# Filesystem

**cd** [-] [*directory*] ·**Change directory**
- :Change to the previous directory you were in

**chmod** [*opts*] <*mode*> <*filepattern*> ·**Change permissions**
-R :Change permissions recursively

**chown** [-R] <*user*>[.*group*] <*file*> ·**Change ownership**
-R :Change ownership recursively

**cp** [*opts*] <*from*> <*to*> ·**Copy files and directories**
-i :Interactive mode. Prompt before overwriting
-p :Preserve file permissions and ownership
-R :Copy directories recursively

**df** [*opts*] [*device name*] ·**Print filesystem usage info**
-a :Show all filesystems.
-h :Human readable format. Quantify byte information.
-i :Show inode usage info.

**du** [*opts*] [*pattern*] ·**Show space usage on files and dirs**
-c :Produce a grand total for all arguments.
-h :Human readable format. Quantify byte information.
-s :Summarize. Only show a total for each argument.
-S :Do not include size of subdirectories.

**find** <*path*> [-name '*pattern*'] ·**Search for a file.**
Ex: find /home -name 'index.html' -perms -644 -ls

**gzip** [*opts*] <*filepattern*> ·**Compress a file or files**
-1..9 : Set compression level. 9=highest, 1=lowest.
-d :Decompress file. Same as the **gunzip** command.
-l :List the statistics for a compressed file.

**ln** [*opts*] <*tofile*> <*linkfile*> ·**Create a sym/hard link**
-s :Create a symbolic link between files. (alias name)
-f :Force creation, even if the link file exists.

**ls** [*opts*] [*pattern*] ·**List file and directory entries**
-a :List all files including . and ..
-d :List directories themselves, not their contents.
-l :Long list. Shows permissions and modified time.
-r :Recusively list files in directories.
-S :Sort output by file size.
-u :Sort by the last access time.
-X :Sort by filename extension.
-1 :Print output files one per line.
--time=atime :Show last access timestamp for file.

**mkdir** [*opts*] <*dirname*> ·**Make a new directory**
-m *mode*: Set the initial permissions to *mode*.
-p :Create parent directories if they don't exist.

**mv** [-i] <*frompattern*> <*tofile*> ·**Move/Rename a file**
-i :Interactive move. Prompt before moving files.

**rm** [*opts*] <*filepattern*> ·**Remove a file**
-f :Force removal. Don't ask if it's ok to remove.
-i :Interactive remove. Prompt before each file.
-r :Recusively delete directories an their contents.

**tar** [*opts*] [*tarfile*] [*pattern*] ·**Create an archive**
c :Create mode. Create a tar archive.
x :Extract mode. Untar archive contents.
t :List mode. List the contents of the archive.
f :Specify a tarfile to use.
v :Verbose mode. Show files being added or untared.
z :Compress. Filter input/output through gzip.

**touch** [*opts*] <*pattern*> ·**Update the timestamp on a file**
-a :Only change the access time on the file.
-t :Specify a timestamp to use instead of current time

# Informational

**cat** [*opts*] [*filepattern*] ·**Print file contents on STOUT**
-E :Display a $ at the end of each line.
-T :Show tabs as ^I.
-v :Show non-printing characters.

**date** [*opts*] ·**Print or set the system date and time**
--date=*STRING* :display time described by *STRING*.
--set=*STRING* :set time described by *STRING*.

**dmesg** [*opts*] ·**Print or control the kernel ring buffer**
-c :Clear the contents of the ring buffer.

**file** [*opts*] [*filepattern*] ·**Determine the file type**
-z :Try to look inside compressed files.

**finger** [*opts*] [*userpattern*] ·**Show info about system users**
-m :Match the exact username specified.

**free** [*opts*] ·**Display free and used memory in the system**
-b :Display the information in bytes.

**hexdump** [*opts*] ·**Show all the characters of a file**
-c :Display the input offset in hexidecimal

**last** [*opts*] [*username*] ·**Show last system logins for users**
-*num* :Show last *num* of sessions.
-a :Display the hostname in the last column.
-d :Translates IP numbers to their hostname.
-f <file> :Use file as last log.

