

By Jitendra Narayan, BioinformaticsOnline.com

## Linux Commands Line Cheat Sheet for Bioinformaticians

This Command Line Cheat Sheet is prepared for the benefit of Bioinformatics and Computational Biologist. Mail you valuable comments on [info@bioinformaticsonline.com](mailto:info@bioinformaticsonline.com)

COMMAND	DESCRIPTION
<ul style="list-style-type: none"> <li>• <b>System Information</b></li> </ul>	
arch	show architecture of machine(1)
uname -m	show architecture of machine(2)
uname -r	show used kernel version
dmidecode -q	show hardware system components - (SMBIOS / DMI)
hdparm -i /dev/hda	displays the characteristics of a hard-disk
hdparm -tT /dev/sda	perform test reading on a hard-disk
cat /proc/cpuinfo	show information CPU info
cat /proc/interrupts	show interrupts
cat /proc/meminfo	verify memory use
cat /proc/swaps	show file(s) swap
cat /proc/version	show version of the kernel
cat /proc/net/dev	show network adpters and statistics
cat /proc/mounts	show mounted file system(s)
lspci -tv	display PCI devices
lsusb -tv	show USB devices
date	show system date
cal 2007	show the timetable of 2007
date 041217002007.00	set date and time - MonthDayhoursMinutesYear.Seconds
clock -w	save date changes on BIOS

- **Shutdown and Restart**

shutdown -h now	shutdown system(1)
init 0	shutdown system(2)
telinit 0	shutdown system(3)
shutdown -h hours:minutes &	planned shutdown of the system
shutdown -c	cancel a planned shutdown of the system
shutdown -r now	reboot(1)
reboot	reboot(2)
logout	leaving session

- **Files and Directory**

cd /home	enter to directory '/ home'
----------	-----------------------------

By Jitendra Narayan, BioinformaticsOnline.com

<code>cd ..</code>	go back one level
<code>cd ../../</code>	go back two levels
<code>cd</code>	go to home directory
<code>cd ~user1</code>	go to home directory
<code>cd -</code>	go to previous directory
<code>pwd</code>	show the path of work directory
<code>ls</code>	view files of directory
<code>ls -F</code>	view files of directory
<code>ls -l</code>	show details of files and directory
<code>ls -a</code>	show hidden files
<code>ls *[0-9]*</code>	show files and directory containing numbers
<code>tree</code>	show files and directories in a tree starting from root(1)
<code>lstree</code>	show files and directories in a tree starting from root(2)
<code>mkdir dir1</code>	create a directory called 'dir1'
<code>mkdir dir1 dir2</code>	create two directories simultaneously
<code>mkdir -p /tmp/dir1/dir2</code>	create a directory tree
<code>rm -f file1</code>	delete file called 'file1'
<code>rmdir dir1</code>	delete directory called 'dir1'
<code>rm -rf dir1</code>	remove a directory called 'dir1' and contents recursively
<code>rm -rf dir1 dir2</code>	remove two directories and their contents recursively
<code>mv dir1 new_dir</code>	rename / move a file or directory
<code>cp file1 file2</code>	copying a file
<code>cp dir/* .</code>	copy all files of a directory within the current work directory
<code>cp -a /tmp/dir1 .</code>	copy a directory within the current work directory
<code>cp -a dir1 dir2</code>	copy a directory
<code>ln -s file1 lnk1</code>	create a symbolic link to file or directory
<code>ln file1 lnk1</code>	create a physical link to file or directory
<code>touch -t 0712250000 file1</code>	modify timestamp of a file or directory - (YYMMDDhhmm)
<code>file file1</code>	outputs the mime type of the file as text
<code>iconv -l</code>	lists known encodings
<code>iconv -f fromEncoding -t toEncoding inputFile &gt; outputFile</code>	creates a new from the given input file by assuming it is encoded in fromEncoding and converting it to toEncoding.
<code>find . -maxdepth 1 -name *.jpg -print -exec convert "{}" -resize 80x60 "thumbs/{" " \;</code>	batch resize files in the current directory and send them to a thumbnails directory (requires convert from Imagemagick)
<ul style="list-style-type: none"> <li>• <b>File search</b></li> </ul>	
<code>find / -name file1</code>	search file and directory into root filesystem from '/'
<code>find / -user user1</code>	search files and directories belonging to 'user1'

