options. Some of the commands are toggles, meaning they turn on a feature when it is off and vice versa.

**Options**

**-d** Enable debugging.

**-g** Disable filename globbing.

**-I** Turn off interactive prompting.

**-n** No autologin upon initial connection.

**-v** Verbose. Show all responses from remote server.

**Commands**

**!**[*command* [*args*]]

Invoke an interactive shell on the local machine. If arguments are given, the first is taken as a command to execute directly, with the rest of the arguments as that commands arguments.

**ascii** Set the file transfer type to network ASCII (default).

**bell** Sound a bell after each file transfer command is completed.

**binary** Set file transfer type to support binary image transfer.

**bye** Terminate FTP session and then exit **ftp**.

**case** Toggle remote computer filename case mapping during **mget**. The default is off. When **case** is on, files on the remote machine with all-uppercase names will be copied to the local machine with all-lowercase names.

**cd** *remote-directory*

Change working directory on remote machine to *remote-directory*.

**chmod** [*mode*] [*remote-file*]

Change file permissions of *remote-file*. If options are omitted, the command prompts for them.

**close** Terminate FTP session and return to command interpreter.

**cr** Toggle carriage return stripping during ASCII-type file retrieval.

**delete** *remote-file*

Delete file *remote-file* on remote machine.

**dir** [*remote-directory*] [*local-file*]

Print a listing of the contents in the directory *remote-directory*, and, optionally, place the output in *local-file*. If no directory is specified, the current working directory on the remote machine is used.

**disconnect**

Synonym for **close**.

**form** *format*

Set the file transfer form to *format*. Default format is *file*.

**get** *remote-file* [*local-file*]

Retrieve the *remote-file* and store it on the local machine. If the local filename is not specified, it is given the same name it has on the remote machine, subject to alteration by the current **case**, **ntrans**, and **nmap** settings. If local file is -, output comes to the terminal.

**hash** Toggle hash-sign (#) printing for each data block transferred.

**help** [*command*] or **?**

Print help information for *command*. With no argument, **ftp** prints a list of commands.

**lcd** [*directory*]

Change working directory on local machine. If *directory* is not specified, the users home directory is used.

**ls** [*remote-directory*] [*local-file*]

Print listing of contents of directory on remote machine, in a format chosen by the remote machine. If *remote-directory* is not specified, current working directory is used.

**mdelete** *remote-files*

Delete the *remote-files* on the remote machine.

**mdir** *remote-files local-file*

Like **dir**, except multiple remote files may be specified.

**mget** *remote-files*

Expand the wildcard expression *remote-files* on the remote machine and do a **get** for each filename thus produced.

**mkdir** *directory-name*

Make a directory on the remote machine.

**mode** [*mode-name*]

Set file transfer mode to *mode-name*. Default mode is stream mode.

**modtime** [*file-name*]

Show last modification time of the file on the remote machine.

**mput** [*local-files*]

Expand wildcards in *local-files* given as arguments and do a **put** for each file in the resulting list.

**newer** *remote-file* [*local-file*]

Get file if remote file is newer than local file.

**ntrans** [*inchars* [*outchars* ]]

Set or unset the filename character translation mechanism. Characters in a filename matching a

character in *inchars* are replaced with the corresponding character in *outchars*. If no arguments

are specified, the filename mapping mechanism is unset. If arguments are specified:

* Characters in remote filenames are translated during **mput** and **put** commands issued without a specified remote target filename.
* Characters in local filenames are translated during **mget** and **get** commands issued without a specified local target filename.

**open** *host* [*port*]

Establish a connection to the specified *host* FTP server. An optional *port* number may be supplied, in which case **ftp** will attempt to contact an FTP server at that port.

**prompt**

Toggle interactive prompting.

**put** *local-file* [*remote-file*]

Store a local file on the remote machine. If *remote-file* is left unspecified, the local filename is used after processing according to any **ntrans** or **nmap** settings in naming the remote file. File transfer uses the current settings for *type*, *file*, *structure*, and *transfer mode*.

**pwd** Print name of the current working directory on the remote machine.

**quit** Synonym for **bye**.

**reget** *remote-file* [*local-file*]

Retrieve a file (like **get**), but restart at the end of *local-file*. Useful for restarting a dropped transfer.

**rename** [*from*] [*to*]

Rename file *from* on remote machine to *to*.

**reset** Clear reply queue.

**rmdir** [*directory-name*]

Delete a directory on the remote machine.

**site** [*command*]

Get/set site-specific information from/on remote machine.

**size** *filename*

Return size of *filename* on remote machine.

**status** Show current status of **ftp**.

**struct** [*struct-name*]

Set the file transfer structure to *struct-name*. By default, **stream** structure is used.

**system** Show type of operating system running on remote machine.

**trace** Toggle packet tracing.

**type** [*type-name*]

Set file transfer **type** to *type-name*. If no type is specified, the current type is printed. The default type is network ASCII.

**user** *username* [*password*] [*account*]

Identify yourself to the remote FTP server. **ftp** will prompt the user for the password, if not specified and the server requires it, and the account field.

**verbose**

Toggle verbose mode.

**grep** Search file(s) for specific text.

SYNTAX

**grep <options> "Search String" [filename]**

**grep <options> [-f FILE] [FILE...]**

A simple example: **$grep “Needle in a Haystack” /etc/\***

**OPTIONS**

**-c - -count**

Suppress normal output; instead print a count of matching lines for each input file. With the -v,

- -invert-match option, count non-matching lines.

**-i - -ignore-case**

Ignore case distinctions in both the pattern and the input files.

**-l - -files-with-matches**

Suppress normal output; instead print the name of each input file from which output would normally

have been printed. The scanning of every file will stop on the first match.

**-n - -line-number**

Prefix each line of output with the line number within its input file.

**-r - -recursive**

(GNU Extension) For each directory mentioned in the command line, read and process all files in that directory, recursively. This is the same as the -d recurse option.

**-v - -invert-match**

Invert the sense of matching, to select non-matching lines.

**-w - -word**

(GNU Extension) Select only those lines containing matches that form whole words. The test is that the matching substring must either be at the beginning of the line, or preceded by a non-word constituent character. Similarly, it must be either at the end of the line or followed by a non-word constituent character. Word-constituent characters are letters, digits, and the underscore.

**-x - -line**

Select only those matches that exactly match the whole line.

**Diagnostics**

Normally, exit status is 0 if matches were found, and 1 if no matches were found (the -v option inverts the sense of the exit status). Exit status is 2 if there were syntax errors in the pattern, inaccessible input files, or other system errors.

**groupadd** [*options*] *group*

System administration command. Create new group account *group*.

**Options**

**-g** *gid* Assign numerical group ID. (By default, the first available number above 500 is used.) The value

must be unique unless the **-o** option is used.

**-o** Accept a nonunique *gid* with the **-g** option.

**groupdel** *group*

System administration command. Remove *group* from system account files. You may still need to find and change permissions on files that belong to the removed group.

**groupmod** [*options*] *group*

System administration command. Modify group information for *group*.

**Options**

**-g** *gid* Change the numerical value of the group ID. Any files that have the old *gid* will have to be

changed manually. The new *gid* must be unique unless the **-o** option is used.

**-n** *name* Change the group name to *name*.

**-o** Override. Accept a nonunique *gid*.

**groups** Displays group names a user belongs to

SYNTAX

**groups [USERNAME]...**

DESCRIPTION  
 Prints the names of the primary and any supplementary groups for each given USERNAME, or the current process if no names are given. If names are given, the name of each user is printed before the list

of that users groups.

**grub-install** *boot-device*

DESCRIPTION

grub-install must be run by the root user (super user). The boot-device is the drive that contains the Master

Boot Record (MBR). For example: # grub-install /dev/hda.

USAGE

If a new OS such as Windows XP is installed after Linux, the MBR is modified and Linux will no longer be a boot option. You can use loadlin.exe and vmlinuz to boot into Linux and use grub-install to restore the Grub Boot Loader.

**hostname** Print or set system name

SYNTAX

**hostname [NAME]**

DESCRIPTION

With no arguments, hostname prints the name of the current host system. With one argument, it sets the current host name to the specified string. You must have appropriate privileges to set the host name.

**id** Print real and effective user id (uid) and group id (gid), prints information about the given user, or if no

user is specified the process running it

SYNTAX

**id [OPTION]... [USERNAME]**

DESCRIPTION

By default, it prints the real user id, real group id, effective user id if different from the real user id, effective group id if different from the real group id, and supplemental group ids. Each of these numeric values is preceded by an identifying string and followed by the corresponding user or group name in parentheses.

OPTIONS

The options cause id to print only part of the above information.

**-g - -group** Print only the group id.

**-G --groups**  Print only the supplementary groups.

**-n - -name** Print the user or group name instead of the ID number. Requires -u, -g, or -G

**-r --real** Print the real, instead of effective, user or group id. Requires -u, -g, or -G

**-u - -user** Print only the user id.

**ifconfig** [*interface | interface* *address\_family* *parameters* *addresses*]

TCP/IP command. Assign an address to a network interface and/or configure network interface parameters. ifconfig is typically used at boot time to define the network address of each interface on a machine. It may be used at a later time to redefine an interfaces address or other parameters. Without arguments, ifconfig displays the current configuration for a network interface. Used with a single *interface* argument, ifconfig displays that particular interfaces current configuration.

**Arguments**

*interface* String of the form *name unit*, for example, **eth0**.

*address\_family*

Since an interface may receive transmissions in differing protocols, each of which may require separate naming schemes, you can specify the *address\_family* to change the interpretation of the remaining parameters. You may specify **inet** (the default; for TCP/IP), **ax25** (AX.25 Packet Radio), **ddp** (Appletalk Phase 2), or **ipx** (Novell).

***Parameters***

The following parameters may be set with **ifconfig**:

**dest\_address**

Specify the address of the correspondent on the other end of a point-to-point link.

**down** Mark an interface "down" (unresponsive).

**irq** *addr* Set the devices interrupt line.

**metric** *n* Set routing metric of the interface to *n*. Default is 0.

**mtu** *num* Set the interfaces Maximum Transfer Unit (MTU).

**multicast** Set the multicast flag.

**netmask** *mask*

(**inet** only.) Specify how much of the address to reserve for subdividing networks into

subnetworks. *mask* can be specified as a single hexadecimal number with a leading 0x, with

a dot notation Internet address, or with a pseudonetwork name listed in the network table

*/etc/networks*.

**pointopoint/-pointopoint** [*address*]

Enable/disable point-to-point interfacing, so that the connection between the two machines

is dedicated.

**up** Mark an interface "up" (ready to send and receive).

***address***

Either a hostname present in the hostname database (*/etc/hosts*), or an Internet address expressed in the

Internet standard dot notation.

**info** Display helpful information about commands.

SYNTAX

**info [command]**

**help [-s] [command]**

**man -k [command]**

OPTIONS

-s restrict the information displayed to a short usage synopsis.

-k Search by command description rather than command name.

Press <Space bar> to view the next page. Press <return> to view next line

Press Q to exit

**init** System administration command that changes the current run level

SYNTAX

**init [*option*] [*runlevel*]**

.