**less** [*opts*] [*filepattern*] ·**View a file a page at a time**
-i :Do case insensitive searching.
-S :Don't wrap long lines.

**man** [*opts*] [*section*] <*manpage*> ·**View online manual pages.**
-a :View all available manual pages for name.
-k *string* :Search for the specified *string*.

**md5sum** [*opts*] [*filepattern*] ·**Show the uniqueness of files**

**ps** [*opts*] ·**Show what processes are running on the system**
a :Select all processes on a terminal.
u :Display user oriented format. More columns.
x :Select processes without a controlling TTY.
w :Show an extra line of process entry per w.
Ex: ps auxwww =Displays all process information on system.

**quota** [*opts*] [*user*] ·**Display disk usage and limits**
-v :Display filesystems where no quota is set.

**time** [*opts*] [*command*] ·**Show resource usage for a command**

**top** [*opts*] ·**Display top CPU processes every X seconds**
-d *sec* :Set the delay to *sec* seconds before refreshing.

**umask** [*opts*] [*mode*] ·**Set the default file permissions**
-S :Show current symbolic umask.

**uname** [*opts*] ·**Show OS and system information**
-a :Show everything

**uptime** ·**Show system uptime and load**

**w** [*opts*] [*user*] ·**Show who is logged in/what they are doing**

**whereis** [*command*] ·**Locate the related files for a command**

**which** [*command*] ·**Show full path to the specified command**

**who** [*opts*] [*args*] ·**Show who is logged in**

# Text Filtering / Mutative

**awk** [*opts*] [*exp*] ·**pattern scanning and processing language**
-F*fs* :Set the field separator for commands.
Ex: cat access_log | awk {'print $1'} (prints hostnames)
Do a 'man awk' for more information and examples.

**comm** [*opts*] [*file1*] [*file2*] ·**Compare two sorted files**
-1 :Suppress lines unique to left file.
-2 :Suppress lines unique to right file.
-3 :Supress lines unique to both files.

**csplit** [*opts*] [*file*] [*pattern*] ·**Split a file on context**
-f *prefix* :Use *prefix* instead of xx in output filenames.
-n *digits* :Use *digits* number of digits instead of 2.
-z :Remove empty output files.
Ex: csplit mailspoolfile "/^From /" {*}

**cut** [*opts*] [*filepattern*] ·**Remove sections from each line**
-c *range* :Output only the characters in *range*
Ex: cut -c 1-80 *file* (truncate lines at 80 characters)

**diff** [*opts*] [*file1*] [*file2*] ·**Differentiate two files**
Ex: diff program-old.c program.c > program.patch

**echo** [*opts*] [*string*] ·**Print a line of text**
-e :Enable interpretation of backslashed sequences.
-n :Don't automatically insert a newline character.

**grep** [*opts*] [*pattern*] [*file*] ·**Print lines matching pattern**
-B *num* :Print *num* lines of leading context on matches.
-C *num* :Print *num* lines of trailing context on matches.
-E :Interpret *pattern* as an extended regular expression
-i :Do case insensitve matching.
-l :Just print the files that match the *pattern*.
-r :Read all files under each directory recursively.
-v :Print the lines that don't match *pattern*.

**head** [*opts*] [*file*] ·**Print the first part of a file**
-n *num* :Print the first *num* lines instead of the first 10.

**nl** [*opts*] [*file*] ·**Number the lines of a file**

**patch** [*opts*] <[*patchfile*] ·**Patch a file using a diff file**

**sed** [*expression*] [*file*] ·**Stream editor**
Ex: cat file | sed 's/*frompattern*/*topattern*/' > output

**sort** [*opts*] [*file*] ·**Sort lines of text files**
-n :Compare according to string numerical value.
-r :Reverse the result of comparisons.

