Foreword

This guide stems from the notes I have been taking while studying and working as a Linux sysadmin. It contains useful information about standards and tools for Linux system administration, as well as a good amount of topics from the certification exams LPIC-1 (Linux Professional Institute Certification level 1), LPIC-2, RHCSA (Red Hat Certified System Administrator), and RHCE (Red Hat Certified Engineer). Unless otherwise specified, the shell of reference is Bash.

This is an independent publication and is not affiliated with, authorized by, sponsored by, or otherwise approved by LPI or Red Hat. You can freely use and share the whole guide or the single pages, provided that you distribute them unmodified and not for profit.

This document was composed with Apache OpenOffice.

Happy Linux hacking,

Daniele Raffo

Version history

<table>
<thead>
<tr>
<th>Edition</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>May 2013</td>
</tr>
<tr>
<td>2nd</td>
<td>September 2014</td>
</tr>
<tr>
<td>3rd</td>
<td>July 2015</td>
</tr>
<tr>
<td>4th</td>
<td>June 2016</td>
</tr>
<tr>
<td>5th</td>
<td>September 2017</td>
</tr>
<tr>
<td>6th</td>
<td>August 2018</td>
</tr>
<tr>
<td>7th</td>
<td>May 2019</td>
</tr>
</tbody>
</table>

Bibliography and suggested readings

- Evi Nemeth et al., UNIX and Linux System Administration Handbook, O'Reilly
- Adam Haeder et al., LPI Linux Certification in a Nutshell, O'Reilly
- Heinrich W. Klöpping et al., The LPIC-2 Exam Prep, http://lpic2.unix.nl
- Ellen Siever et al., Linux in a Nutshell, O'Reilly, http://archive.oreilly.com/linux/cmd
- Christoph Braun, Unix System Security Essentials, Addison-Wesley
- A-Z index of Bash command line, https://ss64.com/bash
- Shell command line snippets, http://www.commandlinefu.com
- Bash command line snippets, http://www.bashoneliners.com
- Regular expressions tester, http://www.regextester.com
- Bash pitfalls, http://mywiki.wooledge.org/BashPitfalls
Logical Volume Management (LVM) introduces an abstraction between physical and logical storage, allowing a more versatile use of filesystems. LVM uses the Linux device mapper feature (/dev/mapper).

Disks, partitions, and RAID devices are made of **Physical Volumes**, which are grouped into a **Volume Group**. A Volume Group is divided into small fixed-size chunks called Physical Extents, which are mapped 1-to-1 to Logical Extents. Logical Extents are grouped into **Logical Volumes**, on which filesystems are created.

### How to create a Logical Volume

1. **Add a new physical or virtual disk to the machine**
2. `lsblk`
3. `fdisk /dev/sda`
4. `pvcreate /dev/sda1`
5. `vgcreate -s 8M myvg0 /dev/sda1`
   or `vgextend myvg0 /dev/sda1`
6. `lvcreate -L 1024M -n mylv myvg0`
7. `mkfs -t ext3 /dev/myvg0/mylv`
8. `mount /dev/myvg0/mylv /mnt/mystuff`

### How to increase the size of a Logical Volume (operation possible only if the underlying filesystem allows it)

1. **Add a new physical or virtual disk to the machine; this will provide the extra disk space**
2. `fdisk /dev/sdc`
3. `pvcreate /dev/sdc`
4. `vgextend myvg0 /dev/sdc`
5. `lvextend -L+2048M /dev/myvg0/mylv`
   or `lvresize -L+2048M /dev/myvg0/mylv`
   or `lvresize -l+100%FREE /dev/myvg0/mylv`
6. `resize2fs /dev/myvg0/mylv`

### How to reduce the size of a Logical Volume (operation possible only if the underlying filesystem allows it)

1. `resize2fs /dev/myvg0/mylv 900M`
2. `lvreduce -L 900M /dev/myvg0/mylv`
   or `lvresize -L 900M /dev/myvg0/mylv`

### How to snapshot and backup a Logical Volume

1. `lvcreate -s -L 1024M -n snapshot0 /dev/myvg0/mylv`
2. `tar cvzf snapshot0.tar.gz snapshot0`
3. `lvremove /dev/mvvg0/snapshot0`
<table>
<thead>
<tr>
<th>PV commands</th>
<th>VG commands</th>
<th>LV commands</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>pvs</code></td>
<td><code>vgs</code></td>
<td><code>lvs</code></td>
</tr>
<tr>
<td>Report information about Physical Volumes</td>
<td>Report information about Volume Groups</td>
<td>Report information about Logical Volumes</td>
</tr>
<tr>
<td><code>pvscan</code></td>
<td><code>vgscan</code></td>
<td><code>lvscan</code></td>
</tr>
<tr>
<td>Scan all disks for Physical Volumes</td>
<td>Scan all disks for Volume Groups</td>
<td>Scan all disks for Logical Volumes</td>
</tr>
<tr>
<td><code>pvdisplay</code></td>
<td><code>vgdisplay</code></td>
<td><code>lvdisplay</code></td>
</tr>
<tr>
<td>Display Physical Volume attributes</td>
<td>Display Volume Group attributes</td>
<td>Display Logical Volume attributes</td>
</tr>
<tr>
<td><code>pvck</code></td>
<td><code>vgck</code></td>
<td></td>
</tr>
<tr>
<td>Check Physical Volume metadata</td>
<td>Check Volume Group metadata</td>
<td></td>
</tr>
<tr>
<td><code>pvcreate</code></td>
<td><code>vgcreate</code></td>
<td><code>lvcreate</code></td>
</tr>
<tr>
<td>Initialize a disk or partition for use with LVM</td>
<td>Create a Volume Group using Physical Volumes</td>
<td>Create a Logical Volume in a Volume Group</td>
</tr>
<tr>
<td><code>pvchange</code></td>
<td><code>vgchange</code></td>
<td><code>lvchange</code></td>
</tr>
<tr>
<td>Change Physical Volume attributes</td>
<td>Change Volume Group attributes</td>
<td>Change Logical Volume attributes</td>
</tr>
<tr>
<td><code>pvremove</code></td>
<td><code>vgremove</code></td>
<td><code>lvremove</code></td>
</tr>
<tr>
<td>Remove a Physical Volume</td>
<td>Remove a Volume Group</td>
<td>Remove a Logical Volume</td>
</tr>
<tr>
<td><code>pvresize</code></td>
<td><code>vgresize</code></td>
<td><code>lvresize</code></td>
</tr>
<tr>
<td>Resize a disk or partition in use with LVM</td>
<td>Modify the size of a Logical Volume</td>
<td></td>
</tr>
</tbody>
</table>

LVM global commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>lvmdiskscan</code></td>
<td>Scan the system for disks and partitions usable by LVM</td>
</tr>
<tr>
<td><code>dmsetup command</code></td>
<td>Perform low-level LVM operations</td>
</tr>
</tbody>
</table>

Mapping of Logical Volumes in the filesystem

```
/dev/mapper/vgname-lvname
/dev/vgname/lvname
```
## System boot

<table>
<thead>
<tr>
<th>Boot sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>POST</strong> (Power-On Self Test)</td>
</tr>
<tr>
<td><strong>BIOS</strong> (Basic I/O System)</td>
</tr>
</tbody>
</table>

**Chain loader**

**GRUB (GRand Unified Bootloader)**

GRUB stage 1 is loaded from the MBR and executes GRUB stage 2 from filesystem. GRUB chooses which OS to boot on.

The chain loader hands over to the boot sector of the partition on which resides the OS.

The chain loader also mounts `initrd`, an initial ramdisk (typically a compressed ext2 filesystem) to be used as the initial root device during kernel boot; this make possible to load kernel modules that recognize hard drives hardware and that are hence needed to mount the real root filesystem. Afterwards, the system runs `/linuxrc` with PID 1.

(From Linux 2.6.13 onwards, the system instead loads into memory `initramfs`, a cpio-compressed image, and unpacks it into an instance of tmpfs in RAM. The kernel then executes `/init` from within the image.)

**Linux kernel**

Kernel decompression into memory.

Kernel execution.

Detection of devices.

The real root filesystem is mounted on `/` in place of the initial ramdisk.

**init**

Execution of `init`, the first process (PID 1).

The system tries to execute in the following order:

- `/sbin/init`
- `/etc/init`
- `/bin/init`
- `/bin/sh`

If none of these succeeds, the kernel panics.

**Startup**

The system loads startup scripts and runlevel scripts.

**Login**

If in text mode, `init` calls the `getty` process, which runs the `login` command that asks the user for login and password.

If in graphical mode, the X Display Manager starts the X Server.

Newer systems use UEFI (Unified Extensible Firmware Interface) instead of BIOS. UEFI does not use the MBR boot code; it has knowledge of partition table and filesystems, and stores its application files required for launch in a EFI System Partition, mostly formatted as FAT32.

