

LINUX COMMAND REFERENCE

RECREATED BY: GREGORY V.

```
SYSTEM
$ uname -a => Display linux system information
$ uname -r => Display kernel release information (refer uname command in detail)
$ cat /etc/redhat_release => Show which version of redhat installed
$ cat /proc/cpuinfo => CPU info
$ cat /proc/meminfo => Memory info
$ uptime => Show how long system running + load (learn uptime command)
$ hostname => Show system host name $ hostname -i => Display the IP address of the host
$ last reboot => Show system reboot history (more examples last command)
$ date => Show the current date and time (options of date command)
$ cal => Show this month calendar (what more in cal)
$ w => Display who is online (learn more about w command)
$ whoami => Who you are logged in as (example + screenshots)
$ finger user => Display information about user (many options of finger command)
$ man command => Show the manual for command
```

HARDWARE

```
$ dmesg => Detected hardware and boot messages (dmesg many more options)
$ cat /proc/cpuinfo => CPU model
$ cat /proc/meminfo => Hardware memory
$ cat /proc/interrupts => Lists the number of interrupts per CPU per I/O device
$ lshw => Displays information on hardware configuration of the system
$ lsblk => Displays block device related information in linux (sudo yum install util-linux-ng)
$ free -m => Used and free memory (-m for MB)
$ lspci -tv => Show PCI devices (very useful to find vendor ids)
$ lsusb -tv => Show USB devices (read more lsusb options)
$ lshal => Show a list of all devices with their properties
$ dmidecode => Show hardware info from the BIOS (vendor details)
$ hdparm -I /dev/sda # Show info about disk sda
$ hdparm -Tt /dev/sda # Do a read speed test on disk sda
$ badblocks -s /dev/sda # Test for unreadable blocks on disk sda
```

PROCESS RELATED

```
$ ps # Display your currently active processes (many parameters to learn)
$ ps aux | grep 'telnet' # Find all process id related to telnet process
$ pmap # Memory map of process (kernel,user memory etc)
$ top # Display all running processes
$ kill pid # Kill process with mentioned pid id
$ killall proc # Kill all processes named proc
$ pkill processname # Send signal to a process with its name
$ bg # Resumes suspended jobs without bringing them to foreground (bg and fg command)
$ fg # Brings the most recent job to foreground
$ fg n # Brings job n to the foreground
$ sudo systemctl enable application.service #start service at boot
$ sudo systemctl (start/stop/reload) application.service #control service
$ sudo systemctl list-units # list active services
$ sudo systemctl list-dependencies application.service #show dependencies tree for service
```

STATISTICS

```
$ top => Display and update the top CPU processes
$ mpstat 1 => Display processors related statistics
$ vmstat 2 => Display virtual memory statistics (very useful performance tool)
$ iostat 2 => Display I/O statistics (2sec intervals)
$ tail -n 500 /var/log/messages => Last 10 kernel/syslog messages
$ tcpdump -i eth1 => Capture all packets flows on interface eth1 (useful for network issues)
$ tcpdump -i eth0 'port 80' => Monitor all traffic on port 80 ( HTTP )
$ lsof => List all open files belonging to all active processes.(sysadmin favorite command)
$ lsof -u testuser => List files opened by specific user
$ free -m => Show amount of RAM (daily usage command)
$ watch df -h => Watch changeable data continuously(interesting linux command)
```

USERS

```
$ id => Show the active user id with login and group
$ last => Show last logins on the system
$ who => Show who is logged on the system(real user who logged in)
$ groupadd admin => Add group "admin" (force add existing group)
$ useradd -c "Sam Tomshi" -g admin -m sam => Create user "sam" and add to group "admin"
$ userdel sam => Delete user sam (force,file removal)
$ adduser sam => Add user "sam"
$ usermod => Modify user information(mostly useful for linux system admins)
$ sudo -u root ./setup.sh => Run setup.sh as admin root (also: sudo ls /usr/local/classified)
$ sudo -u root visudo => Modify the etc/sudoers file to grant admin access rights
```