1. OPTION

**-t *seconds*** *W*hen changing runlevels, send SIGKILL *seconds* after SIGTERM. Default is 20.

1. Files

init is the first process run by any Unix machine at boot time. It verifies the integrity of all filesystems and then creates other processes, using fork and exec, as specified by */etc/inittab*. Which processes may be run are controlled by *runlevel*. All process terminations are recorded in */var/run/utmp* and */var/log/wtmp*. When the runlevel changes, init sends SIGTERM and then, after 20 seconds, SIGKILL to all processes that cannot be run in the new runlevel.

1. Runlevels

The current runlevel may be changed by telinit, which is often just a link to init. The default runlevels vary from

distribution to distribution, but these are standard for most Linux systems:

**0** Halt the system.

**1, s, S** Single-User mode.

**2** Multi-User withour Networking

**3** Multi-User mode

**5** X11, X-Windows mode

**6** Reboot the system.

**ipcalc** - perform simple manipulation of IP addresses

SYNTAX

**ipcalc [OPTION]... <IP address>[/prefix] [netmask]**

DESCRIPTION

ipcalc provides a simple way to calculate IP information for a host. The various options specify what

information ipcalc should display on standard out. Multiple options may be specified. An IP address to operate on must always be specified. Most operations also require a netmask or a CIDR prefix as well.

OPTIONS

**-b, - -broadcast** Display the broadcast address for the given IP address and netmask.

**-h, - -hostname**  Display the hostname for the given IP address.

**-m, - - netmask**  Calculate the netmask for the given IP address. It assumes that the IP address is in a complete class A, B, or C network. Many networks do not use the default net- masks, in which case an inappropriate value will be returned.

**-p, - -prefix** Show the prefix for the given mask/IP address.

**-n, - -network**  Display the network address for the given IP address and netmask.

**-s, - -silent**  Don’t ever display error messages.

**kill** Kill a process (job).

SYNTAX

**kill [-s sigspec] [-n signum] [-sigspec] jobspec or pid**

**kill -l [exit\_status]**

KEY

**-l** : List the signal names

**-s** : Send a specific signal

**-n** : Send a specific signal number

Send a signal specified by sigspec or signum to the process named by job specification jobspec or process ID pid. sigspec is either a signal name such as SIGINT (with or without the SIG prefix) or a signal number; signum is a signal number. If sigspec and signum are not present, SIGTERM is used (Terminate). If any arguments are supplied when -l is given, the names of the signals corresponding to the arguments are listed, and the return status is zero. exit\_status is a number specifying a signal number or the exit status of a process terminated by a signal. The return status is zero if at least one signal was successfully sent, or non-zero if an error occurs or an invalid option is encountered.

List the running process

$ **ps**

PID TTY TIME CMD

1293 pts/5 00:00:00 MyProgram

Then Kill it

$ **kill 1293**

[2]+ Terminated MyProgram

To run a command and then kill it after 5 seconds:

**my\_command & sleep 5**

**kill -0 $! && kill $!**

**killall** Kill processes by command name. If more than one process is running the specified command, kill

all of them. Treat command names that contain a **/** as files; kill all processes that are executing that

file.

SYNTAX

**killall [*options*] *names***

1. OPTIONS

**-***signal* Send *signal* to process (default is TERM). *signal* may be a name or number.

**-e** Require an exact match to kill very long names (i.e., longer than 15 characters). Normally, **killall** kills everything that matches within the first 15 characters. With **-e**, such entries are skipped. (Use **-v** to

print a message for each skipped entry.)

**-g** Kill the process group to which the process belongs.

**-i** Prompt for confirmation before killing processes.

**-l** List known signal names.

**-q** Quiet; do not complain of processes not killed.

**-v** Verbose: after killing process, report success and process ID.

**-w** Wait for all killed processes to die. Note that **killall** may wait forever if the signal was ignored or had

no effect, or if the process stays in zombie state.

**kudzu** detects and configures new and/or changed hardware on a system

SYNTAX

kudzu [options]

DESCRIPTION

kudzu detects and configures new and/or changed hardware on a system. When started, kudzu detects the current hardware, and checks it against a database stored in /etc/sysconfig/hwconf, if one exists. It then determines if any hardware has been added or removed from the system. If so, it gives the users the opportunity to configure any added hardware, and unconfigure any removed hardware. It then updates the database in /etc/sysconfig/hwconf.

If no previous database exists, kudzu attempts to determine what devices have already been configured, by looking at /etc/modules.conf, /etc/sysconfig/network-scripts/, and /etc/X11/XF86Config.

OPTIONS

**- -usage** Show short usage message.

**- -help, -?** Print help information.

**-q, - -quiet** Run ’quietly’; do only configuration that doesn’t require user input.

**-s, - -safe** Do only ’safe’ probes that won’t disturb hardware. Currently, this disables the serial

probe, the DDC monitor probe, and the PS/2 probe.

**-t, - -timeout [seconds]**

This sets the timeout for the initial dialog. If no key is pressed before the timeout

elapses, kudzu exits, and /etc/sysconfig/hwconf is not updated.

**-k, - -kernel [version]**

When determining whether a module exists, use the specified kernel version. (If this is

not set, it defaults to the current kernel version.)

**-b, - -bus [bus]** Only probe on the specified bus.

**-c, - -class [class]**

Only probe for the specified class.

**-f, - -file [file]** Read hardware probe info from file file and do not do an actual probe.

**-p, - -probe** Print probe information to the screen, and do not actually configure or unconfigure any devices.

FILES

/etc/sysconfig/**hwconf**

Listing of current installed hardware.

/etc/sysconfig/**kudzu**

Configuration for the boot-time hardware probe. Set ’SAFE’ to something other than ’no’ to force only safe probes.

/etc/**modules.conf**

Module configuration file.

/etc/sysconfig/network-scripts/**ifcfg-\***

Network interface configuration files.

**less Display output one screen at a time, Search through output.**

SYNTAX

**less [options]**

**<command> | less [options]**

DESCRIPTION

**Searching**

/pattern \* Search forward for (N-th) matching line.

?pattern \* Search backward for (N-th) matching line.

n \* Repeat previous search (for N-th occurrence).

N \* Repeat previous search in reverse direction.

ESC-n \* Repeat previous search, spanning files.

ESC-N \* Repeat previous search, reverse dir. & spanning files.

ESC-u \* Undo (toggle) search highlighting.

---------------------------------------------------

Search patterns may be modified by one or more of:

^N or ! Search for NON-matching lines.

^E or \* Search multiple files (pass thru END OF FILE).

^F or @ Start search at FIRST file (for /) or last file (for ?).

^K Highlight matches, but dont move (KEEP position).

^R Dont use REGULAR EXPRESSIONS.

**ln** Make links between files, by default it makes hard links; with the -s option, it makes symbolic links.

SYNTAX

**ln [OPTION]... TARGET [LINKNAME]**

**ln [OPTION]... TARGET... DIRECTORY**

DESCRIPTION

\* If the last argument names an existing directory, ln creates a link to each TARGET file in that directory, using the TARGET's names. (But see the description of the --no-dereference option below.)

\* If two filenames are given, ln creates a link from the second to the first.

\* If one TARGET is given, ln creates a link to that file in the current directory.

\* It is an error if the last argument is not a directory and more than two files are given. Without -f or -i , ln will not remove an existing file. Use the --backup option to make ln rename existing files.

A "**hard link**" is another name for an existing file; the link and the original are indistinguishable. Technically speaking, they share the same inode, and the inode contains all the information about a file--indeed, it is not incorrect to say that the inode \_is\_ the file. On all existing implementations, you cannot make a hard link to a directory, and hard links cannot cross filesystem boundaries.

"**Symbolic links**" ("symlinks" for short), on the other hand, are a special file type (which not all kernels support: System V release 3 (and older) systems lack symlinks) in which the link file actually refers to a different file, by name. When most operations (opening, reading, writing, and so on) are passed the symbolic link file, the kernel automatically "dereferences" the link and operates on the target of the link. But some operations (e.g., removing) work on the link file itself, rather than on its target.

OPTIONS

**-b - -backup** Make a backup of each file that would otherwise be overwritten or removed.

**-f - -force** Remove existing destination files.

**-I - -interactive** Prompt whether to remove existing destination files.

**-s - -symbolic** Make symbolic links instead of hard links. This option merely produces an error

message on systems that do not support symbolic links.

**-S SUFFIX --suffix=SUFFIX**

Append SUFFIX to each backup file made with -b.

**-v - -verbose** Print the name of each file before linking it.

**-V METHOD - -version-control=METHOD**

Change the type of backups made with -b. The METHOD argument can be numbered (or t), existing (or nil), or never (or simple).

Examples:

**ln -s /some/name # creates link ./name pointing to /some/name**

**ln -s /some/name myname # creates link ./myname pointing to /some/name**

**ln -s a b .. # creates links ../a and ../b pointing to ./a and ./b**

1. **Loadlin.exe** LOADLIN.EXE is an MSDOS program that is used to boot Linux.
2. This handy program by Hans Lermen also passes along kernel options. LOADLIN.EXE is very helpful when you must boot from DOS to properly initialize modems or sound cards to make them work under Linux.

You need to do two things before using LOADLIN.EXE:

**1.** Copy LOADLIN.EXE to a DOS partition (for example, C:\LOADLIN).

**2.** Put a copy of your kernel image (/vmlinuz) on your DOS partition.

Loadlin.exe and vmlinuz can usually be located in the \Dosutils directory of the first installation CDROM for Redhat and Fedora distributions.

For example, type the following from the DOS command line to boot Linux:

**loadlin c:\vmlinuz root=/dev/hda3 ro**

The Linux root partition must be specified in the command line. The ro stands for read-only. When you are first booting a Linux partition, it should be mounted as read-only to prevent data loss.

**locate** find data files, programs, directories, & objects matching your search.

**DESCRIPTION**

**locate** searches a database called **slocate.db** on your hard drive which is updated every night. To manually update the database, login as root & then run: **updatedb**

Note that locate searches by name only. To search by any other attribute, use **find**. To search only for commands, use **whereis**.

Example:

**locate license**

Since you usually receive quite a long list when running locate, you’ll probably want to do this:

**locate <search term> | more** -or- **locate <search term> | less**

**lpc** line printer control program

SYNTAX

**lpc [command [argument ...]]**

DESCRIPTION

Lpc is used by the system administrator to control the operation of the line printer system. For each line printer configured in /etc/printcap, lpc may be used to:

* disable or enable a printer,
* disable or enable a printers spooling queue,
* rearrange the order of jobs in a spooling queue,
* find the status of printers, and their associated spooling queues and printer daemons.

Without any arguments, lpc will prompt for commands from the standard input. If arguments are supplied, lpc interprets the first argument as a command and the remaining arguments as parameters to the command. The standard input may be redirected causing lpc to read commands from file. Commands may be abbreviated;

COMMANDS

**? [command ...] help [command ...]**

Print a short description of each command specified in the argu­ment list, or, if no arguments are given, a list of the recog­nized commands.

**abort { all | printer }**

Terminate an active spooling daemon on the local host immediately and then disable printing (preventing new daemons from being started by lpr) for the specified printers.

**clean { all | printer }**

Remove any temporary files, data files, and control files that cannot be printed (i.e., do not form a complete printer job) from the specified printer queue(s) on the local machine.

**disable { all | printer }**

Turn the specified printer queues off. This prevents new printer jobs from being entered into the queue by lpr.

**down { all | printer } message ...**

Turn the specified printer queue off, disable printing and put message in the printer status file. The

message doesn’t need to be quoted, the remaining arguments are treated like echo(1). This is normally used to take a printer down and let others know why lpq(1) will indicate the printer is down and print the status message).