After the POST, the system loads the UEFI firmware which initializes the hardware required for booting, then reads its Boot Manager data to determine which UEFI application to launch. The launched UEFI application may then launch another application, e.g. the kernel and `initramfs` in case of a boot loader like GRUB.
SysV startup sequence

<table>
<thead>
<tr>
<th>Startup sequence</th>
<th>Debian</th>
<th>Red Hat</th>
</tr>
</thead>
<tbody>
<tr>
<td>At startup /sbin/init executes all instructions on /etc/inittab. This script at first switches to the default runlevel...</td>
<td>id:2:initdefault:</td>
<td>id:5:initdefault:</td>
</tr>
<tr>
<td>... then it runs the following script (same for all runlevels) which configures peripheral hardware, applies kernel parameters, sets hostname, and provides disks initialization...</td>
<td>/etc/init.d/rcS</td>
<td>/etc/rc.d/rc.sysinit or /etc/rc.sysinit</td>
</tr>
<tr>
<td>... and then, for runlevel N, it calls the script /etc/init.d/rc N (i.e. with the runlevel number as parameter) which launches all services and daemons specified in the following startup directories:</td>
<td>/etc/rcN.d/</td>
<td>/etc/rc.d/rcN.d/</td>
</tr>
</tbody>
</table>

The startup directories contain symlinks to the init scripts in /etc/init.d/ which are executed in numerical order. Links starting with K are called with argument stop, links starting with S are called with argument start.

```
lrwxrwxrwx. 1 root root   14 Feb 11 22:32 K88sssd -> ../init.d/sssd
lrwxrwxrwx. 1 root root   15 Nov 28 14:50 K89rdisc -> ../init.d/rdisc
lrwxrwxrwx. 1 root root   17 Nov 28 15:01 S01sysstat -> ../init.d/sysstat
lrwxrwxrwx. 1 root root   18 Nov 28 14:54 S05cgconfig -> ../init.d/cgconfig
lrwxrwxrwx. 1 root root   16 Nov 28 14:52 S07iscsid -> ../init.d/iscsid
lrwxrwxrwx. 1 root root   18 Nov 28 14:42 S08iptables -> ../init.d/iptables
```

The last script to be run is S99local -> ../init.d/rc.local; therefore, an easy way to run a specific program upon boot is to call it from this script file.

```
/proc/total (SUSE)
/proc/before.local (SUSE)
/proc/after.local (SUSE)
```

To add or remove services at boot sequence:

```
update-rc.d service defaults
update-rc.d -f service remove
chkconfig --add service
chkconfig --del service
```

When adding or removing a service at boot, startup directories will be updated by creating or deleting symlinks for the default runlevels: K symlinks for runlevels 0 1 6, and S symlinks for runlevels 2 3 4 5.

Service will be run via the xinetd super server.

### Service operation parameters supported by the init scripts

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>Start the service</td>
</tr>
<tr>
<td>stop</td>
<td>Stop the service</td>
</tr>
<tr>
<td>restart</td>
<td>Restart the service (stop, then start)</td>
</tr>
<tr>
<td>status</td>
<td>Display daemon PID and execution status</td>
</tr>
<tr>
<td>force-reload</td>
<td>Reload configuration if service supports it, otherwise restart</td>
</tr>
<tr>
<td>condrestart</td>
<td>Restart the service only if already running</td>
</tr>
<tr>
<td>try-restart</td>
<td></td>
</tr>
<tr>
<td>reload</td>
<td>Reload the service configuration</td>
</tr>
</tbody>
</table>

#### Mandatory

#### Optional

---

**Linux Standard Base (LSB)**

The Linux Standard Base defines a format to specify default values on an init script /etc/init.d/foo:

```bash
### BEGIN INIT INFO
# Provides: foo
# Required-Start: bar
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Description: Service Foo init script
### END INIT INFO
```

Default runlevels and S/K symlinks values can also be specified as such:

```bash
# chkconfig: 2345 85 15
# description: Foo service
```
Login

/etc/init/start-ttys.conf (Red Hat)  Start the specified number of terminals at bootup via getty, which manages physical or virtual terminals (TTYS)

/etc/sysconfig/init (Red Hat)  Control appearance and functioning of the system during bootup

/etc/machine-id (Red Hat)  Randomly-generated machine ID

rm /etc/machine-id && 
    systemctl-machine-id-setup (Red Hat)  Regenerate the machine ID

/etc/securetty  List of TTYs from which the root user is allowed to login

/etc/issue  Message printed before the login prompt.  Can contain the following escape codes:

\b  Baudrate of line
\d  Date
\s  System name and OS
\l  Terminal device line
\m  Machine architecture identifier
\n  Nodename aka hostname
\o  Domain name
\r  OS release number
\t  Time
\u  Number of users logged in
\n  "n users" logged in
\v  OS version and build date

/etc/issue.net  Message printed before the login prompt on a remote session

/etc/motd  Message printed after a successful login, before execution of the login shell

/etc/nologin  If this file exists, login and sshd deny logging in to all unprivileged users. Useful when doing system maintenance

To prevent a specific user to log in, their shell can be set either as:

/bin/false  user is forced to exit immediately
/sbin/nologin  user is prompted a message and forced to exit; the message is ”This account is currently not available” or the contents of file /etc/nologin.txt if it exists

/var/log/auth.log  Logfile containing user logins and authentication mechanisms
/var/log/pwdfail  Logfile containing failed authentication attempts

who  Print the list of users logged into the system
w  Print the list of users logged into the system, and what they are doing

last  Print the list of users that logged in and out. Searches through the file /var/log/wtmp
lastb  Print the list of bad login attempts. Searches through the file /var/log/btmp

fail2ban  Temporarily ban IP addresses (via firewall rules) that have too many failed password logins. This information is taken from authentication logs
### Runlevels

<table>
<thead>
<tr>
<th>Runlevel (SysV)</th>
<th>Target (Systemd)</th>
<th>Debian</th>
<th>Red Hat</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>Shutdown</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>Single user / maintenance mode</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>multi-user.target</td>
<td>Multi-user mode (default)</td>
<td>Multi-user mode without network</td>
</tr>
<tr>
<td>3</td>
<td>multi-user.target</td>
<td>Multi-user mode</td>
<td>Multi-user mode with network</td>
</tr>
<tr>
<td>4</td>
<td>multi-user.target</td>
<td>Multi-user mode</td>
<td>Unused, for custom use</td>
</tr>
<tr>
<td>5</td>
<td>graphical.target</td>
<td>Multi-user mode</td>
<td>Multi-user mode with network and X (default)</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Reboot</td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
<td>Single user / maintenance mode (usually accessed through runlevel 1)</td>
<td></td>
</tr>
</tbody>
</table>

**Systemd's target** `runleveln.target` emulates a SysV's runlevel `n`.

- `runlevel`  
  - Display the previous and the current runlevel
- `who -r`  
  - Change to `runlevel`
- `init runlevel`  
  - Change to `runlevel`
- `telinit runlevel`  
  - Change to `runlevel`
- `systemctl get-default`  
  - Get the default target
- `systemctl set-default target`  
  - Set the default target
- `systemctl isolate target`  
  - Change to `target`
- `systemctl emergency`  
  - Change to maintenance single-user mode with only `/root` filesystem mounted
- `systemctl rescue`  
  - Change to maintenance single-user mode with only local filesystems mounted
- `init 0`  
  - Halt the system
- `telinit 0`  
  - Halt the system
- `shutdown -h now`  
  - Halt the system
- `halt`  
  - Halt the system
- `poweroff`  
  - Halt the system
- `init 6`  
  - Reboot the system
- `telinit 6`  
  - Reboot the system
- `shutdown -r now`  
  - Reboot the system
- `reboot`  
  - Reboot the system
- `shutdown`  
  - Shut down the system in a secure way: all logged-in users are notified via a message to their terminal, and login is disabled. Can only be run by the root user
- `shutdown -a`  
  - Non-root users that are listed in `/etc/shutdown.allow` can use this command to shut down the system
- `shutdown -h 16:00 message`  
  - Schedule a shutdown for 4 PM and send a warning message to all logged-in users
- `shutdown -f`  
  - Skip fsck on reboot
- `shutdown -F`  
  - Force fsck on reboot
- `shutdown -c`  
  - Cancel a shutdown that has been already initiated
<table>
<thead>
<tr>
<th>System V</th>
<th>Systemd</th>
<th>Action performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/init.d/service</td>
<td>systemctl operation service</td>
<td>Perform one of these operations on the specified service: start, stop, restart, status, force-reload, conda-restart, try-restart, reload.</td>
</tr>
<tr>
<td>service</td>
<td>(Red Hat)</td>
<td></td>
</tr>
<tr>
<td>rcservice</td>
<td>(SUSE)</td>
<td></td>
</tr>
<tr>
<td>update-rc.d service</td>
<td>defaults (Debian)</td>
<td>Add a service at boot</td>
</tr>
<tr>
<td>chkconfig --add</td>
<td>service (Red Hat)</td>
<td></td>
</tr>
<tr>
<td>update-rc.d -f</td>
<td>service remove (Debian)</td>
<td>Remove a service at boot</td>
</tr>
<tr>
<td>chkconfig --del</td>
<td>service (Red Hat)</td>
<td></td>
</tr>
<tr>
<td>update-rc.d -f</td>
<td>start 30 2 3 4 5, stop 70 0 1 6.</td>
<td>Add a service on the default runlevels; creates S30 symlinks for starting the service and K70 symlinks for stopping it</td>
</tr>
<tr>
<td>chkconfig --levels</td>
<td>2 4 5 service on</td>
<td>Add the service on runlevels 2 4 5</td>
</tr>
<tr>
<td>chkconfig service</td>
<td>on</td>
<td>Add the service on default runlevels</td>
</tr>
<tr>
<td>chkconfig service</td>
<td>off</td>
<td>Remove the service on default runlevels</td>
</tr>
<tr>
<td>chkconfig service</td>
<td>reset</td>
<td>Check if the service is enabled on the current runlevel</td>
</tr>
<tr>
<td>chkconfig service</td>
<td>resetpriorities</td>
<td>Reset the start/stop priorities of the service for all runlevels to whatever the LSB specifies in the init script</td>
</tr>
<tr>
<td>chkconfig --list</td>
<td>service</td>
<td>Display current configuration of service (its status and the runlevels in which it is active)</td>
</tr>
<tr>
<td>chkconfig --list</td>
<td>--type=service</td>
<td>List all active services and their current configuration</td>
</tr>
<tr>
<td>ls /etc/rcn.d</td>
<td>systemctl</td>
<td>List services started on runlevel n</td>
</tr>
<tr>
<td></td>
<td>---all</td>
<td>List loaded and active units</td>
</tr>
<tr>
<td></td>
<td>systemctl --t target</td>
<td>List targets</td>
</tr>
</tbody>
</table>
`/etc/inittab` describes which processes are started at bootup and during normal operation; it is read and executed by `init` at bootup. All its entries have the form `id:runlevels:action:process`.