DISK USAGE

```
$ df -h # Show free space on mounted filesystems(commonly used command)
$ df -i # Show free inodes on mounted filesystems
$ fdisk -l # Show disks partitions sizes and types(fdisk command output)
$ du -ah # Display disk usage in human readable form
$ du -sh # Display total disk usage on the current directory
$ findmnt # Displays target mount point for all filesystem (refer type,list,evaluate output)
$ mount device-path mount-point # Mount a device
```

FILE COMMANDS

```
$ ls -al => Display all information about files/ directories
$ pwd => Show current directory path(simple but need every day)
$ mkdir directory-name => Create a directory(create mutiple directory)
$ rm file-name => Delete file(be careful of using rm command)
$ rm -r directory-name => Delete directory recursively
$ rm -f file-name => Forcefully remove file
$ rm -rf directory-name => Forcefully remove directory recursively
$ cp file1 file2 => Copy file1 to file2
$ cp -r dir1 dir2 => Copy dir1 to dir2, create dir2 if it doesn't exist
$ mv file1 file2 => Move files from one place to another
$ ln -s /path/to/file-name link-name => Create symbolic link to file-name
$ touch file => Create or update file (timestamp change)
$ cat > file => Place standard input into file
$ more file => Output the contents of file (help display long tail files)
$ head file => Output the first 10 lines of file (with different parameters)
$ tail file => Output the last 10 lines of file
$ tail -f file => Outputs contents of file as it grows starts with last 10 lines
$ gpg -c file => Encrypt file (how to use gpg)
$ gpg file.gpg => Decrypt file
```

CHANGING FILE PERMISSIONS

```
ls-l => List files with security
file permissions in 4 groups
(-)(R-)(R-)(R-)
EXAMPLE: -RWXr--r-- 1 roman users 0 2013-09-28 19:23 testfile
FIRST GROUP (-) SPECIAL PERMISSIONS:"-" NONE, d - DIRECTORY, l - FILE OR DIR SYMBOLIC LINK
SECOND GROUP (RWX) - RIGHTS FOR OWNER (R-Read)(W-Write)(X-Execute)
THIRD GROUP (R-) - RIGHTS FOR THE GROUP
FOURTH GROUP (R-) - RIGHTS FOR ALL USERS (World / Others)
ROMAN IS FILE OWNER
USERS IS FILE GROUP
CHMOD - RWX [FILE OR FOLDER] (CAN JUST BE -X) REMOVES RIGHTS FROM FILE OWNER
CHMOD + RWX ADDS RIGHTS TO FILE OWNER
CHMOD G +WX ADDS RIGHTS TO GROUP
CHMOD O+R ADDS RIGHTS TO WORLD
CHMOD UOg+r ADDS RIGHTS TO USER,GROUP,WORLD
CHMOD NUMERIC COMMAND -3 NUMBERS "chmod 640 file" 6=OWNER RIGHTS, 4=GROUP RIGHTS, 0 = WORLD RIGHTS
X = 1, W = 2, R = 4 (ADDITIVE: 7 = FULL CONTROL, 6 = READ/WRITE, 5 = READ/EXECUTE, 0 = NO ACCESS)
CHMOD A+X - ADD EXEC BITS TO FILE BUT NOT TOUCH THE OTHER BITS, EX: FILE MAYBE UNREADABLE TO GROUP
CHOWN COMMAND - CHOWN user1:ACCOUNTING testfile = ASSIGN OWNERSHIP TO USER1 AND ACCOUNTING GROUP
CHGRP ACCOUNTING testfile - CHANGES FILE GROUP TO ACCOUNTING
CHOWN GREG testfile - CHANGES OWNER TO GREG
CHOWN -R GREG:ACCOUNTING /home/Roman/FILES - RECURSIVE = GREG OWNERSHIP OF DIR, SUBFOLDERS AND FILES
```

INSTALL PACKAGE

```
$ rpm -i pkgname.rpm # Install rpm based package (install, uninstall, updating, querying ,verify)
$ rpm -e pkgname # Remove package install from source ./configure make make install (what it is)
$ sudo apt-get update => Update the the package index
$ sudo apt-get upgrade => Upgrade packages
$ sudo apt-get update && sudo apt-get upgrade => Combine commands to run both at same time
$ sudo apt-get install nmap => Install nmap network scanner
$ sudo -u root yum install postgresql-server => Get package and install postgresql server
```