**enable { all | printer }**

Enable spooling on the local queue for the listed printers. This will allow lpr(1) to put new jobs in the spool queue.

**exit quit** Exit from lpc.

**restart { all | printer }**

Attempt to start a new printer daemon. This is useful when some abnormal condition causes the

daemon to die unexpectedly leaving jobs in the queue. Lpq will report that there is no daemon present when this condition occurs. If the user is the super-user, try to abort the current daemon first (i.e., kill and restart a stuck daemon).

**start { all | printer }**

Enable printing and start a spooling daemon for the listed print­ers.

**status { all | printer }**

Display the status of daemons and queues on the local machine.

**stop { all | printer }**

Stop a spooling daemon after the current job completes and dis­able printing.

**topq printer [ jobnum ... ] [ user ... ]**

Place the jobs in the order listed at the top of the printer queue.

**up { all | printer }**

Enable everything and start a new printer daemon. Undoes the ef­fects of down.

FILES

/etc/printcap printer description file

/var/spool/\* spool directories

/var/spool/\*/lock lock file for queue control

**lpr** off line print - sends a print job to the default system queue.

SYNTAX

**lpr [-Pprinter] [-#num] [-C class] [-J job] [-T title] [-U user] [-i [numcols]]**

**[-1234 font] [-wnum] [-cdfghlnmprstv] [name ...]**

DESCRIPTION

Lpr uses a spooling daemon to print the named files when facilities be­ come available. If no names appear, the standard input is assumed. The following single letter options are used to modify or filter the

default handling of print jobs. The spooling daemon will use the appropriate filters to print the data

accordingly.

OPTIONS

**-l** Use a filter which allows control characters to be printed and suppresses page breaks.

**-p** Use pr to format the files (equivalent to print).

**-P** Force output to a specific printer. Normally, the default print­er is used (site dependent), or the value of the environment variable PRINTER is used.

**-h** Suppress the printing of the burst page.

**-m** Send mail upon completion.

**-r**  Remove the file upon completion of spooling. Can not be used with the -s option, due to security

concerns.

**-s** Use symbolic links. Usually files are copied to the spool direc­tory. The -s option will use symlink(2) to link data files rather than trying to copy them so large files can be printed. This means the files should not be modified or removed until they have been printed.

The remaining options apply to copies, the page display, and headers:

**-#num**

The quantity num is the number of copies desired of each file named. For example,

**lpr -#3 foo.c bar.c more.c**

would result in 3 copies of the file foo.c, followed by 3 copies of the file bar.c, etc.

**cat foo.c bar.c more.c | lpr -#3**

will give three copies of the concatenation of the files.

**-[1234]font**

Specifies a font to be mounted on font position i. The daemon will construct a .railmag file referencing the font pathname.

**-J job**

Job name to print on the burst page. Normally, the first files name is used.

**-T title**

Title name for pr(1), instead of the file name.

**-U user**

User name to print on the burst page, also for accounting purposes. This option is only honored if the real user-id is daemon (or that specified in the printcap file instead of daemon), and is intended for those instances where print filters wish to requeue jobs.

**-i [numcols]**

The output is indented. If the next argument is numeric (numcols), it is used as the number of blanks to be printed be­ fore each line; otherwise, 8 characters are printed.

**-wnum**

Uses num as the page width for pr.

**lprm** Remove jobs from the line printer spooling queue

SYNTAX

**lprm [-Pprinter] [-] [job # ...] [user ...]**

DESCRIPTION

Lprm will remove a job, or jobs, from a printers spool queue. Since the spooling directory is protected from users, using Lprm is normally the only method by which a user may remove a job. The owner of a job is determined by the users login name and host name on the machine where the lpr command was invoked.

Options and arguments:

**-Pprinter**

Specify the queue associated with a specific printer (otherwise the default printer is used).

**-** If a single - is given, lprm will remove all jobs which a user owns. If the super-user employs this flag, the spool queue will be emptied entirely.

**user**

Causes lprm to attempt to remove any jobs queued belonging to that user (or users). This form of

invoking lprm is useful only to the super-user.

**job#**

A user may dequeue an individual job by specifying its job number. This number may be obtained from the lpq(1) program, e.g.

**% lpq -l**

**1st:ken [job #013ucbarpa] (standard input) 100 bytes**

**% lprm 13**

If neither arguments or options are given, lprm will delete the currently active job if it is owned by the user.

**ls** List information about Files, by default the current directory.

SYNTAX

**ls [OPTION]... [FILE]...**

KEY

**-a, - -all** Do not hide entries starting with .

**-A, - -almost-all** Do not list implied . and ..

-B, - -ignore-backups Do not list implied entries ending with ~

-c Sort by change time; with -l: show ctime

-C List entries by columns

- -color[=WHEN] Use color to distinguish file types. WHEN may be never, always or auto

**-d, - -directory**  List directory entries instead of contents

**-F, - -classify** Append indicator (one of \*/=@|) to entries

-G, - -no-group Inhibit display of group information

-h, - -human-readable Print sizes in human readable format (e.g., 1K 234M 2G)

**-i, - -inode**  Print index number of each file

**-l**  Use a long listing format

-n, - -numeric-uid-gid List numeric UIDs and GIDs instead of names

-o Use long listing format without group info

-r, - -reverse Reverse order while sorting

**-R, - -recursive**  List subdirectories recursively

-S Sort by file size

--sort=WORD time -t, version -v, status –c, size -S, extension -X, none –U, access -u, use -u

-t sort by modification time

-T, - -tabsize=COLS assume tab stops at each COLS instead of 8

-u sort by last access time; with -l: show atime

-U do not sort; list entries in directory order

-v sort by version

-w, - -width=COLS assume screen width instead of current value

-x list entries by lines instead of by columns

-X sort alphabetically by entry extension

-1 list one file per line

**minicom** [OPTION]... [configuration]

DESCRIPTION

A terminal program for Linux and other UNIX-like systems. minicom can be run with the -s switch to setup parameters such as TTY (COM) port, BPS, Data Bits, Parity, Stop Bits and Flow Control. These settings can be saved in the default configuration file or under a specified name if you need more than one configuration. These options can also be specified in the MINICOM environment variable.

OPTIONS

**-s,** - -**setup** : enter setup mode (only as root)

-o, - -noinit : do not initialize modem & lockfiles at startup

-l, - -ansi : literal; assume screen uses the IBM-PC character set

-L, - -iso : Ditto, but assume screen uses ISO8859

-w, - -wrap : Linewrap on

-z, - -statline : try to use terminals status line

-c, - -color=on/off : ANSI style color usage on or off

-a, - -attrib=on/off : use reverse or highlight attributes on or off

-t, - -term=TERM : override TERM environment variable

-S, - -script=SCRIPT : run SCRIPT at startup

-d, - -dial=ENTRY : dial ENTRY from the dialing directory

-p, - -ptty=TTYP : connect to pseudo terminal

**-C, - -capturefile=FILE**

: start capturing to FILE

**configuration**  : configuration file to use

**mkbootdisk** - creates a stand-alone boot floppy for the running Linux system

SYNTAX

**mkbootdisk** [- -version] [- -noprompt] [- -verbose] [- -mkinitrdargs *<args>*] **- -device** *<devicefile>* *<kernel>*

DESCRIPTION

**mkbootdisk** creates a boot floppy appropriate for the running system. The boot disk is entirely self-contained,

and includes an initial ramdisk image which loads any necessary SCSI modules for the system. The created

boot disk looks for the root filesystem on the device suggested by /etc/fstab. **The only required argument is the kernel version to put onto the boot floppy.**

The resultant boot disk may also be used to boot from a floppy root filesystem by typing **rescue** at the **lilo:** prompt rather then letting the default image boot.

OPTIONS

**- -device** *<devicefile>*  The boot image is created on *<devicefile>.* *If* **- -device** is not specified, /dev/fd0 is

used.

**- -mkinitrdargs** *<args>* Passes *<args>* *to* **mkinitrd** (which is used to create the initial ramdisk image

for the boot floppy). Any **mkinitrd** arguments may be specified.

**- -noprompt** Normally, **mkbootdisk** instructs the user to insert a floppy and waits for

confirmation before continuing. If **--noprompt** is specified, no prompt is displayed.

**- -verbose** Instructs **mkbootdisk** to talk about what it's doing as it's doing it. Normally, there is

no output from **mkbootdisk**.

**mkdir** Create new folder(s), if they do not already exist.

SYNTAX

**mkdir [OPTION] folder...**

**mkdir "Name with spaces"**

OPTIONS

-m, - -mode=MODE set permission mode (as in [chmod](http://www.ss64.com/bash/chmod.html)), not rwxrwxrwx - umask

-p, - -parents no error if existing, make parent directories as needed

DESCRIPTION

mkdir creates the standard entries . (dot) for the current folder and .. (dot dot) for its parent. Any folder name that includes spaces may appear in some applications as "Name%20with%20spaces"

**mkfs** [*options*] [*fs-options*] *filesys* [*blocks*]

DESCRIPTION

System administration command. Construct a filesystem on a device (such as a hard disk partition). *filesys* is either the name of the device or the mountpoint. **mkfs** is actually a frontend that invokes the appropriate version of **mkfs** according to a filesystem type specified by the **-t** option. For example, a Linux Second Extended Filesystem uses **mkfs.ext2** (which is the same as **mke2fs**); MS-DOS filesystems use **mkfs.msdos**. *fs-options* are options specific to the filesystem type. *blocks* is the size of the filesystem in 1024-byte blocks.

**Options**

**-V** Produce verbose output, including all commands executed to create the specific filesystem.

**-t** *fs-type* Tells **mkfs** what type of file system to construct. choices include: ext2, ext3, msdos, vfat, xfs

**filesystem-specific options**

These options must follow generic options and not be combined with them. Most filesystem builders support these three options:

**-c** Check for bad blocks on the device before building the file system.

**-l** *file* Read the file *file* for the list of bad blocks on the device.

**-v** Produce verbose ouput.

**more** Display output one screen at a time, file perusal filter for crt viewing, Users should realize that [less](http://www.ss64.com/bash/less.html) provides more emulation and extensive enhancements.

SYNTAX

**more [-dlfpcsu] [-num] [+/ pattern] [+ linenum] [file ...]**

OPTIONS

**-num** This option specifies an integer which is the screen size (in lines).

**-d** more will prompt the user with the message "[Press space to continue, q to quit.]" and will display "[Press h for instructions.]" instead of ringing the bell when an illegal key is pressed.

**-l** more usually treats ^L (form feed) as a special character, and will pause after any line that contains a form feed. The -l option will prevent this behavior.

**-f** Causes more to count logical, rather than screen lines (i.e., long lines are not folded).

**-p** Do not scroll. Instead, clear the whole screen and then display the text.

**-c** Do not scroll. Instead, paint each screen from the top, clearing the remainder of each line as it is

displayed.

**-s** Squeeze multiple blank lines into one.

**-u** Suppress underlining.

**+/** The +/ option specifies a string that will be searched for before each file is displayed.

**+num** Start at line number num.

COMMANDS

Interactive commands for more are based on vi(1). Some commands may be preceded by a decimal

number, called k in the descriptions below. In the following descriptions, ^X means control-X.