<table>
<thead>
<tr>
<th>id</th>
<th>1-4 characters, uniquely identifies an entry. For gettys and other login processes it should be equal to the suffix of the corresponding tty runlevels</th>
</tr>
</thead>
<tbody>
<tr>
<td>runlevels</td>
<td>Runlevels for which the specified action must be performed. If empty, action is performed on all runlevels</td>
</tr>
<tr>
<td>action</td>
<td>Process to execute. If prepended by a <code>+</code>, <code>utmp</code> and <code>wtmp</code> accounting will not be performed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>respawn</td>
<td>Process will be restarted when it terminates. Process is started at the specified runlevel and <code>init</code> will wait for its termination (i.e. execution of further lines of <code>/etc/inittab</code> stops until the process exits)</td>
</tr>
<tr>
<td>wait</td>
<td>Process is executed once at the specified runlevel.</td>
</tr>
<tr>
<td>once</td>
<td>Process is executed at system boot. Runlevels field is ignored</td>
</tr>
<tr>
<td>boot</td>
<td>Process is executed at system boot and <code>init</code> will wait for its termination. Runlevels field is ignored</td>
</tr>
<tr>
<td>bootwait</td>
<td>Does nothing</td>
</tr>
<tr>
<td>off</td>
<td>Process is executed when an on-demand runlevel (A, B, C) is called</td>
</tr>
<tr>
<td>ondemand</td>
<td>Specifies the default runlevel to boot on. Process field is ignored</td>
</tr>
<tr>
<td>initdefault</td>
<td>Process is executed at system boot, before any <code>boot</code> or <code>bootwait</code> entries. Runlevels field is ignored</td>
</tr>
<tr>
<td>sysinit</td>
<td>Process is executed when power goes down and an UPS kicks in. <code>init</code> will not wait for its termination</td>
</tr>
<tr>
<td>powerfail</td>
<td>Process is executed when power goes down and an UPS kicks in. <code>init</code> will wait for its termination</td>
</tr>
<tr>
<td>powerwait</td>
<td>Process is executed when power is down and the UPS battery is almost empty</td>
</tr>
<tr>
<td>powerfailnow</td>
<td>Process is executed when <code>init</code> receives a <code>SIGINT</code> via <code>CTRL</code>, <code>ALT</code>, <code>DEL</code></td>
</tr>
<tr>
<td>powerokwait</td>
<td>Process is executed when power has been restored from UPS</td>
</tr>
<tr>
<td>ctrlaltdel</td>
<td>Process is executed when a special key combination is pressed on console</td>
</tr>
<tr>
<td>kbdrequest</td>
<td></td>
</tr>
</tbody>
</table>
## Filesystem Hierarchy Standard (FHS)

<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/bin</code></td>
<td>Essential command binaries</td>
</tr>
<tr>
<td><code>/boot</code></td>
<td>Bootloader files (e.g. OS loader, kernel image, initrd)</td>
</tr>
<tr>
<td><code>/dev</code></td>
<td>Virtual filesystem containing device nodes to devices and partitions</td>
</tr>
<tr>
<td><code>/etc</code></td>
<td>System configuration files and scripts</td>
</tr>
<tr>
<td><code>/home</code></td>
<td>Home directories for users</td>
</tr>
<tr>
<td><code>/lib</code></td>
<td>Libraries for the binaries in <code>/bin</code> and <code>/sbin</code>, kernel modules</td>
</tr>
<tr>
<td><code>/lost+found</code></td>
<td>Storage directory for recovered files in this partition</td>
</tr>
<tr>
<td><code>/media</code></td>
<td>Mount points for removable media</td>
</tr>
<tr>
<td><code>/mnt</code></td>
<td>Mount points for temporary filesystems</td>
</tr>
<tr>
<td><code>/net</code></td>
<td>Access to directory tree on different external NFS servers</td>
</tr>
<tr>
<td><code>/opt</code></td>
<td>Optional, large add-on application software packages</td>
</tr>
<tr>
<td><code>/proc</code></td>
<td>Virtual filesystem providing kernel and processes information</td>
</tr>
<tr>
<td><code>/root</code></td>
<td>Home directory for the root user</td>
</tr>
<tr>
<td><code>/sbin</code></td>
<td>Essential system binaries, system administration commands</td>
</tr>
<tr>
<td><code>/srv</code></td>
<td>Data for services provided by the system</td>
</tr>
<tr>
<td><code>/sys</code></td>
<td>Virtual filesystem providing information about hotplug hardware devices</td>
</tr>
<tr>
<td><code>/tmp</code></td>
<td>Temporary files; deleted at reboot</td>
</tr>
<tr>
<td><code>/usr</code></td>
<td>User utilities and applications</td>
</tr>
<tr>
<td><code>/usr/bin</code></td>
<td>Non-essential command binaries for all users</td>
</tr>
<tr>
<td><code>/usr/include</code></td>
<td>C header files</td>
</tr>
<tr>
<td><code>/usr/lib</code></td>
<td>Libraries for the binaries in <code>/usr/bin</code> and <code>/usr/sbin</code></td>
</tr>
<tr>
<td><code>/usr/local</code></td>
<td>Software installed locally</td>
</tr>
<tr>
<td><code>/usr/local/bin</code></td>
<td>Local software binaries</td>
</tr>
<tr>
<td><code>/usr/local/games</code></td>
<td>Local game binaries</td>
</tr>
<tr>
<td><code>/usr/local/include</code></td>
<td>Local C header files</td>
</tr>
<tr>
<td><code>/usr/local/lib</code></td>
<td>Local libraries for the binaries in <code>/usr/local/bin</code> and <code>/usr/local/sbin</code></td>
</tr>
<tr>
<td><code>/usr/local/man</code></td>
<td>Local man pages</td>
</tr>
<tr>
<td><code>/usr/local/sbin</code></td>
<td>Local system binaries</td>
</tr>
<tr>
<td><code>/usr/local/share</code></td>
<td>Local architecture-independent hierarchy</td>
</tr>
<tr>
<td><code>/usr/local/src</code></td>
<td>Local source code</td>
</tr>
<tr>
<td><code>/usr/sbin</code></td>
<td>Non-essential system binaries (daemons and services)</td>
</tr>
<tr>
<td><code>/usr/share</code></td>
<td>Architecture-independent files (e.g. icons, fonts, documentation)</td>
</tr>
<tr>
<td><code>/usr/share/doc</code></td>
<td>Package-specific documentation not included in man pages</td>
</tr>
<tr>
<td><code>/usr/share/man</code></td>
<td>Man pages</td>
</tr>
<tr>
<td><code>/usr/share/info</code></td>
<td>Documentation in Info format</td>
</tr>
<tr>
<td><code>/usr/src</code></td>
<td>Source code for the actual OS</td>
</tr>
<tr>
<td><code>/var</code></td>
<td>Variable files (e.g. logs, caches, mail spools)</td>
</tr>
<tr>
<td><code>/var/log</code></td>
<td>Logfiles</td>
</tr>
<tr>
<td><code>/var/opt</code></td>
<td>Variable files for the application software installed in <code>/opt</code></td>
</tr>
<tr>
<td><code>/var/spool</code></td>
<td>Queued items to be processed (e.g. mail messages, cron jobs, print jobs)</td>
</tr>
<tr>
<td><code>/var/tmp</code></td>
<td>Temporary files that need to be stored for a longer time; preserved between reboots</td>
</tr>
</tbody>
</table>

The manpage on `man hier` contains information about filesystem hierarchy.
Partitioning

/dev/hda
IDE hard drive
/dev/sda
SCSI, PATA, or SATA hard drive
/dev/vda
Virtual disk for KVM-based virtual machines

/dev/hda, /dev/hdb, /dev/hdc ...
First, second, third ... hard drive
/dev/sda1, /dev/sda2, /dev/sda3 ...
First, second, third ... partition of the first hard drive

The superblock contains information relative to the filesystem e.g. filesystem type, size, status, metadata structures. The Master Boot Record (MBR) is a 512-byte program located in the first sector of the hard disk; it contains information about hard disk partitions and has the duty of loading the OS. On recent systems, the MBR has been replaced by the GUID Partition Table (GPT). Most modern filesystems use journaling; in a journaling filesystem, the journal logs changes before committing them to the filesystem, which ensures faster recovery and less corruption in case of a crash.

Partitioning limits for Linux using MBR:
Max 4 primary partitions per hard disk, or 3 primary partitions + 1 extended partition
Max 11 logical partitions (inside the extended partition) per hard disk
Max disk size is 2 Tb.