By Jitendra Narayan, BioinformaticsOnline.com

<pre>find /home/user1 -name \*.bin find /usr/bin -type f -atime +100 find /usr/bin -type f -mtime -10 find / -name \*.rpm -exec chmod 755 '{}' \;</pre>	<pre>search files with '.bin' extension within directory '/home/user1' search binary files are not used in the last 100 days search files created or changed within 10 days search files with '.rpm' extension and modify permits search files with '.rpm' extension ignoring removable partitions as cdrom, pen-drive, etc....</pre>
<pre>find / -xdev -name \*.rpm</pre>	<pre>search files with '.rpm' extension ignoring removable partitions as cdrom, pen-drive, etc....</pre>
<pre>locate \*.ps whereis halt which halt</pre>	<pre>find files with the '.ps' extension - first run 'updatedb' command show location of a binary file, source or man show full path to a binary / executable</pre>

- **Mounting a Filesystem**

<pre>mount /dev/hda2 /mnt/hda2</pre>	<pre>mount disk called hda2 - verify existence of the directory '/mnt/hda2'</pre>
<pre>umount /dev/hda2</pre>	<pre>unmount disk called hda2 - exit from mount point '/mnt/hda2' first</pre>
<pre>fuser -km /mnt/hda2</pre>	<pre>force umount when the device is busy</pre>
<pre>umount -n /mnt/hda2</pre>	<pre>run umount without writing the file /etc/mstab - useful when the file is read-only or the hard disk is full</pre>
<pre>mount /dev/fd0 /mnt/floppy</pre>	<pre>mount a floppy disk</pre>
<pre>mount /dev/cdrom /mnt/cdrom</pre>	<pre>mount a cdrom / dvdrom</pre>
<pre>mount /dev/hdc /mnt/cdrecorder</pre>	<pre>mount a cdrw / dvdrom</pre>
<pre>mount /dev/hdb /mnt/cdrecorder</pre>	<pre>mount a cdrw / dvdrom</pre>
<pre>mount -o loop file.iso /mnt/cdrom</pre>	<pre>mount a file or iso image</pre>
<pre>mount -t vfat /dev/hda5 /mnt/hda5</pre>	<pre>mount a Windows FAT32 file system</pre>
<pre>mount /dev/sda1 /mnt/usbdisk</pre>	<pre>mount a usb pen-drive or flash-drive</pre>
<pre>mount -t smbfs -o username=user,password=pass //WinClient/share /mnt/share</pre>	<pre>mount a windows network share</pre>

- **Disk Space**

<pre>df -h</pre>	<pre>show list of partitions mounted</pre>
<pre>ls -lSr   more</pre>	<pre>show size of the files and directories ordered by size</pre>
<pre>du -sh dir1</pre>	<pre>estimate space used by directory 'dir1'</pre>
<pre>du -sk *   sort -rn</pre>	<pre>show size of the files and directories sorted by size</pre>
<pre>rpm -q -a --qf '%10{SIZE}t%{NAME}n'   sort -k1,1n</pre>	<pre>show space used by rpm packages installed sorted by size fedora, redhat and like)</pre>
<pre>dpkg-query -W -f='\${Installed-Size;10}t\${Package}n'   sort -k1,1n</pre>	

By Jitendra Narayan, BioinformaticsOnline.com

show space used by deb packages installed sorted by size  
(ubuntu, debian and like)