**h or ?** Help: display a summary of these commands. If you forget all the other commands, remember this one.

**SPACE** Display next k lines of text. Defaults to current screen size.

**RETURN** Display next k lines of text. Defaults to 1. Argument becomes new default.

d or ^D Scroll k lines. Default is current scroll size, initially 11. Argument becomes new default.

**q or Q** Exits the more command.

s Skip forward k lines of text. Defaults to 1.

**f**  Skip forward k screenfulls of text. Defaults to 1.

**b or ^B** Skip backwards k screenfulls of text. Defaults to 1.

Go to place where previous search started.

= Display current line number.

/pattern Search for kth occurrence of regular expression. Defaults to 1.

n Search for kth occurrence of last r.e. Defaults to 1.

! or :! Execute in a subshell

^L Redraw screen

**mount** Mounts a file system

SYNTAX

**mount -a [-fFnrsvw] [-t vfstype]**

**mount [-fnrsvw] [-o options [,...]] device | dir**

**mount [-fnrsvw] [-t vfstype] [-o options] device dir**

**mount [-hV]**

All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at /. These files can be spread out over several devices. The mount command serves to attach the file system found on some device to the big file tree. Conversely, the umount(8) command will detach it again. The standard form of the mount command, is **mount -t type device dir** This tells the kernel to attach the file system found on device (which is of type type) at the directory dir. The previous contents (if any) and owner and mode of dir become invisible, and as long as this file system remains mounted, the pathname dir refers to the root of the file system on device.  
  
Three forms of invocation do not actually mount anything:  
 **mount -h** prints a help message,

**mount -V** prints a version string and

**mount [-t type]** lists all mounted file systems of type type:

Most devices are indicated by a file name (of a block special device), like /dev/sda1, but there are other possibilities. For example, in the case of an NFS mount, device may look like knuth.cwi.nl:/dir. It is possible to indicate a block special device using its volume label or UUID (see the -L and -U options below). The file /etc/fstab (see fstab), may contain lines describing what devices are usually mounted where, using which options.

OPTIONS

**-a**  Mount all filesystems (of the given types) mentioned in fstab.

-f Causes everything to be done except for the actual system call; if its not obvious, this fakes mounting the file system. This option is useful in conjunction with the -v flag to determine what the mount command is trying to do. It can also be used to add entries for devices that were mounted earlier with the -n option.

-h Print a help message.

-L label Mount the partition that has the specified label.

-n Mount without writing in /etc/mtab. This is neces­sary for example when /etc is on a read-only file system.

-o Several -o options can be specified in a comma separated string ... see info for more

async All I/O to the file system should be done asynchronously.

atime Update inode access time for each access. This is the default.

noatime Do not update inode access times on this file system (e.g. for faster

access on the news spool to speed up news servers).

auto Can be mounted with the -a option.

noauto Can only be mounted explicitly.

ro Mount the file system read-only.

rw Mount the file system read-write.

suid Allow set-user-identifier or set-group-iden­tifier bits to take effect.

sync All I/O to the file system should be done synchronously.

**-r** Mount the file system read-only. A synonym is -o ro

**-t vfstype** The argument following the -t is used to indicate the file system type. arguments include

ext2, ext3, swap, msdos, vfat, ntfs.

-U uuid Mount the partition that has the specified uuid.

-V Output version.

-v Verbose mode.

**-w** Mount the file system read/write. This is the default. A synonym is -o rw.

**mv** move or rename files or directories.

SYNTAX

**mv [OPTION]... SOURCE DEST**

**mv [OPTION]... SOURCE... DIRECTORY**

DESCRIPTION  
 If the last argument names an existing directory, mv moves each other given file into a file with the same name in that directory. Otherwise, if only two files are given, it renames the first as the second. It is an error if the last argument is not a directory and more than two files are given.  
  
 mv can move only regular files across filesystems. If a destination file exists but is normally unwritable, standard input is a terminal, and the -f or --force option is not given, mv prompts the user for whether to replace the file. (You might own the file, or have write permission on its directory.) If the response does not begin with y or Y, the file is skipped.

OPTIONS

**-b - -backup** Make a backup of each file that would otherwise be overwritten or removed.

**-f - -force**  Remove existing destination files and never prompt the user.

**-i - -interactive** Prompt whether to overwrite each existing destination file, regardless of its permissions. If the response does not begin with y or Y, the file is skipped.

-S SUFFIX --suffix=SUFFIX

Append SUFFIX to each backup file made with -b. The backup suffix is ~, unless set with SIMPLE\_BACKUP\_SUFFIX.

**-u - -update** Do not move a nondirectory that has an existing destination with the same or newer

modification time.

-v - -verbose Print the name of each file before moving it.

**netstat** [*options*]

DESCRIPTION

TCP/IP command. Show network status. For all active sockets, print the protocol, the number of bytes waiting to be received, the number of bytes to be sent, the port number, the remote address and port, and the state of the socket.

1. OPTIONS

**-a** Show the state of all sockets, not just active ones.

**-c** Display information continuously, refreshing once every second.

**-I** Include statistics for network devices.

**-n** Show network addresses as numbers.

**-o** Include additional information such as username.

**-r** Show routing tables.

**-t** List only TCP sockets.

**-u** List only UDP sockets.

**-w** List only raw sockets.

**-x** List only Unix domain sockets.

**nice** [*option*] [*command* [*arguments*]]

DESCRIPTION

Execute a *command* (with its *arguments*) with lower priority (i.e., be "nice" to other users). With no arguments, **nice** prints the default scheduling priority (niceness). If **nice** is a child process, it prints the parent process's scheduling priority. Niceness has a range of -20 (highest priority) to 19 (lowest priority).

* + - 1. OPTIONS

**-n** *adjustment*, -*adjustment*, **- -adjustment=***adjustment*

Run *command* with niceness incremented by *adjustment* (1-19); default is 10. A privileged user can raise priority by specifying a negative *adjustment* (e.g., -5).

**nl** Number lines and write files, writes each FILE to standard output, with line numbers added to some or all of the lines. If no input file (or - ) is given nl will read from standard input.

SYNTAX

**nl [OPTION]... [FILE]...**

OPTIONS

**-b STYLE - -body-numbering=STYLE**

Select the numbering style for lines in the body section of each logical page. The styles are: a (number all lines), t (number only nonempty lines (default for body)), n (do not number lines (default for header and footer)), pREGEXP (number only lines that contain a match for REGEXP).

**-f STYLE - -footer-numbering=STYLE**

Analogous to --body-numbering.

**-h STYLE - -header-numbering=STYLE**

Analogous to --body-numbering.

-i NUMBER - -page-increment=NUMBER

Increment line numbers by NUMBER (default 1).

-l NUMBER - -join-blank-lines=NUMBER

Consider NUMBER (default 1) consecutive empty lines to be one logical line for numbering, and only

number the last one. Where fewer than NUMBER consecutive empty lines occur, do not number them. An empty line is one that contains no characters, not even spaces or tabs.

**-n FORMAT - -number-format=FORMAT**

Select the line numbering format (default is rn):

ln left justified, no leading zeros;

rn right justified, no leading zeros;

rz right justified, leading zeros.

-p - -no-renumber

Do not reset the line number at the start of a logical page.

-s STRING --number-separator=STRING

Separate the line number from the text line in the output with STRING (default is the TAB character).

**-v NUMBER - -starting-line-number=NUMBER**

Set the initial line number on each logical page to NUMBER (default 1).

-w NUMBER - -number-width=NUMBER

Use NUMBER characters for line numbers (default 6).

nl decomposes its input into (logical) pages; by default, the line number is reset to 1 at the top of each logical

page. nl treats all of the input files as a single document; it does not reset line numbers or logical pages between files. A logical page consists of three sections: header, body, and footer. Any of the sections can be empty. Each can be numbered in a different style from the others.

**ntsysv** Simple command line Interface for setting up runlevel services

SYNTAX

**ntsysv - -back] [- -level <levels>]**

DESCRIPTION

ntsysv is a simple interface for configuring runlevel services which are also configurable through chkconfig. By default, it configures the current runlevel. If the user would like to configure other runlevels, those levels can be specified on the command line by listing the levels after --levels, without any spaces. For example, the option --levels 016 edits runlevels 0, 1, and 6.

A service is considered to be started in the runlevel set if it is started in any of the runlevels which are

being edited. The ntsysv window normally contains a Cancel button. If --back is specified, a Back button appears instead.

**ntsysv 1.3.8 - (C) 2000-2001 Red Hat, Inc.**

┌──────────────────┤ Services ├───────────────────┐

│ │

│ What services should be automatically started? │

│ [ ] aep1000 # │

│ [\*] anacron ▒ │

│ [\*] apmd ▒ │

│ [\*] atd ▒ │

│ [\*] autofs ▒ │

│ [ ] bcm5820 ▒ │

│ [ ] chargen ▒ │

│ [ ] chargen-udp ▒ │

│ │

│ ┌────┐ ┌────────┐ │

│ │ Ok │ │ Cancel │ │

│ └────┘ └────────┘ │

└─────────────────────────────────────────────────┘

**Press <F1> for more information on a service.**

**passwd** Modify a user password.

SYNTAX

**passwd**

**passwd [OPTION...]**

OPTIONS

**-d, - -delete**  delete the password for the named account (root only)

**-l, - -lock**  lock the named account (root only)

-S, - -status report password status on the named account (root only)

**-u, - -unlock**  unlock the named account (root only)

-? Show this help message

- -usage Display brief usage message

If no options are specified - passwd will change the password of the currently logged in user - will prompt for the old and new passwords.

**pico [filename]** Type pico at the UNIX prompt, along with a filename. For example, we may type *pico books.txt*.

The status line at the top displays the version number of pico, the filename, and the modifications. The bottom lines at the screen list the available editing commands. The editing commands are listed below. The ^ is the control key.

Editing Commands

^G Displays help while using pico.  
 **^O** Write the current buffer to a file.  
 **^R** Insert a file at the current cursor position.  
 ^Y Moves to the previous page.  
 **^K** Cut selected text.  
 ^C Shows the current position of the cursor.  
 **^X** Exit pico, saves buffer.  
 ^J Justify (format) the current paragraph.  
 ^W Search for text.  
 ^V Moves to next page.  
 **^U** Paste last cut text.  
 ^T Invokes the spell checker.

The Editing Commands Menu (below) is shown on the bottom of the editor window.

^G Get Help ^O WriteOut ^R Read File ^Y Prev Pg ^K Cut Text ^C Cur Pos

^X Exit ^J Justify ^W Where is ^V Next Pg ^U UnCut Text^T To Spell

**ping [*options*] *host***

System administration command. Confirm that a remote host is online and responding. **ping** is intended for use in network testing, measurement, and management. Because of the load it can impose on the network, it is unwise to use **ping** during normal operations or from automated scripts.

**Options**

**-c** *count* Stop after sending (and receiving) *count* ECHO\_RESPONSE packets.

**-f** Flood ping-output packets as fast as they come back or 100 times per second, whichever is

more. This can be very hard on a network and should be used with caution; only a privileged

user may use this option.

**-i** *wait* Wait *wait* seconds between sending each packet. Default is to wait 1 second between each

packet. This option is incompatible with the **-f** option.

**-l** *preload* Send *preload* number of packets as fast as possible before falling into normal mode of behavior.

**-n** Numeric output only. No attempt will be made to look up symbolic names for host addresses.

**-p** *digits* Specify up to 16 pad bytes to fill out packet sent. This is useful for diagnosing data- dependent problems in a network. *digits* are in hex. For example, **-p ff** will cause the sent packet to be filled with all 1s.