GPT makes no difference between primary, extended, or logical partitions; also, it has practically no limits concerning number and size of partitions.

fdisk /dev/sda
Disk partitioning interactive tool

fdisk -l /dev/sda
List the partition table of /dev/sda

parted
Disk partitioning interactive tool

sfdisk /dev/sda
Disk partitioning non-interactive tool

cfdisk
Disk partitioning tool with text-based UI

gparted
gnome-disks
Disk partitioning tool with GUI

partprobe
This command can be run after fdisk operations to notify the OS of partition table changes. Otherwise, the changes will take place only after reboot

mkfs -t fstype device
Create a filesystem of the specified type on a partition (i.e. format the partition). mkfs is a wrapper utility for the actual filesystem-specific maker commands:
mkfs.ext2  aka  mke2fs
mkfs.ext3  aka  mke3fs
mkfs.ext4
mkfs.msdos  aka  mkdosfs
mkfs.ntfs  aka  mkntfs
mkfs.reiserfs  aka  mkreiserfs
mkfs.jfs
mkfs.xfs

mkfs -t ext2  /dev/sda
Create an ext2 filesystem on /dev/sda
mke2fs  /dev/sda

mke2fs -j  /dev/sda
Create an ext3 filesystem (ext2 with journaling) on /dev/sda
mkfs.ext3  /dev/sda
mke3fs  /dev/sda

mkfs -t msdos  /dev/sda
Create a MS-DOS filesystem on /dev/sda
mkfs.msdos  /dev/sda
mkdosfs  /dev/sda
mount

cat /proc/mounts
cat /etc/mtab

Display the currently mounted filesystems. The commands `mount` and `umount` maintain in `/etc/mtab` a database of currently mounted filesystems, but `/proc/mounts` is authoritative

mount -a

Mount all devices listed in `/etc/fstab`, except those indicated as `noauto`

mount -t ext3 /dev/sda /mnt

Mount a Linux-formatted disk. The mount point (directory) must exist

mount -t msdos /dev/fd0 /mnt

Mount a MS-DOS filesystem floppy disk to mount point `/mnt`

mount /dev/fd0

Mount a floppy disk. `/etc/fstab` must contain an entry for `/dev/fd0`

mount -o remount,rw /

Remount the root directory as read-write, supposing it was mounted read-only. Useful to change flags (in this case, read-only to read-write) for a mounted filesystem that cannot be unmounted at the moment

mount -o nolock 10.7.7.7:/export/ /mnt/nfs

Mount a NFS share without running NFS daemons. Useful during system recovery

mount -t iso9660 -o ro,loop=/dev/loop0 cd.img /mnt/cdrom

Mount a CD-ROM ISO9660 image file like a CD-ROM (via the loop device)

umount /dev/fd0

umount /mnt

Unmount a floppy disk that was mounted on `/mnt` (device must not be busy)

umount -l /dev/fd0

Unmount the floppy disk as soon as it is not in use anymore

eject /dev/fd0
eject /mnt

Eject a removable media device

mountpoint /mnt

Tell if a directory is a mount point

The UUID (Universal Unique Identifier) of a partition is a 128-bit hash number, which is associated to the partition when the partition is initialized.

blkid /dev/sda1

Print the UUID of the specified partition

blkid -L /boot

Print the UUID of the specified partition, given its label

blkid -U 652b786e-b87f-49d2-af23-8087ced0c667

Print the name of the specified partition, given its UUID

findfs UUID=652b786e-b87f-49d2-af23-8087ced0c667

Print the name of the specified partition, given its UUID

findfs LABEL=/boot

Print the name of the specified partition, given its label

e2label /dev/sda1

Print the label of the specified partition
### Filesystem types

<table>
<thead>
<tr>
<th>Partition types</th>
<th>Filesystem types</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00</td>
<td>Empty</td>
</tr>
<tr>
<td>0x01</td>
<td>FAT12</td>
</tr>
<tr>
<td>0x02</td>
<td>XENIX root</td>
</tr>
<tr>
<td>0x03</td>
<td>XENIX usr</td>
</tr>
<tr>
<td>0x04</td>
<td>FAT16 &lt;32M</td>
</tr>
<tr>
<td>0x05</td>
<td>Extended</td>
</tr>
<tr>
<td>0x06</td>
<td>FAT16</td>
</tr>
<tr>
<td>0x07</td>
<td>HPFS / NTFS / exFAT</td>
</tr>
<tr>
<td>0x08</td>
<td>AIX</td>
</tr>
<tr>
<td>0x09</td>
<td>AIX bootable</td>
</tr>
<tr>
<td>0x0a</td>
<td>OS/2 Boot Manager</td>
</tr>
<tr>
<td>0x0b</td>
<td>W95 FAT32</td>
</tr>
<tr>
<td>0x0c</td>
<td>W95 FAT32 (LBA)</td>
</tr>
<tr>
<td>0x0e</td>
<td>W95 FAT16 (LBA)</td>
</tr>
<tr>
<td>0x0f</td>
<td>W95 extended (LBA)</td>
</tr>
<tr>
<td>0x10</td>
<td>OPUS</td>
</tr>
<tr>
<td>0x11</td>
<td>Hidden FAT12</td>
</tr>
<tr>
<td>0x12</td>
<td>Compaq diagnostics</td>
</tr>
<tr>
<td>0x14</td>
<td>Hidden FAT16 &lt;32Mb</td>
</tr>
<tr>
<td>0x16</td>
<td>Hidden FAT16</td>
</tr>
<tr>
<td>0x17</td>
<td>Hidden HPFS/NTFS</td>
</tr>
<tr>
<td>0x18</td>
<td>AST SmartSleep</td>
</tr>
<tr>
<td>0x1b</td>
<td>Hidden W95 FAT32</td>
</tr>
<tr>
<td>0x1c</td>
<td>Hidden W95 FAT32 (LBA)</td>
</tr>
<tr>
<td>0x1e</td>
<td>Hidden W95 FAT16 (LBA)</td>
</tr>
<tr>
<td>0x24</td>
<td>NEC DOS</td>
</tr>
<tr>
<td>0x27</td>
<td>Hidden NTFS WinRE</td>
</tr>
<tr>
<td>0x29</td>
<td>Plan 9</td>
</tr>
<tr>
<td>0x3c</td>
<td>PartitionMagic recovery</td>
</tr>
<tr>
<td>0x40</td>
<td>Venix 80286</td>
</tr>
<tr>
<td>0x41</td>
<td>PPC PreP Boot</td>
</tr>
<tr>
<td>0x42</td>
<td>SFS</td>
</tr>
<tr>
<td>0x44</td>
<td>QNX4.x</td>
</tr>
</tbody>
</table>

The command `sfdisk -T` prints the above list of partition IDs and names.

### Most used Linux-supported filesystems

- **ext2**: Linux default filesystem, offering the best performances
- **ext3**: ext2 with journaling
- **ext4**: Linux journaling filesystem, an upgrade from ext3
- **Reiserfs**: Journaling filesystem
- **XFS**: Journaling filesystem, developed by SGI
- **JFS**: Journaling filesystem, developed by IBM
- **Btrfs**: B-tree filesystem, developed by Oracle
- **msdos**: DOS filesystem, supporting only 8-char filenames
- **umsdos**: Extended DOS filesystem used by Linux, compatible with DOS
- **fat32**: MS-Windows FAT filesystem
- **vfat**: Extended DOS filesystem, with support for long filenames
- **ntfs**: Replacement for fat32 and vfat filesystems
- **minix**: Native filesystem of the MINIX OS
- **iso9660**: CD-ROM filesystem
- **cramfs**: Compressed RAM disk
- **nfs**: Network filesystem, used to access files on remote machines
- **SMB**: Server Message Block, used to mount Windows network shares
- **proc**: Pseudo filesystem, used as an interface to kernel data structures
- **swap**: Pseudo filesystem, Linux swap area
In Linux, the swap space is a virtual memory area (a file or a partition) used as RAM extension. Usually a partition is preferred because of better performances concerning fragmentation and disk speed. Although listed as filesystem type 0x82, the swap partition is not a filesystem but a raw addressable memory with no structure; therefore it is not shown in the output of `mount` or `df` commands. The `fdisk` tool can be used to create a swap partition.

```
dd if=/dev/zero of=/swapfile bs=1024 count=512000
mkswap /swapfile
swapon /swapfile
swappoff /swapfile
swapon -s
```

Create a 512-Mb swap file

Initialize a (already created) swap file or partition

Enable a swap file or partition, thus telling the kernel that it can use it now

Disable a swap file or partition

Show the sizes of total and used swap areas

---

**How to extend a LVM swap partition**

1. `lvs`
   Determine the name of the swap Logical Volume

2. `swapoff /dev/volgroup0/swap_lv`
   Turn off the swap volume

3. `lvresize -L+1G /dev/volgroup0/swap_lv`
   Extend the swap volume with an additional 1 Gb of space

4. `mkswap /dev/volgroup0/swap_lv`
   Format the swap volume

5. `swapon /dev/volgroup0/swap_lv`
   Turn on the swap volume
### /etc/fstab

<table>
<thead>
<tr>
<th>filesystem</th>
<th>mount point</th>
<th>type</th>
<th>options</th>
<th>dump</th>
<th>pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>/dev/sda2</td>
<td>/</td>
<td>ext2</td>
<td>defaults</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>/dev/sdb1</td>
<td>/home</td>
<td>ext2</td>
<td>defaults</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>/dev/cdrom</td>
<td>/media/cdrom</td>
<td>auto</td>
<td>ro, noauto, user, exec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/fd0</td>
<td>/media/floppy</td>
<td>auto</td>
<td>rw, noauto, user, sync</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>proc</td>
<td>/proc</td>
<td>swap</td>
<td>swap</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>/dev/hda1</td>
<td>swap</td>
<td>swap</td>
<td>pri=42</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>nfsserver:/dirs</td>
<td>/mnt</td>
<td>nfs</td>
<td>intr</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>//smbserver/jdoe</td>
<td>/shares/jdoe</td>
<td>cifs</td>
<td>auto, credentials=/etc/smbcreds</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>LABEL=/boot</td>
<td>/boot</td>
<td>ext4</td>
<td>errors=remount-ro, noatime</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UUID=652b786e-b87f-49d2-af23-8087ced0c667</td>
<td>/test</td>
<td>ext4</td>
<td>errors=remount-ro, noatime</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

/`etc/fstab` contains information about filesystems, including all filesystems that must be automatically mounted at bootup.