- **Users and Groups**

<code>groupadd group_name</code>	create a new group
<code>groupdel group_name</code>	delete a group
<code>groupmod -n new_group_name old_group_name</code>	rename a group
<code>useradd -c "Name Surname" -g admin -d /home/</code>	
<code>user1 -s /bin/bash user1</code>	create a new user belongs "admin" group
<code>useradd user1</code>	create a new user
<code>userdel -r user1</code>	delete a user ('-r' eliminates home directory)
<code>usermod -c "User FTP" -g system -d /ftp/user1 -s /bin/</code>	
<code>nologin user1</code>	change user attributes
<code>passwd</code>	change password
<code>passwd user1</code>	change a user password (only by root)
<code>chage -E 2005-12-31 user1</code>	set deadline for user password
<code>pwck</code>	check correct syntax and file format of '/etc/passwd' and users existence
<code>grpck</code>	check correct syntax and file format of '/etc/group' and groups existence
<code>newgrp group_name</code>	log in to a new group to change default group of newly created files

\* **Permits on Files (use "+" to set permissions and "-" to remove)**

<code>ls -lh</code>	show permits
<code>ls /tmp   pr -T5 -W\$COLUMNS</code>	divide terminal into 5 columns
<code>chmod ugo+rwx directory1</code>	set permissions reading (r), write (w) and (x) access to users owner (u) group (g) and others (o)
<code>chmod go-rwx directory1</code>	remove permits reading (r), write (w) and (x) access to user group (g) and others (o)
<code>chown user1 file1</code>	change owner of a file
<code>chown -R user1 directory1</code>	change user owner of a directory and all the files and directories contained inside
<code>chgrp group1 file1</code>	change group of files
<code>chown user1:group1 file1</code>	change user and group ownership of a file
<code>find / -perm -u+s</code>	view all files on the system with SUID configured
<code>chmod u+s /bin/file1</code>	set SUID bit on a binary file - the user that running that file gets same privileges as owner
<code>chmod u-s /bin/file1</code>	disable SUID bit on a binary file
<code>chmod g+s /home/public</code>	set SGID bit on a directory - similar to SUID but for directory
<code>chmod g-s /home/public</code>	disable SGID bit on a directory

By Jitendra Narayan, BioinformaticsOnline.com

chmod o+t /home/public	set STIKY bit on a directory - allows files deletion only to legitimate owners
chmod o-t /home/public	disable STIKY bit on a directory

- **Special Attributes on files (use "+" to set permissions and "-" to remove)**

chattr +a file1	allows write opening of a file only append mode
chattr +c file1	allows that a file is compressed / decompressed automatically by the kernel
chattr +d file1	makes sure that the program ignores Dump the files during backup
chattr +i file1	makes it an immutable file, which can not be removed, altered, renamed or linked
chattr +s file1	allows a file to be deleted safely
chattr +S file1	makes sure that if a file is modified changes are written in synchronous mode as with sync
chattr +u file1	allows you to recover the contents of a file even if it is canceled
lsattr	show specials attributes

- **Archives and compressed files**

bunzip2 file1.bz2	decompress a file called 'file1.bz2'
bzip2 file1	compress a file called 'file1'
gunzip file1.gz	decompress a file called 'file1.gz'
gzip file1	compress a file called 'file1'
gzip -9 file1	compress with maximum compression
rar a file1.rar test_file	create an archive rar called 'file1.rar'
rar a file1.rar file1 file2 dir1	compress 'file1', 'file2' and 'dir1' simultaneously
rar x file1.rar	decompress rar archive
unrar x file1.rar	decompress rar archive
tar -cvf archive.tar file1	create a uncompressed tarball
tar -cvf archive.tar file1 file2 dir1	create an archive containing 'file1', 'file2' and 'dir1'
tar -tf archive.tar	show contents of an archive
tar -xvf archive.tar	extract a tarball
tar -xvf archive.tar -C /tmp	extract a tarball into / tmp
tar -cvfj archive.tar.bz2 dir1	create a tarball compressed into bzip2
tar -xvfj archive.tar.bz2	decompress a compressed tar archive in bzip2
tar -cvfz archive.tar.gz dir1	create a tarball compressed into gzip
tar -xvfz archive.tar.gz	decompress a compressed tar archive in gzip
zip file1.zip file1	create an archive compressed in zip
zip -r file1.zip file1 file2 dir1	compress in zip several files and directories simultaneously
unzip file1.zip	decompress a zip archive