**split** [*opts*] [*file*] ·**Split a file into pieces**
-l *num* :Put *num* lines per output file.

**tail** [*opts*] [*file*] ·**Print the last lines of a file**
-f :Output appended data as the file grows.
-n *num* :Print last *num* lines of a file instead of last 10

**tr** [*opts*] <*set1*> [*set2*] ·**Translate char. from set1 to set2**
Ex: cat index.html | tr A-Z a-z > index-new.html

**uniq** [*opts*] [*input*] [*output*] ·**Remove duplicate lines**
-c :Prefix lines with number of occurances.
-d :Only print duplicated lines.
-u :Only print unique lines.

**wc** [*opts*] [*file*] ·**Print the number of lines in files, etc.**
-m :Print the character count.
-l :Print the line count.
-w :Print the word count.
-L :Print the length of the longest line.

# Network

**ifconfig** [*devicename*] [*action*] [*options*]

**ipchains** [*opts*] ·**Manip. ipchains firewall(kernel 2.2+)**

**iptables** [*opts*] ·**Manip. iptables firewall(kernel 2.4+)**

**mail** [*opts*] [*address*] ·**Send mail from the command line**
-s *subject* :Specify the subject as *subject*.
-c *list* :Send carbon copy to *list* of users.
-b *list* :Send blind carbon copy to *list* of users.
Ex: echo "Meet me at noon." | mail -s "Reminder" -c \
bob@company.com,suzy@company.com jack@company.com

**netstat** [*opts*] ·**Print network connections and info**
-a :Show both listening and non-listening sockets.
-n :Do not attempt to resolve IP addresses.
-t :Only show tcp socket connection table.

**ping** [*opts*] [*host*] ·**Send ICMP packets to network hosts**
-c *count* :Send *count* number of packets and then quit.
-i *sec* :Wait *sec* seconds between sending packets.

**route** [*opts*] [*target*] ·**Show/Manipulate IP routing table**
-n :Show numerical addresses instead of hostnames.

**scp** [*opts*] [[*host:*]*fromfile*] [[*host:*]*to*] · **Secure copy**
-C :Compresses the data that is sent over the session.
-r :Recursively copy directories.

**ssh** [*opts*] [[*user@*]*host*] [*command*] · **Secure shell/login**
-C :Compresses the data that is sent over the session.

**sniffit** [*opts*] ·**Record TCP network traffic**
-i :Interactive mode. Shows all traffic.
-l <*length*> :Set the maximum sniff length to *length*.
-p <*port*> :Specify the port number on the target host.
-s <*source*> :Set the source IP from which packets come.
-t <*target*> :Set the IP to which packets are going.

**tcpdump** [*opts*] [*expression*] ·**Dump traffic on a network**

**telnet** [*opts*] [*host*] [*port*] ·**Open TCP socket to a host**
-n <file> : Opens *file* for recording trace information.
-x :Turns on encryption of the data stream if possible.

**traceroute** [*opts*] [*host*] ·**Show the route packets take**
-n :Don't do DNS lookups of the IP addresses.

**wget** [*opts*] [*URL*] ·**Make a HTTP request from the shell**
-r :Recursive get the URL and all it's links.
-k :Convert the non-relative links to relative ones.

**whois** [*opts*] <*arg*[*@server*]> ·**Query a whois database**
Ex: whois domain.com
whois domain.com@whois.networksolutions.com
whois 127.0.0.1@arin.net

# Bash Shell

**>** ·**Send STDOUT to a file. overwrite/create a file**
Ex: ls -l > list-of-files.txt
>newfile

**>>** ·**Send STDOUT to a file, appending to te end of the file**
Ex: ps aux > pslog.txt
date >> pslog.txt

**|** ·**Send the STDOUT from a command to the STDIN of another**
Ex: cat listofnames | sort
cat access_log | awk {'print $1'} | sort | uniq