<table>
<thead>
<tr>
<th>filesystem</th>
<th>Device or partition. The filesystem can be identified either by its name, label, or UUID</th>
</tr>
</thead>
<tbody>
<tr>
<td>mount point</td>
<td>Directory on which the partition will be mounted</td>
</tr>
<tr>
<td>type</td>
<td>Filesystem type, or auto if detected automatically</td>
</tr>
</tbody>
</table>
| options      | defaults Use the default options: rw, suid, dev, auto, nouser, exec, async
|              | ro Mount read-only                                                                       |
|              | rw Mount read-write (default)                                                            |
|              | suid Permit SUID and SGID bit operations (default)                                       |
|              | nosuid Do not permit SUID and SGID bit operations                                        |
|              | dev Interpret block special devices on the filesystem (default)                          |
|              | nodel Do not interpret block special devices on the filesystem                          |
|              | auto Mount automatically at bootup, or when command mount -a is given (default)         |
|              | noauto Mount only if explicitly demanded                                                 |
|              | user Partition can be mounted by any user                                                |
|              | nouser Partition can be mounted only by the root user (default)                          |
|              | exec Binaries contained on the partition can be executed (default)                       |
|              | noexec Binaries contained on the partition cannot be executed                           |
|              | sync Write files immediately to the partition                                            |
|              | async Buffer write operations and commit them at once later, or when device is unmounted (default) |
|              | noatime Do not update atime (access time) information for the filesystem. This results in a performance improvement because the system does not need anymore to do filesystem writes for files which are just being read |
|              | acl Support ACLs on files contained in the partition                                      |
|              | context="context" Apply a specific SELinux context to the mount                         |

Other specific options apply to specific partition types (e.g. NFS or Samba)

<table>
<thead>
<tr>
<th>dump</th>
<th>Options for the <code>dump</code> backup utility. 0 = do not backup</th>
</tr>
</thead>
<tbody>
<tr>
<td>pass</td>
<td>Order in which the filesystem must be checked by <code>fsck</code>. 0 = do not check</td>
</tr>
</tbody>
</table>
### Filesystem operations

**df**
- Report filesystem disk space usage
- df 
  - df -h: Report filesystem disk space usage in human-readable output
- df directory: Shows on which device the specified directory is mounted

**du**
- du directory: Report disk usage as size of each file inside directory
- du -s directory: Report the sum of all files contained inside directory
- du -h directory: Report disk usage in human-readable output

**ndu**
- Disk usage analyzer with ncurses UI

**find**
- find /path -type f -exec du -Sh {} +
  | sort -nr | head: Print out the 10 biggest directories under path

**resize2fs**
- resize2fs options device size: Resize an ext2/ext3/ext4 filesystem

**lsblk**
- List information about all available block devices

**lsscsi**
- List information about all SCSI devices

**blockdev --getbsz /dev/sda1**
- Get the block size of the specified partition

**sync**
- Flush the buffer and commit all pending writes.
- To improve performance of Linux filesystems, many write operations are buffered in RAM and written at once; writes are done in any case before unmount, reboot, or shutdown

**chroot /mnt/sysimage**
- Start a shell with /mnt/sysimage as filesystem root.
- Useful during system recovery when the machine has been booted from a removable media (which hence is defined as the filesystem root)

**mknod /dev/sda**
- Create a directory allocating the proper inode.
- Useful during system recovery when experiencing filesystem problems

**hdparm**
- Get/set drive parameters for SATA/IDE devices
- HDparm -g /dev/hda: Display drive geometry (cylinders, heads, sectors) of /dev/hda
- HDparm -i /dev/hda: Display identification information for /dev/hda
- HDparm -tT /dev/hda: Perform disk read benchmarks on the /dev/hda drive
  - Using an unsupported mode can cause filesystem corruption!

**sdparm**
- Access drive parameters for SCSI devices
Filesystem maintenance

fsck device
Check and repair a Linux filesystem (which must be unmounted).
Corrupted files will be placed into the /lost+found directory of the partition.
The exit code returned is the sum of the following conditions:

- 0  No errors
- 1  File system errors corrected
- 2  System should be rebooted
- 4  File system errors left uncorrected
- 8  Operational error
- 16 Usage or syntax error
- 32 Fsck canceled by user
- 128 Shared library error

Fsck is a wrapper utility for the actual filesystem-specific checker commands:

- fsck.ext2 aka e2fsck
- fsck.ext3 aka e2fsck
- fsck.ext4 aka e2fsck
- fsck.msdos
- fsck.vfat
- fsck.cramfs

fsck
Check and repair serially all filesystems listed in /etc/fstab

fsck -f /dev/sda1
Force a filesystem check on /dev/sda1 even if it thinks is not necessary

fsck -y /dev/sda1
During filesystem repair, do not ask questions and assume that the answer is always yes

fsck.ext2 -c /dev/sda1
e2fsck -c /dev/sda1
Check an ext2 filesystem, running the badblocks command to mark all bad blocks and add them to the bad block inode so they will not be allocated to files or directories

touch /forcefsck (Red Hat)
Force a filesystem check after next reboot

tune2fs options device
Adjust tunable filesystem parameters on ext2/ext3/ext4 filesystems

tune2fs -l /dev/sda1
List the contents of the filesystem superblock

tune2fs -j /dev/sda1
Add a journal to this ext2 filesystem, making it an ext3

tune2fs -m 1 /dev/sda1
Reserve 1% of the partition size to privileged processes. This space (5% by default, but can be reduced on modern filesystems) is reserved to avoid filesystem fragmentation and to allow privileged processes to continue to run correctly when the partition is full


tune2fs -C 7 /dev/sda1
Set the mount count of the filesystem to 7

tune2fs -c 20 /dev/sda1
Set the filesystem to be checked by fsck after 20 mounts

tune2fs -i 15d /dev/sda1
Set the filesystem to be checked by fsck each 15 days

Both mount-count-dependent and time-dependent checking are enabled by default for all hard drives on Linux, to avoid the risk of filesystem corruption going unnoticed.

dumpe2fs options device
Dump ext2/ext3/ext4 filesystem information

dumpe2fs -h /dev/sda1
Display filesystem’s superblock information (e.g. number of mounts, last checks, UUID)

dumpe2fs /dev/sda1 | grep -i superblock
Display locations of superblock (primary and backup) of filesystem

dumpe2fs -b /dev/sda1
Display blocks that are marked as bad in the filesystem

debugfs device
Interactive ext2/ext3/ext4 filesystem debugger

debugfs -w /dev/sda1
Debug /dev/sda1 in read-write mode
(by default, debugfs accesses the device in read-only mode)

Many hard drives feature the Self-Monitoring, Analysis and Reporting Technology (SMART) whose purpose is to monitor the reliability of the drive, predict drive failures, and carry out different types of drive self-tests.
The smartd daemon attempts to poll this information from all drives every 30 minutes, logging all data to syslog.

smartctl -a /dev/sda
Print SMART information for drive /dev/sda

smartctl -s off /dev/sda
Disable SMART monitoring and log collection for drive /dev/sda

smartctl -t long /dev/sda
Begin an extended SMART self-test on drive /dev/sda
### XFS, ReiserFS, CD-ROM fs

- **xfs_growfs** options mountpoint
  Expand an XFS filesystem. For this, there must be at least one spare new disk partition available. Note that a XFS filesystem cannot be shrunk.

- **xfs_info** /dev/sdal
  **xfs_growfs -n** /dev/sdal
  Print XFS filesystem geometry.

- **xfs_check** options device
  **xfs_repair** options device
  Check XFS filesystem consistency. Repair a damaged or corrupt XFS filesystem.

- **xfsdump** -v silent -f /dev/tape /
  Dump the root of a XFS filesystem to tape, with the lowest verbosity. Incremental and resumed dumps are stored in the inventory database /var/lib/xfsdump/inventory.

- **xfsrestore** -f /dev/tape /
  Restore a XFS filesystem from tape.

- **xfsdump** -J - / | **xfsrestore** -J - /new
  Copy the contents of a XFS filesystem to another directory, without updating the inventory database.

- **reiserfstune** options device
  Adjust tunable filesystem parameters on ReiserFS filesystem.

- **debugreiserfs** device
  Interactive ReiserFS filesystem debugger.

- **mkisofs** -r -o crom.img data/
  Create a CD-ROM image from the contents of the target directory. Enable Rock Ridge extension and set all content on CD to be public readable, instead of inheriting the permissions from the original files.

### CD-ROM filesystems

<table>
<thead>
<tr>
<th>Filesystem</th>
<th>Commands</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO9660</td>
<td>mkisofs</td>
<td>Create a ISO9660 filesystem</td>
</tr>
<tr>
<td>UDF (Universal Disk Format)</td>
<td>mkudffs</td>
<td>Create a UDF filesystem</td>
</tr>
<tr>
<td></td>
<td>uddfack</td>
<td>Check a UDF filesystem</td>
</tr>
<tr>
<td></td>
<td>wrufd</td>
<td>Maintain a UDF filesystem</td>
</tr>
<tr>
<td></td>
<td>cdrwtool</td>
<td>Manage CD-RW drives (e.g. disk format, read/write speed)</td>
</tr>
<tr>
<td>HFS (Hierarchical File System)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### CD-ROM filesystem extensions

<table>
<thead>
<tr>
<th>Filesystem</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rock Ridge</td>
<td>Contains the original file information (e.g. permissions, filename) for MS Windows 8.3 filenames</td>
</tr>
<tr>
<td>MS Joliet</td>
<td>Used to create more MS Windows friendly CD-ROMs</td>
</tr>
<tr>
<td>El Torito</td>
<td>Used to create bootable CD-ROMs</td>
</tr>
</tbody>
</table>
AutoFS is a client-side service that allows automounting of filesystems, even for nonprivileged users. AutoFS is composed of the autofs kernel module that monitors specific directories for attempts to access them; in this case, the kernel module signals the automount userspace daemon, which mounts the directory when it needs to be accessed and unmounts it when is no longer accessed. Mounts managed by AutoFS should not be mounted/unmounted manually or via /etc/fstab, to avoid inconsistencies.