By Jitendra Narayan, BioinformaticsOnline.com

- **Red Hat and like**

<code>rpm -ivh package.rpm</code>	install a rpm package
<code>rpm -ivh --nodeeps package.rpm</code>	install a rpm package ignoring dependencies requests
<code>rpm -U package.rpm</code>	upgrade a rpm package without changing configuration files
<code>rpm -F package.rpm</code>	upgrade a rpm package only if it is already installed
<code>rpm -e package_name.rpm</code>	remove a rpm package
<code>rpm -qa</code>	show all rpm packages installed on the system
<code>rpm -qa   grep httpd</code>	show all rpm packages with the name "httpd"
<code>rpm -qi package_name</code>	obtain information on a specific package installed
<code>rpm -qg "System Environment/Daemons"</code>	show rpm packages of a group software
<code>rpm -ql package_name</code>	show list of files provided by a rpm package installed
<code>rpm -qc package_name</code>	show list of configuration files provided by a rpm package installed
<code>rpm -q package_name --whatrequires</code>	show list of dependencies required for a rpm packet
<code>rpm -q package_name --whatprovides</code>	show capability provided by a rpm package
<code>rpm -q package_name --scripts</code>	show scripts started during installation / removal
<code>rpm -q package_name --changelog</code>	show history of revisions of a rpm package
<code>rpm -qf /etc/httpd/conf/httpd.conf</code>	verify which rpm package belongs to a given file
<code>rpm -qp package.rpm -l</code>	show list of files provided by a rpm package not yet installed
<code>rpm --import /media/cdrom/RPM-GPG-KEY</code>	import public-key digital signature
<code>rpm --checksig package.rpm</code>	verify the integrity of a rpm package
<code>rpm -qa gpg-pubkey</code>	verify integrity of all rpm packages installed
<code>rpm -V package_name</code>	check file size, permissions, type, owner, group, MD5 checksum and last modification
<code>rpm -Va</code>	check all rpm packages installed on the system - use with caution
<code>rpm -Vp package.rpm</code>	verify a rpm package not yet installed
<code>rpm2cpio package.rpm   cpio --extract --makedirectories *bin*</code>	extract executable file from a rpm package
<code>rpm -ivh /usr/src/redhat/RPMS/`arch`/package.rpm</code>	install a package built from a rpm source
<code>rpmbuild --rebuild package_name.src.rpm</code>	build a rpm package from a rpm source

- **YUM packages updaters (Fedora, RedHat and like)**

<code>yum install package_name</code>	download and install a rpm package
<code>yum localinstall package_name.rpm</code>	PM, and try to resolve all the dependencies for you using your repositories.
<code>yum update package_name.rpm</code>	update all rpm packages installed on the system

By Jitendra Narayan, BioinformaticsOnline.com

yum update package_name	upgrade a rpm package
yum remove package_name	remove a rpm package
yum list	list all packages installed on the system
yum search package_name	find a package on rpm repository
yum clean packages	clean up rpm cache erasing downloaded packages
yum clean headers	remove all files headers that the system uses to resolve dependency
yum clean all	remove from the cache packages and headers files

- **DEB packages (Debian, Ubuntu and like)**

dpkg -i package.deb	install / upgrade a deb package
dpkg -r package_name	remove a deb package from the system
dpkg -l	show all deb packages installed on the system
dpkg -l   grep httpd	show all deb packages with the name "httpd"
dpkg -s package_name	obtain information on a specific package installed on system
dpkg -L package_name	show list of files provided by a package installed on system
dpkg --contents package.deb	show list of files provided by a package not yet installed
dpkg -S /bin/ping	verify which package belongs to a given file