**2>** ·**Send STDERR to a file, overwriting the filename**
Ex: startx 2> .errorlog

**alias** ·**Create a command alias in the shell**
Ex: alias ls='ls -la --color=auto'

**cd** [-] [*directory*] ·**Change the current working directory**
- :Change to the previous directory you were in.

**clear** ·**Clear the terminal display**

**env** [*opts*] [*command*] ·**Run command in modified environment**

**export** [*opts*] [*variable*] ·**Export an environment variable**
Ex: export TERM USERNAME PS1 MAILSPOOL
export TERM=vt100

**for** ·**Execute sequence of commands for a list of items**
Ex: for i in *.mp3 ; do mpg123 $i ; done
for n in 1 2 3 4 ; do mkdir $n ; done

**history** ·**Show the command history up til now**

**nice** [*opts*] [*command*] ·**Set the OS process priority**
Ex: nice 19 gzip access_log (lowest priority on Linux)
Ex: nice -20 kswapd (real time priority on Linux)

**pwd** ·**Print out the current working directory**

**renice** [*opts*] <*arg*> ·**Change priority of a running process**
-p *PID*: Specify a process id to renice
Ex: $ ps auxw | grep gzip
*6319 ? S 0:20 gzip bigfile.txt*
$ renice 19 -p 6319

**reset** ·**Initializes the terminal as if you just logged in**

**set** ·**Set a shell option or variable (run 'help set')**

**sleep** ·**Pause for specified period before continuing**
Ex: ps aux ; sleep 3600 ; ps aux

**umask** ·**Set the default file permissions**
Ex: umask 022 (files will be created 644 by default.)

**while** ·**A loop that runs commands while a condition is true**
Ex: while (true) ; do ps auxw ; sleep 1m ; done > pslog

**xargs** [*opts*] [*command*] ·**Execute a command for each arg**
-n *number* :How many arguments to give each command run.
-p :Prompt the user before each command is run.

# Admin

**adduser** [*opts*] <*username*> ·**Add a user to the local system**
-d <*dir*> : Set the home directory for the user to *dir*.
-g <*group*> : Set the primary group for the user to *group*.
-G <*group,group,…*> : Set additional groups for the user.
-s <*shell*> : Set the default shell for the user to *shell*.

**chfn** ·**Change the finger information for a user**

**chsh** ·**Change the shell used for the user**

**edquota** [*opts*] <*user*> ·**Edit a user's or group's quota**
-g : Edit the group quota instead of user quota.

**kill** [-*signal*] <*pid*> ·**Terminate a process/Send it a signal**
-HUP,-1 : Signal usually makes process to reread config.
-9 :Send a SIGKILL, process must die.
-l :Print a list of signal names and numbers.

**killall** [-*signal*] [*name*]·**Kill processes by name**
-e :Require an exact name of a process.
-i :Interactively ask for confirmation before killing.

**ldd** [*opts*] [*program*] ·**Show a programs library dependencies**

**mount** [*opts*] <*device*> [*mountpoint*] ·**Mount a file system**
-o <*opts*> : Specify options for mounting. Listed below.
loop - Mount a disk file such as a CD-ROM image or floppy image.
remount - Remount the filesystem with new options
ro, rw - Mount filesystem in readonly or read-write mode.
user - Allow normal users to mount this filesystem.
-r :Mount the filesystem read-only. Same as '-o ro'
-t <*vfstype*> :Specify the type of filesystem to mount.
ext2, ext3 - Native Linux partition types.
vfat - Windows 9x 32-bit partition type.
iso9660 - CD-ROM filesystem.
nfs - Network remote filesystem.

**passwd** [*opts*] [*username*]·**Change user's system password**
-l :Lock the password for the account.
-u :Unlock the password for the account.
-S :Show the status of the password for the account.

**su** [-] [*username*] ·**Switch users or login as the superuser**
- :Make shell a login shell
-c <*command*> : Run *command* as *username*.

**umount** [*opts*] [*path/device*]·**Unmount a mounted filesystem**
-f :Force unmounting (in case of unreachable NFS system).

†Most commands accept the use of –v, -h or –help for displaying verbose information and help information.
‡Commands and options displayed in red can only be used by the superuser (root).