### AutoFS configuration files

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/sysconfig/autofs</td>
<td>AutoFS configuration file.</td>
</tr>
<tr>
<td>/etc/auto.master</td>
<td>Master map file for AutoFS. Each line is an indirect map, and each map file stores the configuration for the automounting of the subdirectory. The -hosts map tells AutoFS to mount/unmount automatically any export from the NFS server nfsserver when the directory /net/nfsserver/ is accessed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th># mount point</th>
<th>map</th>
<th>options</th>
</tr>
</thead>
<tbody>
<tr>
<td>/net</td>
<td>-hosts</td>
<td></td>
</tr>
<tr>
<td>/-</td>
<td>/etc/auto.direct</td>
<td></td>
</tr>
<tr>
<td>/misc</td>
<td>/etc/auto.misc</td>
<td>--timeout=60</td>
</tr>
<tr>
<td>/home</td>
<td>/etc/auto.home</td>
<td></td>
</tr>
</tbody>
</table>

### AutoFS map files

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/auto.direct</td>
<td>Direct map file for automounting of a NFS share.</td>
</tr>
<tr>
<td># dir</td>
<td>filesystem</td>
</tr>
<tr>
<td>/mydir</td>
<td>nfsserver1.foo.org:/myshare</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th># subdir</th>
<th>options</th>
<th>filesystem</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td>-ro,soft,intr</td>
<td>ftp.example.org:/pub</td>
</tr>
<tr>
<td>cd</td>
<td>-fstype=iso9660,ro,nosuid,nodev</td>
<td>:/dev/cdrom</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/auto.misc</td>
<td>Indirect map file for automounting of directory /misc.</td>
</tr>
<tr>
<td># subdir options filesystem</td>
<td></td>
</tr>
<tr>
<td>public</td>
<td>-ro,soft,intr</td>
</tr>
<tr>
<td>cd</td>
<td>-fstype=iso9660,ro,nosuid,nodev</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/auto.home</td>
<td>Indirect map file for automounting of directory /home on a NFS share. The * wildcard matches any subdirectory the system attempts to access, and the &amp; variable takes the value of the match.</td>
</tr>
<tr>
<td># subdir options filesystem</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>-rw,soft,intr</td>
</tr>
</tbody>
</table>
**RAID levels**

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
<th>Storage capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAID 0</td>
<td>Striping (data is written across all member disks). High I/O but no redundancy</td>
<td>Sum of the capacity of member disks</td>
</tr>
<tr>
<td>RAID 1</td>
<td>Mirroring (data is mirrored on all disks). High redundancy but high cost</td>
<td>Capacity of the smaller member disk</td>
</tr>
<tr>
<td>RAID 4</td>
<td>Parity on a single disk. I/O bottleneck unless coupled to write-back caching</td>
<td>Sum of the capacity of member disks, minus one</td>
</tr>
<tr>
<td>RAID 5</td>
<td>Parity distributed across all disks. Can sustain one disk crash</td>
<td>Sum of the capacity of member disks, minus one</td>
</tr>
<tr>
<td>RAID 6</td>
<td>Double parity distributed across all disks. Can sustain two disk crashes</td>
<td>Sum of the capacity of member disks, minus two</td>
</tr>
<tr>
<td>RAID 10 (1+0)</td>
<td>Striping + mirroring. High redundancy but high cost</td>
<td>Capacity of the smaller member disk</td>
</tr>
<tr>
<td>Linear RAID</td>
<td>Data written sequentially across all disks. No redundancy</td>
<td>Sum of the capacity of member disks</td>
</tr>
</tbody>
</table>

```
mdadm -C /dev/md0 -l 5 -n 3 /dev/sdb1 /dev/sdc1 /dev/sdd1 -x 1 /dev/sde1
```

Create a RAID 5 array from three partitions and a spare.
Partitions type must be set to 0xFD.
Once the RAID device has been created, it must be formatted e.g. via

```
mke2fs -j /dev/md0
```

```
mdadm --manage /dev/md0 -f /dev/sdd1
mdadm --manage /dev/md0 -r /dev/sdd1
mdadm --manage /dev/md0 -a /dev/sdd1
```

Mark a drive as faulty, before removing it
Remove a drive from the RAID array.
The faulty drive can now be physically removed
Add a drive to the RAID array.
To be run after the faulty drive has been physically replaced

```
mdadm --misc -Q /dev/sdd1
mdadm --misc -D /dev/md0
mdadm --misc -o /dev/md0
mdadm --misc -w /dev/md0
```

Display information about a device
Display detailed information about the RAID array
Mark the RAID array as readonly
Mark the RAID array as read & write

```
/etc/mdadm.conf
```

Configuration file for the `mdadm` command

```
DEVICE /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1
ARRAY /dev/md0 level=raid5 num-devices=3
UUID=0098af43:812203fa:e665b421:002f5e42
devices=/dev/sdb1,/dev/sdc1,/dev/sdd1,/dev/sde1
```

```
cat /proc/mdstat
```

Display information about RAID arrays and devices
## Non-GRUB bootloaders

<table>
<thead>
<tr>
<th>Bootloader</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LILO</strong> (Linux Loader)</td>
<td>Obsolete. Small bootloader that can be placed in the MBR or the boot sector of a partition. The configuration file is <code>/etc/lilo.conf</code> (run <code>/sbin/lilo</code> afterwards to validate changes).</td>
</tr>
<tr>
<td><strong>SYSLINUX</strong></td>
<td>Able to boot from FAT and NTFS filesystems e.g. floppy disks and USB drives. Used for boot floppy disks, rescue floppy disks, and Live USBs.</td>
</tr>
<tr>
<td><strong>ISOLINUX</strong></td>
<td>Able to boot from CD-ROM ISO 9660 filesystems. Used for Live CDs and bootable install CDs. The CD must contain the following files:</td>
</tr>
<tr>
<td></td>
<td><em>isolinux/isolinux.bin</em> — ISOLINUX image, from the SYSLINUX distro</td>
</tr>
<tr>
<td></td>
<td><em>boot/isolinux/isolinux.cfg</em> — ISOLINUX configuration</td>
</tr>
<tr>
<td></td>
<td><em>images/</em> — Floppy images to boot</td>
</tr>
<tr>
<td></td>
<td><em>kernel/memdisk</em></td>
</tr>
<tr>
<td></td>
<td>and can be burnt with the command:</td>
</tr>
<tr>
<td></td>
<td><code>mkisofs -o output.iso -b isolinux/isolinux.bin -c isolinux/boot.cat</code> <code>mkisofs -b /boot/isolinux/isolinux.cfg</code> <code>cd_root_dir</code></td>
</tr>
<tr>
<td><strong>PXELINUX</strong></td>
<td>Able to boot from PXE (Pre-boot eXecution Environment). PXE uses DHCP or BOOTP to enable basic networking, then uses TFTP to download a bootstrap program that loads and configures the kernel. Used for Linux installations from a central server or network boot of diskless workstations. The boot TFTP server must contain the following files:</td>
</tr>
<tr>
<td></td>
<td><em>tftpboot/pxelinux.0</em> — PXELINUX image, from the SYSLINUX distribution</td>
</tr>
<tr>
<td></td>
<td><em>tftpboot/pxelinux.cfg/</em> — Directory containing a configuration file for each machine. A machine with Ethernet MAC address <code>88:99:aa:bb:cc:dd</code> and IP address <code>192.0.2.91</code> (C000025B in hexadecimal) will search for its configuration filename in this order:</td>
</tr>
<tr>
<td></td>
<td><code>01-88-99-aa-bb-cc-dd</code></td>
</tr>
<tr>
<td></td>
<td><code>C000025B</code></td>
</tr>
<tr>
<td></td>
<td><code>C000025</code></td>
</tr>
<tr>
<td></td>
<td><code>C00002</code></td>
</tr>
<tr>
<td></td>
<td><code>C0000</code></td>
</tr>
<tr>
<td></td>
<td><code>C000</code></td>
</tr>
<tr>
<td></td>
<td><code>C0</code></td>
</tr>
<tr>
<td></td>
<td><code>C</code></td>
</tr>
<tr>
<td></td>
<td><code>default</code></td>
</tr>
<tr>
<td><strong>EXTLINUX</strong></td>
<td>General-purpose bootloader like LILO or GRUB. Now merged with SYSLINUX.</td>
</tr>
</tbody>
</table>
GRUB (Grand Unified Bootloader) is the standard boot manager on Linux distributions. The latest version is GRUB 2; the older version is GRUB Legacy. GRUB Stage 1 (446 bytes), as well as the partition table (64 bytes) and the boot signature (2 bytes), is stored in the 512-byte MBR. It then accesses the GRUB configuration and commands available on the filesystem, usually on `/boot/grub`.  

```bash
# Linux Red Hat
menuentry "Fedora 2.6.32" {  
set root=(hd0,1)  
linux /vmlinuz-2.6.32 ro root=/dev/hda5 mem=2048M  
initrd /initrd-2.6.32  
}

# Linux Debian
menuentry "Debian 2.6.36-experimental" {  
set root=(hd0,1)  
linux (hd0,1)/bzImage-2.6.36-experimental ro root=/dev/hda6  
}

# Windows
menuentry "Windows" {  
set root=(hd0,2)  
chainloader +1  
}
```

The GRUB 2 configuration file must not be edited manually. Instead, one must edit the files in `/etc/grub.d/` (these are scripts that will be run in order) and the file `/etc/default/grub` (the configuration file for menu display settings), then run `update-grub` (Debian) or `grub2-mkconfig` (Red Hat) which will recreate this configuration file.