**-q** Quiet output—nothing is displayed except the summary lines at startup time and when finished.

**-r** Bypass the normal routing tables and send directly to a host on an attached network.

**-s** *packetsize*

Specify number of data bytes to be sent. Default is 56, which translates into 64 ICMP data

bytes when combined with the 8 bytes of ICMP header data.

**-v** Verbose—list ICMP packets received other than ECHO\_RESPONSE.

**-R** Set the IP record route option, which will store the route of the packet inside the IP header. The contents of the record route will be printed if the **-v** option is given, and will be set on return packets if the target host preserves the record route option across echoes or the **-l** option is given.

**pr** [*files*]

Convert a text file or files to a paginated, columned version, with headers. If **-** is provided as the filename, read from standard input.

**Options**

**+***beg\_pag*[:*end-pag*], **- -pages=**[*beg\_pag*[:*end-pag*]

Begin printing on page *beg\_pag* and end on *end-pag* if specified.

**-***num\_cols*, **- -columns=***num\_cols*

Print in *num\_cols* number of columns, balancing the number of lines in the columns on each page.

**-a, - -across**

Print columns horizontally, not vertically.

**-c, - -show-control-chars**

Convert control characters to hat notation (such as **^C**) and other unprintable characters to octal backslash format.

**-d, - -double-space**

Double space. **-e**[*tab-char*[*width*]], **--expand-tabs=**[*tab-char*[*width*]]

Convert tabs (or *tab-char*s) to spaces. If *width* is specified, convert tabs to *width* characters-default 8.

**-f, -F, --form-feed**

Separate pages with form feeds, not newlines.

**-h** *header*, **--header=***header*

Use *header* for the header instead of the filename.

**-D, --date-format=FORMAT**

use FORMAT to change format of header date, refer to date command for options

**%** Literal **%**.

**%a** Abbreviated weekday.

**%b** Abbreviated month name.

**%d** Day of month (01-31).

**%k** Hour in 24-hour format, without leading zeros (0-23).

**%l** Hour in 12-hour format, without leading zeros (1-12).

**%m** Month of year (01-12).

**%n** Insert a new line.

**%p** String to indicate AM or PM.

**%r** Time in **%I:%M:%S %p** (12-hour) format.

**%t** Insert a tab.

**%w** Day of week (Sunday = 0).

**%y** Last two digits of year (00-99).

**%A** Full weekday.

**%B** Full month name.

**%D** Date in **%m/%d/%y** format.

**%H** Hour in 24-hour format (00-23).

**%I** Hour in 12-hour format (01-12).

**%M** Minutes (00-59).

**%S** Seconds (00-59).

**%T** Time in **%H:%M:%S** format.

**%U** Week number in year (00-53); start week on Sunday.

**%V** Week number in year (01-52); start week on Monday.

**%W** Week number in year (00-53); start week on Monday.

**%Y** Four-digit year (e.g., 1996).

**%Z** Time zone name.

**-i**[*out-tab-char*[*out-tab-width*]], **--output-tabs**[=*out-tab-char*[*out-tab-width*]]

Replace spaces with tabs on output. Can specify alternative tab character (default is tab) and width (default is 8).

**-J, - -join-lines**

Merge full lines; ignore **-W** if set.

**-l** *lines*, **- -length=***lines*

Set page length to *lines* (default 66). If *lines* is less than 10, omit headers and footers.

**-m, --merge**

Print all files, one file per column.

**-n**[*delimiter*[*digits*]], **- -number-lines**[=*delimiter*[*digits*]]

Number columns, or, with the **-m** option, number lines. Append *delimiter* to each number (default is a tab) and limit the size of numbers to *digits* (default is 5).

**-o** *width*, **--indent=***width*

Set left margin to *width*. **-r, --no-file-warnings**

Continue silently when unable to open an input file.

**-s**[*delimiter*], **--separator**[**=***delimiter*]

Separate columns with *delimiter* (default is a tab) instead of spaces.

**-S**[*string*], **- -sep-string**[**=***string*]

Separate columns with *string*. Default is a tab with **-J** and a space otherwise.

**-t, - -omit-header**

Suppress headers, footers, and fills at end of pages.

**-T, - -omit-pagination**

Like **-t** but also suppress form feeds.

**-v, - -show-non-printing**

Convert unprintable characters to octal backslash format.

**-w** *page\_width*, **--width=***page\_width*

Set the page width to *page\_width* characters for multi-column output. Default is 72.

**-W** *page\_width*, **- -page-width=***page\_width*

Set the page width to always be *page\_width* characters. Default is 72.

**ps** [*options*]

Report on active processes. Note that you do not need to include a - before options. In options, *list* arguments should either be separated by commas or be put in double quotes. In comparing the amount of output produced, note that **e** prints more than **a** and **l** prints more than **f**.

**Options**

*pids* Include only specified processes, which are given in a comma-delimited list.

**a** List all processes.

**c** Consult **task\_struct** for command name.

**e** Include environment.

**f** "Forest" family tree format.

**h** Suppress header.

**j** Jobs format.

**l** Produce a long listing.

**m** Memory format.

**n** Print user IDs and WCHAN numerically.

**r** Exclude processes that are not running.

**s** Signal format.

**- -sort***delimiter*[+|-]*key*[,[+|-]*key*[,...]]

Similar to **O**, but designed to protect multi-letter sort keys. See the later list, "Sort keys".

**t***tty* Display only processes running on *tty*.

**u** Include username and start time.

**v vm** format.

**w** Wide format. Dont truncate long lines.

**x** Include processes without an associated terminal.

**O[+|-]***key*[,[**+**|**-**]*key*[**,**...]]

Sort processes. (See the following list, "Sort keys.")

**+** Return key to default direction.

**-** Reverse default direction on key.

**Sort keys**

**c, cmd** Name of executable.

**C, cmdline** Whole command line.

**f, flags** Flags.

**g, pgrp** Group ID of process.

**G, tpgid** Group ID of associated tty.

**j, cutime** Cumulative user time.

**J, cstime** Cumulative system time.

**k, utime** User time.

**K, stime** System time.

**m, min\_flt** Number of minor page faults.

**M, maj\_flt** Amount of major page faults.

**n, cmin\_flt** Total minor page faults.

**N, cmaj\_flt** Total major page faults.

**o, session** Session ID.

**p, pid** Process ID.

**P, ppid** Parents process ID.

**r, rss** Resident set size.

**R, resident** Resident pages.

**s, size** Kilobytes of memory used.

**S, share** Number of shared pages.

**t, tty** tty.

**T, start\_time** Processes start time.

**U, uid** User ID.

**u, user** Users name.

**v, vsize** Bytes of VM used.

**y, priority** Kernels scheduling priority.

**Fields**

**PAGEIN** Number of major page faults.

**PRI** Processes scheduling priority. A higher number indicates lower priority.

**NI** Processes **nice** value. A higher number indicates less CPU time.

**SIZE** Size of virtual image.

**RSS** Resident set size (amount of physical memory), in kilobytes.

**WCHAN** Kernel function in which process resides.

**STAT** Status:

**R** Runnable

**T** Stopped

**D** Asleep and not interruptible

**S** Asleep

**W** No resident pages (second field)

**N** Positive **nice** value (third field)

**SWAP** Amount of swap used, in kilobytes.

**SHARE** Shared memory.

**TT** Associated tty.

**TRS** Size of resident text.

**pwd** Print Working Directory

SYNTAX

**pwd [-LP]**

OPTIONS (shell built-in)

-P : The pathname printed will not contain symbolic links.

-L : The pathname printed may contain symbolic links

The default action is to show the current folder as an absolute path. All components of the path will be actual folder names - none will be symbolic links.

**rawrite.exe** MSDOS program used to create a boot image for Linux Installations

A boot image boot floppy is required if you cannot access your CDROM as a boot device or you are performing a network installation (FTP, HTTP, Samba, NFS, etc).

The **rawrite** command looks like the following (assuming your CD is drive D:):

C:\> **d:**

D:\> **cd \dosutils**

D:\> dosutils> **rawrite**

**Rawrite** first asks you for the filename of a disk image; enter the directory and name of the image you wish to write.

Enter disk image source file name:**D:\images\*<image file>***

then **rawrite** asks for a disk drive to write the image to:

Enter target diskette drive:**a:**

Finally, **rawrite** asks for confirmation that a formatted disk is in the drive you've selected. After pressing **Enter** to confirm, **rawrite** copies the image file onto the disk.

Please insert a formatted diskette into drive A: and press

--ENTER-- : **Enter**

D:\ dosutils>

The **rawrite** utility will return a message indicating whether your disk was created successfully, or it will inform you if your image failed to transfer correctly. Make sure that you label the disk according to its contents.

**redhat-config-bind** DNS Server configuration tool

**redhat-config-date** Time and Date configuration tool

**redhat-config-httpd** Apache Web Server configuration tool

**redhat-config-kickstart** KickStart (StartUp Config) file tool

**redhat-config-mouse** Configure mouse type

**redhat-config-network** Configure Network Interface Card (NIC) settings

**redhat-config-network-druid** Add a new Network Device (NIC, modem, ISDN, etc)

**redhat-config-network-tui** Command Line Interface (CLI) tool for NIC settings

**redhat-config-nfs** NFS Server configuration tool

**redhat-config-packages** Remove/Install Linux Packages from Redhat CDROM

**redhat-config-printer** Printer configuration tool (same as printtool)

**redhat-config-printer-tui** Command Line Interface tool for Printer configuration

**redhat-config-rootpassword** Change root user password

**redhat-config-samba** SAMBA (SMB) Server configuration tool

**redhat-config-securitylevel** Firewall Security configuration tool

**redhat-config-services** Start/Stop/Restart Services, configure startup services

**redhat-config-soundcard** Check for Soundcard detection

**redhat-config-users** Configuration tool to Add/Delete/Modify Users & Groups

**redhat-config-xfree86** Configuration tool for X-Windows display settings

*The redhat-config-\* commands allow you to configure the settings for Redhat Linux. The commands that end in -tui (text user interface) can be run from a command-line session. The other commands bring up an X-Window GUI application.*

**renice** [*priority*] [*options*] [*target*]

DESCRIPTION

Control the scheduling priority of various processes as they run. May be applied to a process, process group, or user (*target*). A privileged user may alter the priority of other users' processes. *priority* must, for ordinary users, lie between 0 and the environment variable PRIO\_MAX (normally 20), with a higher number indicating increased niceness. A privileged user may set a negative priority, as low as PRIO\_MIN, to speed up processes.

OPTIONS

**+***num* Specify number by which to increase current priority of process, rather than an absolute priority number.

**-***num* Specify number by which to decrease current priority of process, rather than an absolute priority number.

**-g** Interpret *target* parameters as process group IDs.

**-p** Interpret *target* parameters as process IDs (default).

**-u** Interpret *target* parameters as usernames.

**rm** Remove files (delete/unlink)

SYNTAX

**rm [OPTION]... FILE...**

OPTIONS

**-f, - -force** ignore nonexistent files, never prompt

**-i, - -interactive** prompt before any removal

**-r, -R, - -recursive** remove the contents of directories recursively

-v, - -verbose explain what is being done

To remove a file you must have write permission on the file and the folder where it is stored. rm -rf will recursively remove folders and their contents The OWNER of a file does not need rw permissions in order to rm it.