- **APT packages updaters (Debian, Ubuntu e like)**

apt-get install package_name	install / upgrade a deb package
apt-cdrom install package_name	install / upgrade a deb package from cdrom
apt-get update	update the package list
apt-get upgrade	upgrade all of the installed packages
apt-get remove package_name	remove a deb package from system
apt-get check	verify correct resolution of dependencies
apt-get clean	clean up cache from packages downloaded
apt-cache search searched-package	returns list of packages which corresponds string "searched-packages"
cat file1	view the contents of a file starting from the first row
tac file1	view the contents of a file starting from the last line
more file1	view content of a file along
less file1	similar to 'more' command but which allows backward movement in the file as well as forward movement
head -2 file1	view first two lines of a file
tail -2 file1	view last two lines of a file
tail -f /var/log/messages	view in real time what is added to a file
cat file1 file2 ...   command <> file1_in.	

By Jitendra Narayan, BioinformaticsOnline.com

<pre>txt_or_file1_out.txt</pre>	<pre>general syntax for text manipulation using PIPE, STDIN and STDOUT</pre>
<pre>cat file1   command( sed, grep, awk, grep, etc...) &gt; result.txt</pre>	<pre>general syntax to manipulate a text of a file, and write result to a new file</pre>
<pre>cat file1   command( sed, grep, awk, grep, etc...) &gt;&gt; result.txt</pre>	<pre>general syntax to manipulate a text of a file and append result in existing file</pre>
<pre>grep Aug /var/log/messages look up words "Aug" on file '/var/log/ messages'</pre>	<pre>look up words that begin with "Aug" on file '/' var/log/messages'</pre>
<pre>grep ^Aug /var/log/messages</pre>	<pre>look up words that begin with "Aug" on file '/' var/log/messages'</pre>
<pre>grep [0-9] /var/log/messages</pre>	<pre>select from file '/var/log/messages' all lines that contain numbers</pre>
<pre>grep Aug -R /var/log/* sed 's/string1/string2/g' example.txt</pre>	<pre>search string "Aug" at directory '/var/log' and below replace "string1" with "string2" in example.txt</pre>
<pre>sed '/^\$/d' example.txt</pre>	<pre>remove all blank lines from example.txt</pre>
<pre>sed '/ *#/d; /^\$/d' example.txt</pre>	<pre>remove comments and blank lines from example.txt</pre>
<pre>echo 'esempio'   tr '[:lower:]' '[:upper:]'</pre>	<pre>convert from lower case in upper case</pre>
<pre>sed -e '1d' result.txt</pre>	<pre>eliminates the first line from file example.txt</pre>
<pre>sed -n '/string1/p'</pre>	<pre>view only lines that contain the word "string1"</pre>
<pre>sed -e 's/ *\$//' example.txt</pre>	<pre>remove empty characters at the end of each row</pre>
<pre>sed -e 's/string1//g' example.txt</pre>	<pre>remove only the word "string1" from text and leave intact all</pre>
<pre>sed -n '1,5p;5q' example.txt</pre>	<pre>view from 1th to 5th row</pre>
<pre>sed -n '5p;5q' example.txt</pre>	<pre>view row number 5</pre>
<pre>sed -e 's/00*/0/g' example.txt</pre>	<pre>replace more zeros with a single zero</pre>
<pre>cat -n file1</pre>	<pre>number row of a file</pre>
<pre>cat example.txt   awk 'NR%2==1'</pre>	<pre>remove all even lines from example.txt</pre>
<pre>echo a b c   awk '{print \$1}'</pre>	<pre>view the first column of a line</pre>
<pre>echo a b c   awk '{print \$1,\$3}'</pre>	<pre>view the first and third column of a line</pre>
<pre>paste file1 file2</pre>	<pre>merging contents of two files for columns</pre>
<pre>paste -d '+' file1 file2</pre>	<pre>merging contents of two files for columns with '+' delimiter on the center</pre>
<pre>sort file1 file2</pre>	<pre>sort contents of two files</pre>
<pre>sort file1 file2   uniq</pre>	<pre>sort contents of two files omitting lines repeated</pre>
<pre>sort file1 file2   uniq -u</pre>	<pre>sort contents of two files by viewing only unique line</pre>
<pre>sort file1 file2   uniq -d</pre>	<pre>sort contents of two files by viewing only duplicate line</pre>
<pre>comm -1 file1 file2</pre>	<pre>compare contents of two files by deleting only unique lines from 'file1'</pre>
<pre>comm -2 file1 file2</pre>	<pre>compare contents of two files by deleting only unique lines from 'file2'</pre>
<pre>comm -3 file1 file2</pre>	<pre>compare contents of two files by deleting only the lines that appear on both files</pre>