### Common kernel parameters:

- **root=**: Specify the location of the filesystem root. This is a required parameter.
- **ro**: Mount read-only on boot.
- **quiet**: Disable non-critical kernel messages during boot.
- **debug**: Enable kernel debugging.
- **splash**: Show splash image.
- **single**: Boot in single-user mode (runlevel 1).
- **emergency**: Emergency mode: after the kernel is booted, run `sulogin` (single-user login) which asks for the root password for system maintenance, then run a Bash shell. Does not load `init` or any daemon or configuration setting.
- **init=/bin/bash**: Run a Bash shell (may also be any other executable) instead of `init`.  

The GRUB menu, presented at startup, allows choosing the OS or kernel to boot:

- **ENTER**: Boot the currently selected GRUB entry
- **C**: Get a GRUB command line
- **E**: Edit the selected GRUB entry (e.g. to edit kernel parameters in order to boot in single-user emergency mode, or to change IRQ or I/O port of a device driver compiled in the kernel)
- **B**: Boot the currently selected GRUB entry. This is usually done after finishing modifying the entry
- **P**: Bring up the GRUB password prompt. Necessary if a GRUB password has been set

**GRUB 2 commands**

- `grub2-mkconfig -o /boot/grub2/grub.cfg` (BIOS) Regenerate GRUB configuration file
- `grub2-mkconfig -o /boot/efi/EFI/centos/grub.cfg` (EFI)
- `grub-install /dev/sda` Install GRUB on first SATA drive
- `grub` Access the GRUB shell
- `grub2-set-default 1` Set GRUB to automatically boot the second entry in the GRUB menu
- `grub2-editenv list` Display the current GRUB menu entry that is automatically booted

**/boot/grub/device.map**

This file can be created to map Linux device filenames to BIOS drives:

- `(fd0) /dev/fd0`
- `(hd0) /dev/hda`
### GRUB Legacy shell commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>blocklist file</code></td>
<td>Print the block list notation of a file</td>
</tr>
<tr>
<td><code>boot</code></td>
<td>Boot the loaded OS</td>
</tr>
<tr>
<td><code>cat file</code></td>
<td>Show the contents of a file</td>
</tr>
<tr>
<td><code>chainloader file</code></td>
<td>Chainload another bootloader</td>
</tr>
<tr>
<td><code>cmp file1 file2</code></td>
<td>Compare two files</td>
</tr>
<tr>
<td><code>configfile file</code></td>
<td>Load a configuration file</td>
</tr>
<tr>
<td><code>debug</code></td>
<td>Toggle debugging mode</td>
</tr>
<tr>
<td><code>displayapm</code></td>
<td>Display APM BIOS information</td>
</tr>
<tr>
<td><code>displaymem</code></td>
<td>Display memory configuration</td>
</tr>
<tr>
<td><code>embed stage device</code></td>
<td>Embed Stage 1.5 in the device</td>
</tr>
<tr>
<td><code>find file</code></td>
<td>Find a file</td>
</tr>
<tr>
<td><code>fstest</code></td>
<td>Toggle filesystem test mode</td>
</tr>
<tr>
<td><code>geometry drive</code></td>
<td>Print information on a drive geometry</td>
</tr>
<tr>
<td><code>halt</code></td>
<td>Shut down the system</td>
</tr>
<tr>
<td><code>help command</code></td>
<td>Show help for a command, or the available commands</td>
</tr>
<tr>
<td><code>impsprobe</code></td>
<td>Probe the Intel Multiprocessor Specification</td>
</tr>
<tr>
<td><code>initrd file</code></td>
<td>Load an initial ramdisk image file</td>
</tr>
<tr>
<td><code>install options</code></td>
<td>Install GRUB (deprecated, use setup instead)</td>
</tr>
<tr>
<td><code>ioprobe drive</code></td>
<td>Probe I/O ports used for a drive</td>
</tr>
<tr>
<td><code>kernel file</code></td>
<td>Load a kernel</td>
</tr>
<tr>
<td><code>lock</code></td>
<td>Lock a GRUB menu entry</td>
</tr>
<tr>
<td><code>makeactive</code></td>
<td>Set active partition on root disk to GRUB's root device</td>
</tr>
<tr>
<td><code>map drive1 drive2</code></td>
<td>Map a drive to another drive</td>
</tr>
<tr>
<td><code>md5crypt</code></td>
<td>Encrypt a password in MD5 format</td>
</tr>
<tr>
<td><code>module file</code></td>
<td>Load a kernel module</td>
</tr>
<tr>
<td><code>modulenounzip file</code></td>
<td>Load a kernel module without decompressing it</td>
</tr>
<tr>
<td><code>pause message</code></td>
<td>Print a message and wait for a key press</td>
</tr>
<tr>
<td><code>quit</code></td>
<td>Quit the GRUB shell</td>
</tr>
<tr>
<td><code>reboot</code></td>
<td>Reboot the system</td>
</tr>
<tr>
<td><code>read address</code></td>
<td>Read a 32-bit value from memory and print it</td>
</tr>
<tr>
<td><code>root device</code></td>
<td>Set the current root device</td>
</tr>
<tr>
<td><code>rootnoflavor device</code></td>
<td>Set the current root device without mounting it</td>
</tr>
<tr>
<td><code>savedefault</code></td>
<td>Save current menu entry as the default entry</td>
</tr>
<tr>
<td><code>setup device</code></td>
<td>Install GRUB automatically on the device</td>
</tr>
<tr>
<td><code>testload file</code></td>
<td>Test the filesystem code on a file</td>
</tr>
<tr>
<td><code>testvbe mode</code></td>
<td>Test a VESA BIOS EXTENSION mode</td>
</tr>
<tr>
<td><code>uppermem kbytes</code></td>
<td>Set the upper memory size (only for old machines)</td>
</tr>
<tr>
<td><code>vbeprobe mode</code></td>
<td>Probe a VESA BIOS EXTENSION mode</td>
</tr>
</tbody>
</table>

### GRUB Legacy configuration file

```
timeout 10   # Boot the default kernel after 10 seconds
default 0    # Default kernel is 0

# Section 0: Linux boot
title Debian   # Menu item to show on GRUB bootmenu
root /dev/hda1 # root filesystem is /dev/hda1
kernel /boot/vmlinuz-2.6.24-19-generic root=/dev/hda1 ro quiet splash
initrd /boot/initrd.img-2.6.24-19-generic

# Section 1: Windows boot
title Microsoft Windows XP
root (hd0,1)   # root filesystem is /dev/hda2
savedefault
makeactive     # set the active flag on this partition
chainloader +1 # read 1 sector from start of partition and run

# Section 2: Firmware/BIOS update from floppy disk
title Firmware update
kernel /floppy.img-7.7.7
initrd /floppy-img-7.7.7
```

---

## Low-level package managers

<table>
<thead>
<tr>
<th><strong>Debian</strong></th>
<th><strong>Red Hat</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dpkg -i</code> package.deb</td>
<td><code>rpm -i package.rpm</code></td>
</tr>
<tr>
<td><code>dpkg -i ftp://host/package.rpm</code></td>
<td><code>rpm -i http://host/package.rpm</code></td>
</tr>
<tr>
<td><code>dpkg -r package</code></td>
<td><code>rpm -e package</code></td>
</tr>
<tr>
<td><code>rpm -U package.rpm</code></td>
<td></td>
</tr>
<tr>
<td><code>rpm -F package.rpm</code></td>
<td></td>
</tr>
<tr>
<td><code>dpkg -l</code></td>
<td><code>rpm -qa</code></td>
</tr>
<tr>
<td><code>rpm -qa --last</code></td>
<td></td>
</tr>
<tr>
<td><code>dpkg -L package</code></td>
<td><code>rpm -ql package</code></td>
</tr>
<tr>
<td><code>rpm2cpio package.rpm</code></td>
<td></td>
</tr>
<tr>
<td><code>rpm -S file</code></td>
<td><code>rpm -V file</code></td>
</tr>
<tr>
<td><code>rpm -i package.src.rpm</code></td>
<td></td>
</tr>
<tr>
<td><code>rpm -ba package.spec</code></td>
<td></td>
</tr>
</tbody>
</table>

## GUI package managers

<table>
<thead>
<tr>
<th><strong>Debian</strong></th>
<th><strong>Red Hat</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>aptitude</code></td>
<td><code>pirut</code></td>
</tr>
<tr>
<td><code>dselect</code></td>
<td></td>
</tr>
<tr>
<td><code>synaptic</code></td>
<td></td>
</tr>
</tbody>
</table>