**rmdir** [*options*] *directories*

Delete the named *directories* (not the contents). *directories* are deleted from the parent directory and must be empty (if not, **rm -r** can be used instead). See also **mkdir**.

**Options**

**- -ignore-fail-on-non-empty**

Ignore failure to remove directories that are not empty.

**-p, - -parents**

Remove *directories* and any intervening parent directories that become empty as a result; useful for removing subdirectory trees.

**- -verbose**

Turn on verbose mode; print message for each directory as it is processed.

**route** [*option*] [*command*]

TCP/IP command. Manually manipulate the routing tables normally maintained by **routed**. **route** accepts two commands: **add**, to add a route, and **del**, to delete a route. The two commands have the following syntax:

**add** [**-net** | **-host**] *address* [**gw** *gateway*] [**netmask** *mask*] [**mss** *tcp-mss*] [**dev** *device*]

**del** *address*

*address* is treated as a plain route unless **-net** is specified or *address* is found in */etc/networks*. **-host** can be used to specify that *address* is a plain route whether or not it is found in */etc/networks*. The keyword *default* means to use this route for all requests if no other route is known. You can specify the *gateway* through which to route packets headed for that address, its *netmask*, TCP *mss*, and the *device* with which to associate the route. Only a privileged user may modify the routing tables. If no command is specified, **route** prints the routing tables.

**rpm** - RPM Redhat Package Manager

RPM is an open source packaging system that makes it easy to maintain your Linux system. It provides a easy way to install, uninstall, query, and verify the RPM packages on your system. It is included with a number of Linux distributions including Redhat and SUSE. This will introduce you to installing, uninstalling, querying and verifying RPM packages using The RedHat Package Manager.

**Installing**

The syntax for the command to install a new rpm package is:

**rpm -i[install options] <package name>**

The simplest form for installing an RPM is:

**rpm -i example-1.0-1.i386.rpm**

If the RPM installs without errors you will be dropped back to the command prompt.   
 The most common form used for installing an RPM is:

**rpm -ivh example-1.0-1.i386.rpm**

The "v" meaning "verbose" displays the name of the RPM being installed. The "h" meaning "hash" displays a series of hash marks as a progress indicator as the package is being installed. You can also install multiple RPMS at the same time by separating the filenames with a space:

**rpm -ivh example-1.0-1.i386.rpm example2-1.0-1.i386.rpm**

If you want to test an RPM before you install it and see if there are any potential problems you use the -- test option:

**rpm -ivh - -test example-1.0-1.i386.rpm**

This will not install the RPM it will only check for and display any errors there might be when installing the

package.   
  
**Uninstalling**

The syntax for the command to uninstall an rpm package is:

**rpm -e <package name>**

Example of uninstalling an RPM package:

**rpm -e example**

It is best to test the RPM first before you try to uninstall it and make sure you will not be breaking any dependencies:

**rpm -e - -test example**

If you get a dependency error when trying to uninstall an RPM you can use the --nodeps option, but it is not recommended. You will probably break the packages that depend on it.

**Upgrading**

The syntax for the command to upgrade an rpm package is:

**rpm -U[install options] <package name>**

The most common form used for upgrading an RPM is:

**rpm -Uvh example-1.0-1.i386.rpm**

The -U options upgrades the package currently installed to the new version RPM and deletes the old version from the system. You can also use the -U option instead of the -i option to install a new RPM package.

Another way to upgrade packages is to use the -F, freshen option:

**rpm -F|- -freshen[install options] <package name>**

This option will only upgrade a package if an older version is already installed:

**rpm -Fvh example-1.0-1.i386.rpm   
 rpm -vh - -freshen example-1.0-1.i386.rpm**

**Common errors when installing and upgrading.**

If you try to install an RPM that is the same version as one already installed you will see:

**rpm -ivh example-1.0-1.i386.rpm   
 example   -package example-1.0-1 is already installed**

If you need to reinstall an RPM that is already installed you can use the --replacepkgs option.

**rpm -ivh - -replacepkgs example-1.0-1.i386.rpm**

Another way is to use the --force option.

**rpm -ivh - -force example-1.0-1.i386.rpm**

RPM packages can require files from other RPMs to be installed in order to run properly. If you try to install an RPM that requires something not already installed on your system you will see:

**rpm -ivh example-1.0-1.i386.rpm   
 failed dependencies:   
 missing file is needed by example-1.0-1**

You will need to first install the RPM that includes the missing file(s). You can "force" the RPM to install anyway by using the - -nodeps option, but this is not a good idea because the package will probably not run correctly.

**Querying RPMs**

The syntax for the command to query an rpm package is:

**rpm -q[query options]**

If you want to see whether an rpm is installed and the version number use:

**rpm -q example**

If you want to list all the rpms on your system use one of the following:

**rpm -qa   
 rpm -q - -all**

This isn’t really a good way to output a list of RPMS on system. A better way is to output an alphabetized list to a file.

**rpm -qa | sort -o ~/rpmlist.txt**

This will output an alphabetized list of all the RPMS on your system to rpmlist.txt and put it in your home

directory. ~ means home directory.   
  
 Say you wanted to find all the RPMS on your system that contained a specific string. The following example will find all the RPMS on the system with "kde" in the package name:

**rpm -qa | grep kde**

Maybe it would look better alphabetized also:

**rpm -qa | grep kde | sort**

If you want to find out what RPM package installed a file on your system use:

**rpm -qf full/path/to/rpm   
 rpm -q - -file full/path/to/rpm**

Example to find out what RPM /usr/bin/cpp was installed by:

**rpm -qf /usr/bin/cpp**

If you want to display package information, including name, version, and description for a package already installed:

**rpm -qi example**

If you want to display a list of files in a package already installed:

**rpm -ql example   
 rpm -q - -list example**

If the list of files is too long to fit on one screen you can use "less". Use the arrow keys, page up, page down to scroll, hit q to get back to the console prompt.

**rpm -ql example | less**

If you want to use the -i, info or -l, list options on an RPM package that hasn’t been installed yet you need to also use the -p, package option:

**rpm -qpi example-1.0-1.i386.rpm   
 rpm -qpl example-1.0-1.i386.rpm**

**Verifying RPMs**

The syntax for the command to verify an rpm package is:

**rpm -V|- -verify [verify-options]**

Verifying an RPM is a way to check if the installation of a particular package is correct. It checks things like MD5 sum, file size, and permissions on each file installed by an RPM package.   
  
Example of verifying an RPM package:

**rpm -V example   
 rpm - -verify example**

If nothing is outputted when you run verify on an RPM, no errors were detected. If any errors are outputted they will be an 8 character string, followed by a "c" if it is a configuration file and the filename.   
  
Example of verifying the bind RPM:

**rpm -V bind   
 SM5..UGT c /etc/rndc.conf**

The "c" here shows this as a configuration file so the errors are harmless since this file has been edited since it was installed.

. -- test passed   
 ? -- file couldn’t be read, usually a permission problem   
 S -- File size   
 5 -- MD5 sum   
 M -- Mode   
 L -- Symlink   
 D -- Device   
 U -- User   
 G -- Group   
 T -- Mtime

To verify all the RPMs on your system use:

**rpm -Va   
 rpm -V - -all**

**Using wildcards with RPM.**

You just downloaded thirty updated RPMs for your system, now how do you install them? Thirty separate rpm -Fvh <package name> commands? No, you can use the \* wildcard. Change to the directory where you have the new RPMs and use:

**rpm -Fvh \*.rpm**

**service** <servicename> [start | stop | restart]

DESCRIPTION

The service command (actually a script) will start , stop or restart services or daemons. Most services can be found in the directory **/etc/rc.d/init.d/**. You can also manually start, stop or restart daemons directly by

typing: **/etc/rc.d/init.d/** <daemon-name> start | stop | restart.

A partial list service/daemon names that can be used with the service command follows:

**apmd** - Monitors battery status and can shut down the system if power is low.

**atd** - Runs commands scheduled by the "at" program at their scheduled times.

**autofs** - Automount daemon, used to automatically mount filesystems such as CDROMs or floppies.

**crond** - A daemon that executes scheduled commands according to the /etc/crontab file.

**dhcpd** - Provides DHCP services to "lease" out IP addresses to remote machines.

**gpm** - Provides mouse support to Linux.

**httpd** - The Apache hypertext transfer protocol Web server.

**inet** - The internet super daemon (inetd) that provides all the services specified in /etc/inetd.conf. **kudzu** - Detects and configures new or changed hardware on a system.

**lpd** - Provides printing services to Linux. It is a print spooler daemon.

**named** - Provides DNS services. Used to translate local IP addresses to names and vice-versa.

**network** - Brings up all the network interfaces under the directory /etc/sysconfig/network-scripts.

**nfs** - Provides Network File System server services.

**pcmcia** - Provides access to PCMCIA (PC Cards) services configured in the /etc/exports file.

**routed** - Provides automatic router table updates using the RIP dynamic routing information protocol.

**rstatd** - Allows users on a network to get performance information for any machine on the network.

**rusersd** - Provides services that allow users to find one another over the network.

**sendmail** - The sendmail mail transport agent daemon used to move e-mail one machine to another.

**smb** - Provides SMB (Samba) client/server services which include file and print services.

**snmpd** - Provides Simple Network Management Protocol support to Linux.

**squid** - Runs the squid proxy web server

**syslog** - System logging daemon which records system events to log files in the directory "/var/log". **ntpd** - Starts Network Time Protocol NTPv3 daemon. NTP provides a means to synchronize time.

**ypbind** - Binds YP/NIS clients to a yellow pages server. NIS (Network Information Service).

**yppasswdd** - Allows users to change their passwords on systems running YP/NIS

**ypserv** - This daemon provides the YP/NIS (Network Information System) server functions.

**shutdown** Shutdown or restart Linux

SYNTAX

**shutdown [options] when [message]**

OPTIONS

**-c** Cancel a shutdown that is in progress.

-f Reboot fast, by suppressing the normal call to fsck when rebooting.

**-h** Halt the system when shutdown is complete.

-k Print the warning message, but suppress actual shutdown.

-n Perform shutdown without a call to init.

**-r**  Reboot the system when shutdown is complete.

-t sec Ensure a sec-second delay between killing processes and changing the runlevel.

**Examples**

Shutdown immediately: **shutdown -h now**

Reboot immediately: **shutdown -r now**

Shutdown at 8 pm: **shutdown -h 20:00**

Shutdown in 10 minutes: **shutdown -h +10**

**sleep** Delay for a specified time, pause for an amount of time specified by the sum of the values of the command line arguments

SYNTAX

**sleep [NUMBER[smhd]]...**

key:

s : seconds (default)

m : minutes

h : hours

d : days

**sort** [*options*] [*files*]

Sort the lines of the named *files*. Compare specified fields for each pair of lines, or, if no fields are specified, compare them by byte, in machine collating sequence.

**Options**

**-b** Ignore leading spaces and tabs.

**-c** Check whether *files* are already sorted, and, if so, produce no output.

**-d** Sort in dictionary order.

**-f** Fold—ignore uppercase/lowercase differences.

**-I** Ignore nonprinting characters (those outside ASCII range 040-176).

**-m** Merge (i.e., sort as a group) input files.

**-n** Sort in arithmetic order.

**-o***file* Put output in *file*.

**-r** Reverse the order of the sort.

**-t***c* Separate fields with *c* (default is a tab).

**-u** Identical lines in input file appear only one (**u**nique) time in output.

**-z***recsz P*rovide *recsz* bytes for any one line in the file. This option prevents abnormal termination of sort in certain cases.