By Jitendra Narayan, BioinformaticsOnline.com

- **Character set and Format file conversion**

dos2unix filedos.txt fileunix.txt	convert a text file format from MSDOS to UNIX
unix2dos fileunix.txt filedos.txt	convert a text file format from UNIX to MSDOS
recode ..HTML < page.txt > page.html	convert a text file to html
recode -l   more	show all available formats conversion

- **Analysis**

badblocks -v /dev/hda1	check bad blocks in disk hda1
fsck /dev/hda1	repair / check integrity of linux filesystem on disk hda1
fsck.ext2 /dev/hda1	repair / check integrity of ext2 filesystem on disk hda1
e2fsck /dev/hda1	repair / check integrity of ext2 filesystem on disk hda1
e2fsck -j /dev/hda1	repair / check integrity of ext3 filesystem on disk hda1
fsck.ext3 /dev/hda1	repair / check integrity of ext3 filesystem on disk hda1
fsck.vfat /dev/hda1	repair / check integrity of fat filesystem on disk hda1
fsck.msdos /dev/hda1	repair / check integrity of dos filesystem on disk hda1
dosfsck /dev/hda1	repair / check integrity of dos filesystems on disk hda1

- **Format a Filesystem**

mkfs /dev/hda1	create a filesystem type linux on hda1 partition
mke2fs /dev/hda1	create a filesystem type linux ext2 on hda1 partition
mke2fs -j /dev/hda1	create a filesystem type linux ext3 (journal) on hda1 partition
mkfs -t vfat 32 -F /dev/hda1	create a FAT32 filesystem
fdformat -n /dev/fd0	format a floppy disk
mkswap /dev/hda3	create a swap filesystem

- **SWAP filesystem**

mkswap /dev/hda3	create a swap filesystem
swapon /dev/hda3	activating a new swap partition
swapon /dev/hda2 /dev/hdb3	activate two swap partitions

- **Backup**

dump -0aj -f /tmp/home0.bak /home	make a full backup of directory '/home'
dump -1aj -f /tmp/home0.bak /home	make a incremental backup of directory '/ home'
restore -if /tmp/home0.bak	restoring a backup interactively
rsync -rogpav --delete /home /tmp	synchronization between directories
rsync -rogpav -e ssh --delete /home ip_address:/tmp	

By Jitendra Narayan, BioinformaticsOnline.com

rsync via SSH tunnel

```
rsync -az -e ssh --delete ip_addr:/home/public /home/
local
    synchronize a local directory with a remote directory via ssh and
    compression

rsync -az -e ssh --delete /home/local ip_addr:/home/
public
    synchronize a remote directory with a local directory via ssh and
    compression

dd bs=1M if=/dev/hda | gzip | ssh user@ip_addr 'dd
of=hda.gz'
    make a backup of a local hard disk on remote host via ssh

dd if=/dev/sda of=/tmp/file1
    backup content of the harddrive to a file

tar -Puf backup.tar /home/user
    make an incremental backup of directory '/home/user'

(cd /tmp/local/ && tar c .) | ssh -C user@ip_addr
'cd /home/share/ && tar x -p'
    copy content of a directory on remote directory via ssh

(tar c /home) | ssh -C user@ip_addr 'cd /home/
backup-home && tar x -p'
    copy a local directory on remote directory via ssh

tar cf - | (cd /tmp/backup ; tar xf -)
    local copy preserving permits and links from a directory to
    another

find /home/user1 -name '*.txt' | xargs cp -av --targetdirectory=/
home/backup/ --parents
    find and copy all files with '.txt' extension from a directory to
    another

find /var/log -name '*.log' | tar cv --files-from=- |
bzip2 > log.tar.bz2
    find all files with '.log' extension and make an bzip archive

dd if=/dev/hda of=/dev/fd0 bs=512 count=1
    make a copy of MBR (Master Boot Record) to floppy

dd if=/dev/fd0 of=/dev/hda bs=512 count=1
    restore MBR from backup copy saved to floppy
```