NETWORK

```
$ ifconfig -a # Display all network ports and IP address (set mtu,ifconfig deprecated)
$ ifconfig eth0 # Display specific ethernet port IP address and details
$ ip addr show # Display all network interfaces and IP address(iproute2 package better)
$ ip address add 192.168.0.1 dev eth0 # Set IP address
$ ethtool eth0 # Linux tool to show ethernet status (set full duplex , pause parameter)
$ mii-tool eth0 # Linux tool to show ethernet status (more or like ethtool)
$ ping host # Send echo request to test connection (learn sing enhanced ping tool)
$ whois domain # Get who is information for domain
$ dig domain # Get DNS information for domain
$ dig -x host # Reverse lookup host
$ host google.com # Lookup DNS IP address for the name
$ hostname -i # Lookup local IP address (set hostname too)
$ wget file # Download file (very useful other option)
$ netstat -tupl # Listing all active listening ports(tcp,udp,pid)
$ vi /etc/hosts # DNS static host file edit
$ nslookup host.com # Request host information
$ sudo vi /etc/resolv.conf # Edit DNS servers used by host
```

COMPRESSION / ARCHIVES

```
$ tar -cf home.tar home # Create compressed tar named home.tar containing home/
$ tar -xf file.tar # Extract the files from file.tar (tar -xvf = extract, verbose, filename)
$ tar -czf file.tar.gz files # Create a tar with gzip compression
$ gzip file # Compress file and renames it to file.gz (untar gzip file)
```

FILE TRANSFER

```
$ scp -P port filename user@host.net:/home/user/folder/ #Secure copy to host using specific port and send file to specific folder
$ scp nixsavy@server2:/www/*.html /www/tmp # Copy *.html files from remote host to current system /www/tmp folder
$ scp -r nixsavy@server2:/www/www/tmp # Copy all files and folders recursively from remote server to the current system /www/tmp folder
$ rsync -a /home/apps /backup/ # Synchronize source to destination
$ rsync -avz /home/apps linoxide@192.168.10.1:/backup # Synchronize files/directories between the local and remote system with compression enabled
```

DIRECTORY TRAVERSE

```
$ cd .. # To go up one level of the directory tree(simple & most needed)
$ cd # Go to $HOME directory $ cd /test # Change to /test directory
```

SEARCH

```
$ grep pattern files # Search for pattern in files (often used!)
$ grep -r pattern dir # Search recursively for pattern in dir
$ locate file # Find all instances of file
$ find /home/tom -name 'index*' # Find files names that start with "index"
$ find /home -size +10000k # Find files larger than 10000k in /home
```

VI COMMANDS

```
$ vi filename => Edit filename starting at line 1
:x => Quit vi writing file
:q => Quit without write
:w => Write file
:w newfile<RETURN> => Writes file called newfile
I => Insert text before cursor, until <ESC> hit
A => Append text after cursor, until <ESC> hit
O => Open and put text in new line below current line
R => Replace single character under cursor
R => Replace characters, starting with current position
X => Delete single character under cursor
DD => Delete entire line
YY => Copy current line
P => Paste line
/string => Search forward string in text
?string => Search backward string in text
N => More to next search occurrence
N => Search backward to next occurrence
:.= => Gives you current line number
^G => Current line number and total lines in file
```

IRC

```
$ sudo apt-get install irssi => IRC client install
/server
/connect chat.freenode.net => Connect to server
/disconnect
/join #sofreeus => Join channel
/msg => Private message to a user
/msg NickServ REGISTER password youremail@example.com
/msg NickServ SET HIDEEMAIL ON => Hide email address
```

LOGIN (SSH /TELNET / FTP)

```
$ ssh user@host # Connect to host as user (secure)
$ ssh -p port user@host # Connect to host specific port
$ telnet host # Connect to the system using telnet port
$ ftp 123.123.123.123 => Connects to ftp host
FTP COMMANDS: GET [FILE] PUT [FILE] SINGLE FILE
FTP COMMANDS: MGET *.jpg, MPUT *.jpg MULTIPLE FILES
FTP DIR COMMANDS: CD, LCD(local dir), LS, !LS(local dir)
FTP MODE COMMANDS: ASCII, BINARY
```