Linux Commands

Directory Operations

pwd	Show current directory
cd *dir*	Change to directory *dir*
mkdir *dir*	Create a new directory *dir*
rmdir *dir*	Delete directory *dir*
ls *dir*	List contents directory *dir*

Special Directories

	Current directory
..	Up a directory
.	Current directory
~	Home directory
/	Root directory
-	Previous directory

ls Options

-a	all inc. hidden
-l	long format
-t	sort by time
-S	sort by size
-r	reverse order
-R	recursive

File Operations

touch *file*	Create file *file*
cp *file1 file2*	Copy *file1* to *file2*
mv *file1 file2*	Move *file1* to *file2*
rm *file*	Delete *file*
cat *file*	Display contents of *file*
cat *file1 file2*	Concatenate files
less *file*	Display *file* (paginated), q to quit
head *file*	Show first 10 lines
tail *file*	Show last 10 lines
-n *N*	*N* lines
-f	Continuos update

Help

man *cmd*	Manual page for *cmd*
man -k *word*	Search for manual page with *word*
-h	Commands show help when used

File Searching

grep *pattern file*	Search for lines with *pattern* in file
grep -v	Inverted search
grep -r	Recursive search
grep -e *patt* -e *patt*	Multiple patterns
locate *file*	Quick search for *file*
which *cmd*	Find location of binary
find *dir* -name *pattern*	Find file with *pattern* in *dir*

Standard IO Streams

stdin	Input typed on the command line
stdout	Output on the screen
stderr	Errors output on the screen
echo *string*	Write *string* to stdout

Redirection

cmd > *file*	Output of *cmd* to *file*
cmd < *file*	*file* used as input to *cmd*
cmd >> *file*	Append output to *file*
cmd 2> *file*	Write errors to *file*
cmd &> *file*	Errors and stdout to *file*

Pipes and Multiple Commands

cmd1 | *cmd2*	Stdout of *cmd1* is used as input to *cmd2*
cmd1 |&*cmd2*	Stderr of *cmd1* is used as input to *cmd2*
cmdpart1 \ *cmdpart2*	Continue command on next line
cmd1; *cmd2*	Execute *cmd1* then *cmd2*

Processes

ps	Show processes of user
ps -e	Show all processes
ps -fA	Show all processes in detail
top	Show processes in real-time
cmd &	Run command in background
Ctrl-c	Stop (kill) currently active process
Ctrl-z	Suspend currently active process
bg	Place suspended process in background
fg	Bring background process to foreground
kill *pid*	Kill process with process id *pid*
kill -9 *pid*	Kill process *pid* (ungraceful)

Bash Shortcuts

Ctrl-k	Cut line of text
Ctrl-y	Paste line of text
Ctrl-e	Go to end of line
Ctrl-a	Go to start of line
TAB	Autocomplete command/file
TAB-TAB	Show list of possible autocompletes
up arrow	Scroll previous commands
down arrow	Scroll previous commands
history	List recent commands
!!	Repeat last command
!*N*	Execute command *N* from history
!*abc*:p	Print last command starting with *abc*
!*abc*	Execute last command starting with *abc*

Editing Text Files

nano	Text editor
Shortcuts	
Ctrl-o	Save file
Ctrl-x	Close file
Ctrl-r	Open file
Ctrl-k	Cut line of text
Ctrl-u	Paste line of text
Ctrl-d	Delete character
Ctrl-w	Search for text

Text File Operations

wc	Line, word and character count
sort *file*	Sort *file*, line by line
uniq *file*	Display only unique lines of *file*
sed 's/*abc*/*def*/g' *file*	Replace all occurrences of *abc* with *def*, output to stdout
cut -d " " -f *N file*	Display field *N* of space delimited file
cut -d "," -f *M-N file*	Display fields *M-N* of comma delimited *file*

GUI applications via Command line

gedit	Text editor
wireshark	Packet capture and display
eog	Image viewer
evince	PDF viewer
nautilus	File explorer

Administrator Privileges

sudo *cmd*	Execute *cmd* with admin privilege
su *username*	Switch to user *username*

r733

Networking Commands and Files

Interface Configuration
```
ifconfig [<interface>] [up | down]
ifconfig <interface> <ipaddress> netmask
<netmask>
ethtool <interface>
ethtool -k <interface>
File: /etc/network/interfaces
```