- **CDROM**

```
cdrecord -v gracetime=2 dev=/dev/cdrom -eject
blank=fast -force
    clean a rewritable cdrom

mkisofs /dev/cdrom > cd.iso
    create an iso image of cdrom on disk

mkisofs /dev/cdrom | gzip > cd_iso.gz
    create a compressed iso image of cdrom on disk

mkisofs -J -allow-leading-dots -R -V "Label CD" -isolevel
4 -o ./cd.iso data_cd
    create an iso image of a directory

cdrecord -v dev=/dev/cdrom cd.iso
    burn an ISO image

gzip -dc cd_iso.gz | cdrecord dev=/dev/cdrom -
    burn a compressed ISO image

mount -o loop cd.iso /mnt/iso
    mount an ISO image

cd-paranoia -B
    rip audio tracks from a CD to wav files

cd-paranoia -- "-3"
files
    rip first three audio tracks from a CD to wav
files

cdrecord --scanbus
    scan bus to identify the channel scsi

dd if=/dev/hdc | md5sum
    perform an md5sum on a device, like a CD
```

By Jitendra Narayan, BioinformaticsOnline.com

- **Networking (LAN and WiFi)**

ifconfig eth0	show configuration of an ethernet network card
ifup eth0	activate an interface 'eth0'
ifdown eth0	disable an interface 'eth0'
ifconfig eth0 192.168.1.1 netmask 255.255.255.0	configure IP Address
ifconfig eth0 promisc	configure 'eth0' in promiscuous mode to gather packets (sniffing)
dhclient eth0	active interface 'eth0' in dhcp mode
route -n	show routing table
route add -net 0/0 gw IP_Gateway	configure default gateway
route add -net 192.168.0.0 netmask 255.255.0.0 gw 192.168.1.1	configure static route to reach network 192.168.0.0/16'
route del 0/0 gw IP_gateway	remove static route
echo "1" > /proc/sys/net/ipv4/ip_forward	activate ip routing
hostname	show hostname of system
host <u>www.example.com</u>	lookup hostname to resolve name to ip address and viceversa(1)
nslookup <u>www.example.com</u>	lookup hostname to resolve name to ip address and viceversa(2)
ip link show	show link status of all interfaces
mii-tool eth0	show link status of 'eth0'
ethtool eth0	show statistics of network card 'eth0'
netstat -tup	show all active network connections and their PID
netstat -tupl s	how all network services listening on the system and their PID
tcpdump tcp port 80	show all HTTP traffic
iwlist scan	show wireless networks
iwconfig eth1	show configuration of a wireless network card
hostname	show hostname
host <u>www.example.com</u>	lookup hostname to resolve name to ip address and viceversa
nslookup <u>www.example.com</u>	lookup hostname to resolve name to ip address and viceversa
whois <u>www.example.com</u>	lookup on Whois database

- **Microsoft Windows networks (SAMBA)**

nbtscan ip_addr	netbios name resolution
nmblookup -A ip_addr	netbios name resolution
smbclient -L ip_addr/hostname	show remote shares of a windows host
smbget -Rr smb://ip_addr/share	like wget can download files from a host windows via smb
mount -t smbfs -o username=user,password=pass //WinClient/share /mnt/share	mount a windows network share