**+***n* [-*m*] Skip *n* fields before sorting, and sort up to field position *m*. If *m* is missing, sort to end of line.

Positions take the form *a*.*b*, which means character *b* of field *a*. If .*b* is missing, sort at the first character of the field.

**-k** *n*[,*m*] Similar to +. Skip *n*-1 fields and stop at *m*-1 fields (i.e., start sorting at the *n*th field, where the fields are numbered beginning with 1).

**-M** Attempt to treat the first three characters as a month designation (JAN, FEB, etc.). In comparisons, treat JAN < FEB and any valid month as less than an invalid name for a month.

**-T** *tempdir* Directory pathname to be used for temporary files.

**Examples**

List files by decreasing number of lines: **wc -l \* | sort -r**

Alphabetize, remove duplicates, and print the frequency of each word: **sort -fd wordlist | uniq -c**

Sort the password file numerically by the third field (user ID):  **sort +2n -t: /etc/passwd**

**su** Run a command with substitute user and group id, allow one user to temporarily become another

user. It runs a command (often an interactive shell) with the real and effective user id, group id, and Supplemental groups of a given USER

SYNTAX

**su [OPTION]... [USER [ARG]...]**

OPTIONS

-**c COMMAND --command=COMMAND**

Pass COMMAND, a single command line to run, to the shell with a -c option instead of starting an interactive shell.

**- l - -login**

Make the shell a login shell. This means the following. Unset all environment variables except

TERM, HOME, and SHELL (which are set as described above), and USER and LOGNAME (which are set, even for the super-user, as described above), and set PATH to a compiled-in default value. Change to USER’s Home directory. Prepend - to the shells name, intended to make it read its login startup file(s).

-**m -p - -preserve-environment**

Do not change the environment variables HOME, USER, LOGNAME, or SHELL. Run the shell given in the environment variable SHELL instead of the shell from USER's passwd entry, unless the user running su is not the superuser and USER’s shell is restricted. A "restricted shell" is one that is not listed in the file /etc/shells, or in a compiled-in list if that file does not exist. Parts of what this option does can be overridden by --login and --shell.

-s SHELL - -shell=SHELL

Run SHELL instead of the shell from USER’s passwd entry, unless the user running su is not the

superuser and USER’s shell is restricted (see -m just above). if no USER is given, the default is

root, the super-user. The shell to use is taken from USER’s passwd entry, or /bin/sh if none is

specified there. If USER has a password, su prompts for the password unless run by a user with

effective user id of zero (the super-user).   
  
By default, su does not change the current directory. It sets the environment variables HOME and SHELL from the password entry for USER, and if USER is not the super-user, sets USER and LOGNAME to USER. By default, the shell is not a login shell. Any additional ARG's are passed as

additional arguments to the shell.   
  
GNU su does not treat /bin/sh or any other shells specially (e.g., by setting argv[0] to -su, passing -c only to certain shells, etc.). su can optionally be compiled to use syslog to report failed, and optionally successful, su attempts. (If the system supports syslog.) However, GNU su does not check if the user is a member of the wheel group; see options above.

**tar** Tape ARchiver, store, list or extract files in an archive.

SYNTAX

**tar OPTION... [NAME]...**

MOST COMMON OPTIONS

- -create -c Create a new tar archive.

- -list -t List the contents of an archive.

- -extract -x Extract one or more members from an archive.

- -file=ARCHIVE-NAME -f ARCHIVE-NAME

Specify the name of an archive file. You can specify an argument for the - -file=ARCHIVE-NAME (-f ARCHIVE-NAME) option whenever you use tar; this option determines the name of the archive file that tar will work on. If you dont specify this argument, then tar will use a default, usually some physical tape drive attached to your machine. If there is no tape drive attached, or the default is not meaningful, then tar will print an error message. The error message might look roughly like one of the following:

tar: cant open /dev/rmt8 : No such device or address

tar: cant open /dev/rsmt0 : I/O error

To avoid confusion, we recommend that you always specify an archive file name by using - -file=ARCHIVE-NAME (-f ARCHIVE-NAME) when writing your tar COMMANDS

- -verbose -v Show the files being worked on as tar is running. - -verbose (-v) shows details about the results of running tar. This can be especially useful when the results might not be obvious. For example, if you want to see the progress of tar as it writes files into the archive, you can use the --verbose option. In the beginning, you may find it useful to use --verbose at all times; when you are more accustomed to tar, you will likely want to use it at certain times but not at others. We will use - -verbose at times to help make something clear, and we will give many examples both using and not using - -verbose to show the differences. Sometimes, a single instance of - -verbose on the command line will show a full, ls style listing of an archive or files, giving sizes, owners, and similar information. Other times, - -verbose will only show files or members that the particular operation is operating on at the time. In the latter case, you can use - -verbose twice in a command to get a listing such as that in the former case.

The 8 **BASIC MODES OF OPERATION** (Use only one at a time)

- -append -r Appends files to the end of the archive.

- -compare - -diff -d

Compares archive members with their counterparts in the file system, and reports

differences in file size, mode, owner, modification date and contents.

- -concatenate - -catenate -A

Appends other tar archives to the end of the archive.

- -create -c Creates a new tar archive.

- -delete Deletes members from the archive. Dont try this on a archive on a tape!

- -extract - -get -x

Extracts members from the archive into the file system.

- -list -t Lists the members in an archive.

- -update -u Adds files to the end of the archive, but only if they are newer than their counterparts already in the archive, or if they do not already exist in the archive.

Examples  
 To tar and zip a file

**tar -czvf MyArchive Source\_file** or **tar** **- -create - -gzip - -verbose --file=MyArchive Source\_file**

The reverse process to extract the file

**tar -xzvf MyArchive Source\_file** or **tar - -extract - -gunzip --verbose - -file=MyArchive Source\_file**

**top** [*options*]

Provide information (frequently refreshed) about the most CPU-intensive processes currently running. See **ps** for explanations of the field descriptors.

**Options**

**-b** Run in batch mode; dont accept command-line input. Useful for sending output to another command or to a file.

**-c** Show command line in display instead of just command name.

**-d** *delay* Specify delay between refreshes.

**-I** Suppress display of idle and zombie processes.

**-n** *num* Update display *num* times, then exit.

**-p** *pid* Monitor only processes with the specified process ID.

**-q** Refresh without any delay. If user is privileged, run with highest priority.

**-s** Secure mode. Disable some (dangerous) interactive commands.

**-S** Cumulative mode. Print total CPU time of each process, including dead child processes.

**Interactive commands**

**space** Update display immediately.

**c** Toggle display of command name or full command line.

**f, F** Add fields to display or remove fields from the display.

**h, ?** Display help about commands and the status of secure and cumulative modes.

**k** Prompt for process ID to kill and signal to send (default is 15) to kill it.

**I** Toggle suppression of idle and zombie processes.

**l** Toggle display of load average and uptime information.

**m** Toggle display of memory information.

**n, #** Prompt for number of processes to show. If 0 is entered, fill screen (default).

**o, O** Change order of displayed fields.

**q** Exit.

**r** Apply **renice** to a process. Prompt for PID and **renice** value. Suppressed in secure mode.

**s** Change delay between refreshes. Prompt for new delay time, which should be in seconds.

Suppressed in secure mode.

**t** Toggle display of processes and CPU states information.

**A** Sort by age, with newest first.

**^L** Redraw screen.

**M** Sort tasks by resident memory usage.

**N** Sort numerically by process ID.

**P** Sort tasks by CPU usage (default).

**S** Toggle cumulative mode. (See the **-S** option.)

**T** Sort tasks by time/cumulative time.

**W** Write current setup to *~/.toprc*. This is the recommended way to write a **top** configuration file.

**touch** [*options*] *files*

For one or more *files*, update the access time and modification time (and dates) to the current time and date. **touch** is useful in forcing other commands to handle files a certain way; e.g., the operation of **make**, and sometimes **find**, relies on a files access and modification time. If a file doesn’t exist, **touch** creates it with a filesize of 0.

**Options**

**-a, - -time=atime, --time=access, --time=use**

Update only the access time.

**-c, - -no-create**

Do not create any file that doesn’t already exist.

**-d** *time*, **- -date** *time*

Change the time value to the specified *time* instead of the current time. *time* can use several formats and may contain month names, time zones, a.m. and p.m. strings, as well as others.

**-m, --time=mtime, --time=modify**

Update only the modification time.

**-r** *file*, **--reference** *file*

Change times to be the same as those of the specified *file*, instead of the current time.

**-t** *time*

Use the time specified in *time* instead of the current time. This argument must be of the format:

*[[cc]yy]mmddhhmm[.ss]*, indicating optional century and year, month, date, hours, minutes, and optional seconds.

**tr** [*options*] [*string1* [*string2*]]

Translate characters—copy standard input to standard output, substituting characters from *string1* to *string2* or deleting characters in *string1*.

**Options**

**-d, - -delete** Delete characters in *string1* from output.

**-t, - -truncate-set1** Truncate *string1* to the length of *string2* before translating.

**Special characters**

Include brackets ([]) where shown.

**\a** ^G (bell)

**\b** ^H (backspace)

**\f** ^L (form feed)

**\n** ^J (newline)

**\r** ^M (carriage return)

**\t** ^I (tab)

**\v** ^K (vertical tab)

**\***nnn* Character with octal value *nnn*.

**\\** Literal backslash.

*char1*-*char2*

All characters in the range *char1* thru *char2*. If *char1* does not sort before *char2*, an error.

[*char1*-*char2*]

Same as *char1*-*char2* if both strings use this.

[*char*\*] In *string2*

Expand *char* to the length of *string1*.

[*char*\**number*]

Expand *char* to number occurrences. **[x\*4]** expands to **xxxx**, for instance.

[:*class*:]

Expand to all characters in *class*, where *class* can be:

**alnum** Letters and digits **alpha** Letters

**blank** Whitespace **cntrl** Control characters

**digit** Digits **graph** Printable characters except space

**lower** Lowercase letters **print** Printable characters

**punct** Punctuation **space** Whitespace (horizontal or vertical)

**upper** Uppercase letters **xdigit** Hexadecimal digits

[**=***char***=**] The class of characters in which *char* belongs.

**Examples**

Change uppercase to lowercase in a file:

**cat file | tr [A-Z] [a-z]**

Turn spaces into newlines (ASCII code 012):

**tr \012 < file**

Strip blank lines from **file** and save in **new.file** (or use **011** to change successive tabs into one tab):

**cat file | tr -s "" "\012" > new.file**

Delete colons from **file**; save result in **new.file**:

**tr -d : < file > new.file**

**traceroute** [*options*] *host* [*packetsize*]

TCP/IP command. Trace route taken by packets to reach network host. **traceroute** attempts tracing by launching UDP probe packets with a small TTL (time to live), then listening for an ICMP "time exceeded" reply from a gateway. *host* is the destination hostname or the IP number of host to reach. *packetsize* is the packet size in bytes of the probe datagram. Default is 38 bytes.

**Options**

**-d** Turn on socket-level debugging.

**-g** *addr* Enable the IP LSRR (Loose Source Record Route) option in addition to the TTL tests, to ask

how someone at IP address *addr* can reach a particular target.