Remote Login
```
ssh [-l <username] <address>
```

File Transfer
```
nc -l <port>
nc <ip> <port>
scp <srcfile> <address>:<dstdir>
scp <address>:<srcfile> <dstdir>
wget <url>
iperf -s
iperf -c <address>
```

Packet Capture
```
tcpdump [-i <interface>] [-w <file>]
```

Statistics
```
netstat [-t | -i | -s | -r] [-n]
```

Tracepath
```
tracepath <destination>
```

DNS
```
nslookup <domain> [<dnsserver>]
File: /etc/resolv.conf
File: /etc/hosts
```

ARP
```
arp [-n]
```

DHCP
```
dhclient [<interface>]
File: /var/lib/dhcp/dhclient.leases
```

Web Server
```
apache2ctl [start | stop | restart]
htpasswd <passwordfile> <username>
Dir: /etc/apache2/sites-available/
Dir: /var/www/
```

Ping
```
ping [-c <count>] [-s <packetsize>] [-i <interval>] <destination>
```

Routing
```
route [-n]
route add -net <netaddress> netmask <subnet> [gw <gateway>] [dev <interface>]
route del -net <netaddress> netmask <subnet> [gw <gateway>] [dev <interface>]
route add default gw <gateway>
sysctl net.ipv4.ip_forward=[0 | 1]
File: /proc/sys/net/ipv4/ip_forward
```

Firewall
```
iptables -A <chain> [<options>]
    where <options> include:
    [-s <sourceip>] [-d <destip>] [-i <ininterface>] [-o <outinterface>]
    [-p <protocol>] [--sport <sourceport>] [--dport <destport>]
    [-j <action>]
iptables -D <chain> [<options>]
iptables -L <chain>
iptables -F <chain>
    where <chain> may be: INPUT | OUTPUT | FORWARD
```

File Permissions

Access Rights

r	read file; list the contents of directory
w	write to file; create and remove files in direct.
x	execute file; access files in directory

Subjects

u	user that owns the file
g	users in the file's group
o	other users
a	all users, i.e. the above three)

Commands

`ls -l`	display directory contents
`stat file`	display file status and inode info
`df`	report file system disk usage
`chmod mode file`	change file mode bits, i.e. set permissions

Output of ls -l

E.g.: `-rw-rwx---  drwxr-x--x`
First letter indicates file type:
    d = directory
    - = normal file
Next 9 letters: access right letter indicates the permission is set; - indicates the permission is not set

Modes

Modes can be specified by combining subject with access right

  + Grant with right
  - Remove the right
  = Set the right
Examples:
    u+r
    g-rw
    u+r,g-wx,o-r
    a=rx

Security

Random Numbers

`echo $RANDOM`	Generate random integer [0..32767]
`cat /dev/urandom | xxd -l 1 -b`	Generate 8 bit random value, binary
`openssl rand 16 -hex`	Generate 16 Byte random value, hex

OpenSSL (following must be preceded with openssl)

`enc -des-ecb -in p.txt -out c.bin`	Symmetric encrypt *p.txt* using DES-ECB
`enc -d -des-ecb -in c.bin -out r.txt`	Symmetric decrypt *c.bin* using DES-ECB
`dgst -md5 file`	MD5 hash of *file*
`genpkey -algorithm RSA -out priv.pem`	Generate RSA public/private key pair
`pkey -in priv.pem -out pub.pem -pubout`	Extract public key from key pair, save in *pub.pem*
`pkey -in priv.pem -text`	View public/private key values, text
`dgst -sha1 -sign priv.pem -out s.bin p.txt`	Sign *p.txt* using SHA and private key in *priv.pem*
`pkeyutl -encrypt -in p.txt -pubin -inkey pub.pem -out c.bin`	Encrypt *p.txt* using *pub.pem*, output *c.bin*
`pkeyutl -decrypt -in c.bin -inkey priv.pem -out r.txt`	Decrypt *c.bin* using *priv.pem*, output *r.txt*
`dgst -sha1 -verify pub.pem -signature s.bin r.txt`	Verify signature of r.txt