By Jitendra Narayan, BioinformaticsOnline.com

- **IPTABLES (firewall)**

<code>iptables -t filter -L</code>	show all chains of filtering table
<code>iptables -t nat -L</code>	show all chains of nat table
<code>iptables -t filter -F</code>	clear all rules from filtering table
<code>iptables -t nat -F</code>	clear all rules from table nat
<code>iptables -t filter -X</code>	delete any chains created by user
<code>iptables -t filter -A INPUT -p tcp --dport telnet -j ACCEPT</code>	allow telnet connections to input
<code>iptables -t filter -A OUTPUT -p tcp --dport http -j DROP</code>	block HTTP connections to output
<code>iptables -t filter -A FORWARD -p tcp --dport pop3 -j ACCEPT</code>	allow POP3 connections to forward chain
<code>iptables -t filter -A INPUT -j LOG --log-prefix "DROP INPUT"</code>	logging sulla chain di input Logging on chain input
<code>iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE</code>	configure a PAT (Port Address Traslation) on eth0 masking outbound packets
<code>iptables -t nat -A PREROUTING -d 192.168.0.1 -p tcp -m tcp --dport 22 -j DNAT --to-destination 10.0.0.2:22</code>	redirect packets addressed to a host to another host

- **Monitoring and debugging**

<code>top</code>	display linux tasks using most cpu
<code>ps -efw</code>	displays linux tasks
<code>ps -e -o pid,args --forest</code>	displays linux tasks in a hierarchical mode
<code>pstree</code>	mostra un albero dei processi sistema Shows a tree system processes
<code>kill -9 ID_Processo</code>	force closure of the process and finish it
<code>kill -1 ID_Processo</code>	force a process to reload configuration
<code>lsof -p \$\$</code>	display a list of files opened by processes
<code>lsof /home/user1</code>	displays a list of open files in a given path system
<code>strace -c ls &gt;/dev/null</code>	display system calls made and received by a process
<code>strace -f -e open ls &gt;/dev/null</code>	display library calls
<code>watch -n1 'cat /proc/interrupts'</code>	display interrupts in real-time
<code>last reboot</code>	show history reboot
<code>lsmod</code>	display kernel loaded
<code>free -m</code>	displays status of RAM in megabytes
<code>smartctl -A /dev/hda</code>	monitoring reliability of a hard-disk through SMART
<code>smartctl -i /dev/hda</code>	check if SMART is active on a hard-disk
<code>tail /var/log/dmesg</code>	show events inherent to the process of booting kernel
<code>tail /var/log/messages</code>	show system events

By Jitendra Narayan, BioinformaticsOnline.com

- **Others useful commands**

<code>apropos ...keyword</code>	display a list of commands that pertain to keywords of a program , useful when you know what your program does, but you don't know the name of the command
<code>man ping</code>	display the on-line manual pages for example on ping command - use '-k' option to find any related commands
<code>whatis ...keyword</code>	displays description of what a program does
<code>mkbootdisk --device /dev/fd0 `uname -r`</code>	create a boot floppy
<code>gpg -c file1</code>	encrypt a file with GNU Privacy Guard
<code>gpg file1.gpg</code>	decrypt a file with GNU Privacy Guard
<code>wget -r <a href="http://www.example.com">www.example.com</a></code>	download an entire web site
<code>wget -c <a href="http://www.example.com/file.iso">www.example.com/file.iso</a></code>	download a file with the ability to stop the download and resume later
<code>echo 'wget -c <a href="http://www.example.com/files.iso">www.example.com/files.iso</a>'</code>	<b>at 09:00</b>
<code>ldd /usr/bin/ssh</code>	start a download at any given time
<code>alias hh='history'</code>	show shared libraries required by ssh program
<code>chsh</code>	set an alias for a command - hh = history
<code>chsh --list-shells</code>	change shell command
<code>who -a</code>	nice command to know if you have to remote into another box show who is logged on, and print: time of last system boot, dead processes, system login processes, active processes spawned by init, current runlevel, last system clock change