**-l** Include the time-to-live value for each packet received.

**-m** *max\_ttl* Set maximum time-to-live used in outgoing probe packets to *max-ttl* hops. Default is 30 hops.

**-n** Show numerical addresses; do not look up hostnames. (Useful if DNS is not functioning properly.)

**-p** *port* Set base UDP port number used for probe packets to *port*. Default is (decimal) 33434.

**-q** *n* Set number of probe packets for each time-to-live setting to the value *n*. Default is 3.

**-r** Bypass normal routing tables and send directly to a host on an attached network.

**-s** *src\_addr* Use *src\_addr* as the IP address that will serve as the source address in outgoing probe

packets.

**-t** *tos* Set the type-of-service in probe packets to *tos* (default 0). The value must be a decimal integer in the range 0 to 255.

**-v** Verbose—received ICMP packets (other than TIME\_EXCEEDED and

PORT\_UNREACHABLE) will be listed.

**-w** *wait* Set time to wait for a response to an outgoing probe packet to *wait* seconds (default is 3 seconds).

**umount** [*options*] [*special-device/directory*]

System administration command. Unmount a filesystem. **umount** announces to the system that the removable file structure previously mounted on device *special-device* is to be removed. **umount** also works by specifying the directory. Any pending I/O for the filesystem is completed, and the file structure is flagged as clean.

**Options**

**-a** Unmount all filesystems that are listed in */etc/mtab*.

**-n** Unmount, but do not record changes in */etc/mtab*.

**-t** *type* Unmount only filesystems of type *type*.

**uname** [*options*] Print information about the machine and operating system. Without options, print the

name of the operating system (Linux).

1. OPTIONS

**-a, --all** Combine all the system information from the other options.

**-m, --machine** Print the hardware the system is running on.

**-n, --nodename** Print the machine's hostname.

**-r, --release** Print the release number of the kernel.

**-s, --sysname** Print the name of the operating system (Linux).

**-p, --processor** Print the type of processor (not available on all versions).

**-v** Print build information about the kernel.

**umask** User's file creation mask. Set the shell process's file creation mask to *mode*.

SYNTAX

**umask [-p] [-S] [*mode*]**

OPTIONS

mode File creation mask

-S Print the mask in symbolic format

-p Output in a form that may be reused as input

The current value of the mask will be printed if *mode* is omitted,

If *mode* begins with a digit, it is interpreted as an octal number; if not, it is interpreted as a symbolic mode mask similar to that accepted by the chmod command. The return status is zero if the mode is successfully changed or if no *mode* argument is supplied, and non-zero otherwise.   
  
Note that when the mode is interpreted as an octal number, each number of the umask is subtracted from 7. Thus, a umask of 022 results in permissions of 755.

**unalias** Removes alias definitions.

SYNTAX

**unalias** [-a](http://publib16.boulder.ibm.com/pseries/en_US/cmds/aixcmds5/unalias.htm#elmni326harr)

**unalias AliasName ...**

DESCRIPTION

The unalias command removes the definition for each alias name specified, or removes all alias definitions if the -a flag is used. Alias definitions are removed from the current shell environment.

FLAGS

|  |  |
| --- | --- |
| -a | Removes all alias definitions from the current shell environment. |

**EXIT STATUS**

The following exit values are returned:

|  |  |
| --- | --- |
| 0 | Successful completion. |
| >0 | One of the alias names specified did not represent a valid alias definition, or an error occurred. |

**useradd** [*options*] [*user*]

System administration command. Create new user accounts or update default account information. Unless invoked with the **-D** option, *user* must be given. **useradd** will create new entries in system files. Home directories and initial files may also be created as needed.

**OPTIONS**

**-d** *dir* Home directory. The default is to use *user* as the directory name under the *home* directory

specified with the **-D** option.

**-e** *date* Account expiration *date*. *date* is in the format MM/DD/YYYY. Two-digit year fields are also

accepted. The value is stored as the number of days since January 1, 1970. This option requires the use of shadow passwords.

**-f** *days* Permanently disable account this many *days* after the password has expired. A value of -1

disables this feature. This option requires the use of shadow passwords.

**-g** *group* Initial *group* name or ID number. If a different default group has not been specified using the **-D** option, the default group is 1.

**-G** *groups* Supplementary *groups* given by name or number in a comma-separated list with no whitespace.

**-k** [*dir*] Copy default files to users home directory. Meaningful only when used with the **-m** option.

Default files are copied from */etc/skel/* unless an alternate *dir* is specified.

**-m** Make users home directory if it does not exist. The default is not to make the home directory.

**-o** Override.

**-s** *shell* Login *shell*.

**-u** *uid* Numerical user ID. The value must be unique unless the **-o** option is used. The default value is the smallest ID value greater than 99 and greater than every other *uid*.

**-D** [*options*] Set or display defaults. If *options* are specified, set them. If no options are specified, display

current defaults. The options are:

*-b dir* Home directory prefix to be used in creating home directories. If the **-d** option is not

used when creating an account, the *user* name will be appended to *dir*.

*-e date* Expire *date*. Requires the use of shadow passwords.

*-f days* Number of *days* after a password expires to disable an account. Requires the use

of shadow passwords.

*-g group* Initial *group* name or ID number.

*-s shell* Default login *shell*.

**userdel** [*option*] *user*

System administration command. Delete all entries for *user* in system account files.

**Option**

**-r** Remove the home directory of *user* and any files contained in it.

**usermod [*options*] *user***

System administration command. Modify *user* account information.

**Options**

**-d** *dir* Home directory.

**-e** *date* Account expiration *date*. *date* is in the format MM/DD/YYYY. Two-digit year fields are also

accepted, but the value is stored as the number of days since January 1, 1970. This option

requires the use of shadow passwords.

**-f** *days* Permanently disable account this many *days* after the password has expired. A value of -1

disables this feature. This option requires the use of shadow passwords.

**-g** *group* Initial *group* name or number.

**-G** *groups* Supplementary *groups* given by name or number in a comma-separated list with no whitespace. *user* will be removed from any groups to which they currently belong that are

not included in *groups*.

**-l** *name* Login *name*. This cannot be changed while the user is logged in.

**-o** Override. Accept a nonunique *uid* with the **-u** option.

**-s** *shell* Login *shell*.

**-u *uid***Numerical user ID. The value must be unique unless the -o option is used. Any files owned by*user* in the users home directory will have their user ID changed automatically. Files outside of the home directory will not be changed. *user* should not be executing any processes while this is changed.

**users** print the user names of users currently logged in to the current host

SYNTAX

**users [ FILE ]**

DESCRIPTION

Output who is currently logged in according to FILE. If FILE is not specified, use /var/run/utmp. /var/log/wtmp as FILE is common.

**wc** Displays byte, word, and line counts

DESCRIPTION

wc counts the number of bytes, characters, whitespace-separated words, and newlines in each given FILE, or standard input if none are given or for a FILE of -.

SYNTAX

**wc [OPTION]... [FILE]...**

wc prints one line of counts for each file, and if the file was given as an argument, it prints the file name following the counts. If more than one FILE is given, wc prints a final line containing the cumulative counts, with the file name total. The counts are printed in this order: newlines, words, characters, bytes. By default, each count is output right-justified in a 7-byte field with one space between fields so that the numbers and file names line up nicely in columns.

By default, wc prints three counts: the newline, words, and byte counts. Options can specify that only certain counts be printed. Options do not undo others previously given, so wc --bytes –words prints both the byte counts and the word counts. With the --max-line-length option, wc prints the length of the longest line per file, and if there is more than one file it prints the maximum (not the sum) of those lengths.

OPTIONS

-c - -bytes Print only the byte counts.

-m - -chars Print only the character counts.

-w - -words Print only the word counts.

-l - -lines Print only the newline counts.

-L - -max-line-length

Print only the maximum line lengths.

**whereis** [*options*] *files*

Locate the binary, source, and manual page files for specified commands/files. The supplied filenames are first stripped of leading pathname components and any (single) trailing extension of the form *.ext* (for example, *.c*). Prefixes of *s.* resulting from use of source code control are also dealt with. **whereis** then attempts to locate the desired program in a list of standard Linux directories (e.g., */bin*, */etc*, */usr/bin*, */usr/local/bin/*, etc.).

**Options**

**-b** Search only for binaries.

**-f** Terminate the last directory list and signal the start of filenames; required when any of the **-B**, **-M**, or **-S** options are used.

**-m** Search only for manual sections.

**-s** Search only for sources.

**-u** Search for unusual entries, that is, files that do not have one entry of each requested type. Thus, the command **whereis -m -u \*** asks for those files in the current directory that have no documentation.

**-B** *directories*

Change or otherwise limit the directories to search for binaries.

**-M** *directory*

Change or otherwise limit the directories to search for manual sections.

**-S** *directory*

Change or otherwise limit the directories to search for sources.

**which - shows the full path of (shell) commands.**

SYNTAX

**which [options] [--] program\_name [...]**

DESCRIPTION

Which takes one or more arguments. For each of its arguments it prints to stdout the full path of the executables that would have been executed when this argument had been entered at the shell prompt. It does this by searching for an executable or script in the directories listed in the environment variable PATH using the same algorithm as bash(1).

OPTIONS

- -all, -a Print all matching executables in PATH, not just the first.

- -read-alias, -i Read aliases from stdin, reporting matching ones on stdout. Useful in combination with using an alias for which itself. For example alias which=alias | which -i.

- -skip-alias Ignore option --read-alias, if any. This is useful to explicitly search for normal

binaries, while using the --read-alias option in an alias or function for which.

- -read-functions Read shell function definitions from stdin, reporting matching ones on stdout. This is

useful in combination with using a shell function for which itself. For example:

which() { declare -f | which --read-functions $@ } export -f which

- -skip-functions Ignore option --read-functions, if any. This is useful to explicitly search for normal

binaries, while using the --read-functions option in an alias or function for which.

- -skip-dot Skip directories in PATH that start with a dot.

- -skip-tilde Skip directories in PATH that start with a tilde and executables which reside in the

HOME directory.

- -show-dot If a directory in PATH starts with a dot and a matching executable was found for that path, then print "./programname" rather than the full path.

- -show-tilde Output a tilde when a directory matches the HOME directory. This option is ignored when which is invoked as root.

- -tty-only Stop processing options on the right if not on tty.

RETURN VALUE

Which returns the number of failed arguments, or -1 when no program\_name was given.

EXAMPLE

The recommended way to use this utility is by adding an alias like the following: [ba]sh:

alias which alias | /usr/bin/which --tty-only --read-alias --show-dot --show-tilde

**who** show users who is logged on

**whoami** show current user.

SYNTAX

**who [OPTION]**

OPTIONS

-a, - -all same as -b -d --login -p -r -t -T -u

-b, - -boot time of last system boot

-H, - -heading print line of column headings

- -login print system login processes (equivalent to SUS -l)

-l, - -lookup attempt to canonicalize hostnames via DNS (-l is deprecated, use --lookup)

-m only hostname and user associated with stdin

-p, - -process print active processes spawned by init

-q, - -count all login names and number of users logged on

-r, - -runlevel print current runlevel

-s, - -short print only name, line, and time (default)

-t, - -time print last system clock change

-T, -w, - -mesg add user’s message status as +, - or ?

-u, - -users list users logged in

- -message same as -T