## Package management utilities

<table>
<thead>
<tr>
<th><strong>Debian</strong></th>
<th><strong>Red Hat</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>alien -i package.rpm</code></td>
<td></td>
</tr>
<tr>
<td><code>rpm2cpio package.rpm</code></td>
<td></td>
</tr>
<tr>
<td><code>apt-key add keyfile</code></td>
<td></td>
</tr>
<tr>
<td><code>createrpo directory</code></td>
<td><code>repoquery --tree-requires package</code></td>
</tr>
<tr>
<td><code>subscription-manager register</code></td>
<td></td>
</tr>
<tr>
<td><code>subscription-manager attach</code></td>
<td></td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>High-level package managers</th>
<th>Debian</th>
<th>Red Hat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install a package</td>
<td><code>apt-get install package</code></td>
<td><code>yum install package</code></td>
</tr>
<tr>
<td>Install a package file</td>
<td><code>apt-get install package.rpm</code></td>
<td><code>yum install package.rpm</code></td>
</tr>
<tr>
<td>Remove a package</td>
<td><code>apt-get remove package</code></td>
<td><code>yum remove package</code></td>
</tr>
<tr>
<td>Upgrade an installed package</td>
<td><code>apt-get upgrade</code></td>
<td><code>yum update package</code></td>
</tr>
<tr>
<td>Upgrade all installed packages</td>
<td><code>apt-get dist-upgrade</code></td>
<td><code>yum update</code></td>
</tr>
<tr>
<td>Upgrade all installed packages and handle dependencies with new versions</td>
<td><code>apt-get dist-upgrade</code></td>
<td><code>yum update</code></td>
</tr>
<tr>
<td>Replace a package with another</td>
<td><code>yum swap packageout packagein</code></td>
<td></td>
</tr>
<tr>
<td>Get the source code for a package</td>
<td><code>apt-get source package</code></td>
<td></td>
</tr>
<tr>
<td>Check for broken dependencies and update package cache</td>
<td><code>apt-get check</code></td>
<td></td>
</tr>
<tr>
<td>Fix broken dependencies</td>
<td><code>apt-get install -f</code></td>
<td></td>
</tr>
<tr>
<td>Update information on available packages</td>
<td><code>apt-get update</code></td>
<td><code>yum list</code></td>
</tr>
<tr>
<td>List all installed and available packages</td>
<td></td>
<td><code>yum list searchterm</code></td>
</tr>
<tr>
<td>List installed and available packages that match the search term</td>
<td></td>
<td><code>yum list searchterm</code></td>
</tr>
<tr>
<td>List installed packages</td>
<td><code>yum list installed</code></td>
<td></td>
</tr>
<tr>
<td>List packages available for install</td>
<td><code>yum list available</code></td>
<td></td>
</tr>
<tr>
<td>Search for a package</td>
<td><code>apt-cache search package</code></td>
<td></td>
</tr>
<tr>
<td>Search for packages that match the search term in the package name or summary</td>
<td></td>
<td><code>yum search searchterm</code></td>
</tr>
<tr>
<td>Search for packages that match the search term in the package name, summary, or description</td>
<td></td>
<td><code>yum search all searchterm</code></td>
</tr>
<tr>
<td>Show package dependencies</td>
<td><code>apt-cache depends package</code></td>
<td><code>yum deplist package</code></td>
</tr>
<tr>
<td>Show package records</td>
<td><code>apt-cache show package</code></td>
<td><code>yum list package</code></td>
</tr>
<tr>
<td>Show information about a package</td>
<td><code>apt-cache showpkg package</code></td>
<td><code>yum info package</code></td>
</tr>
<tr>
<td>Show the installation history (installs, updates, etc.)</td>
<td><code>yum history</code></td>
<td><code>yum history list</code></td>
</tr>
<tr>
<td>Show the installation history about a package</td>
<td><code>yum history package package</code></td>
<td><code>yum history list package package</code></td>
</tr>
<tr>
<td>Update information about package contents</td>
<td><code>apt-file update</code></td>
<td></td>
</tr>
<tr>
<td>List the content of an uninstalled package</td>
<td><code>apt-file list package</code></td>
<td></td>
</tr>
<tr>
<td>Show which package provides a specific file</td>
<td><code>apt-file search file</code></td>
<td><code>yum whatprovides file</code></td>
</tr>
<tr>
<td>Add a CD-ROM to the sources list</td>
<td><code>apt-cdrom add</code></td>
<td></td>
</tr>
<tr>
<td>Download package and all its dependencies</td>
<td></td>
<td><code>yumdownloader --resolve package</code></td>
</tr>
<tr>
<td>Show URLs that would be downloaded</td>
<td></td>
<td><code>yumdownloader --urls package</code></td>
</tr>
<tr>
<td>Try to complete unfinished or aborted package installations</td>
<td></td>
<td><code>yum-complete-transaction</code></td>
</tr>
<tr>
<td>Execute the command but only considering a specific repository</td>
<td><code>yum command --disablerepo=&quot;*** --enablerepo=&quot;repository&quot;</code></td>
<td></td>
</tr>
<tr>
<td>Print list of available repositories</td>
<td><code>cat /etc/apt/sources.list</code></td>
<td><code>yum repolist</code></td>
</tr>
<tr>
<td>Package format</td>
<td>compressed with <code>ar</code></td>
<td>compressed with <code>cpio</code></td>
</tr>
</tbody>
</table>

High-level package managers are able to install remote packages and automatically solve dependencies.
How to install a package on an offline machine

On an online machine 1 (must be identical to the offline machine 2):

1. Do a minimal install of the base packages
2. Install the package and all its dependencies in a local directory
   mkdir /tmp/repo
   yum --downloadonly --downloaddir=/tmp/repo install package
3. Create a local yum repository
   createrepo /tmp/repo
   chown -R root:root /tmp/repo && chmod -R 755 /tmp/repo
4. Transfer the directory /tmp/repo from the online machine 1 to the offline machine 2

On the offline machine 2:

5. Create a yum repo file /etc/yum.repos.d/local.repo for the new repository:

   [local]
   name=Local
   baseurl=file:///tmp/repo
   enabled=1
   gpgcheck=0
   protect=1

6. Install the package from the local repo
   yum install package

---

Configuration for a Fedora repository (Red Hat)

```
[Fedora]
name=Fedora $releasever - $basearch
baseurl=http://download.fedoraproject.org/pub/fedora/
    linux/releases/$releasever/Everything/$basearch/os/
    http://foo.org/linux/$releasever/$basearch/os/
    http://bar.org/linux/$releasever/$basearch/os/
enabled=1
gpgcheck=1
failovermethod=priority
metalink=https://mirrors.fedoraproject.org/metalink?
    repo=fedora-$releasever&arch=$basearch

gpgkey=file:///etc/pki/rpm-gpg/
    RPM-GPG-KEY-fedora-$releasever-$basearch
```

Repository ID
Repository name
List of URLs to the repository’s repodata directory. Can be any of these types:
file:/// local file
file:// NFS
http:// HTTP
https:// HTTPS
ftp:// FTP
Whether this repository is enabled
Whether to perform a GPG signature check on the packages downloaded from this repository
Makes yum try the baseurls in the order they’re listed. By default, if more than one baseurl is specified, yum chooses one randomly
URL to a metalink file that specifies the list of mirrors to use. Can be used with or in alternative to a baseurl
ASCII-armored GPG public key file of the repository

This repository configuration must be located in a repo file e.g. /etc/yum.repos.d/fedora.repo. The same repo file can contain multiple repository definitions.
The manpage man yum.conf lists all repository configuration options.
Backup

**dd**

Tool to copy data, byte by byte, from a file or block device. Should not be used on a mounted block device, because of write cache issues.

```
dd if=/dev/sda of=/dev/sdb
```

Copy the content of one hard disk over another

```
cat /dev/sda > /dev/sdb
```

Generate the image file of a partition

```
zd if=/dev/cdrom of=cdrom.iso bs=2048
```

Create an ISO file from a CD-ROM, using a block size transfer of 2 Kb

```
dd if=install.iso of=/dev/sdc bs=512k
```

Write an installation ISO file to a device (e.g. a USB thumb drive)

**ddrescue**

Tool for data recovery. Like *dd*, but with high tolerance for read errors

**rsync**

Tool for local and remote file synchronization. For all copies subsequent to the first, copies only the blocks that have changed, making it a very efficient backup solution in terms of speed and bandwidth

```
rsync -rzv /home /tmp/bak
rsync -rzv /home/ /tmp/bak/home
rsync -avz /home root@10.0.0.7:/backup/
```

Synchronize the content of the home directory with the temporary backup directory. Use recursion, compression, and verbosity

Synchronize the content of the home directory with the backup directory on the remote server, using SSH. Use archive mode (i.e. operates recursively and preserves owner, group, permissions, timestamps, and symlinks)

**burp**

Backup and restore program

---

### Tape libraries

<table>
<thead>
<tr>
<th>Devices</th>
<th>(/dev/st0)</th>
<th>(/dev/nst0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>First SCSI tape device</td>
<td></td>
<td></td>
</tr>
<tr>
<td>First SCSI tape device (no-rewind device file)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Utility for magnetic tapes</th>
<th>mt -f (/dev/nst0) asf 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position the tape at the start of 3rd file</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Utility for tape libraries</th>
<th>mtx -f (/dev/sgl) status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display status of tape library</td>
<td></td>
</tr>
<tr>
<td>mtx -f (/dev/sgl) load 3</td>
<td></td>
</tr>
<tr>
<td>Load tape from slot 3 to drive 0</td>
<td></td>
</tr>
<tr>
<td>mtx -f (/dev/sgl) unload</td>
<td></td>
</tr>
<tr>
<td>Unlock tape from drive 0 to original slot</td>
<td></td>
</tr>
<tr>
<td>mtx -f (/dev/sgl) transfer 3 4</td>
<td></td>
</tr>
<tr>
<td>Transfer tape from slot 3 to slot 4</td>
<td></td>
</tr>
<tr>
<td>mtx -f (/dev/sgl) inventory</td>
<td></td>
</tr>
<tr>
<td>Force robot to rescan all slots and drives</td>
<td></td>
</tr>
<tr>
<td>mtx -f (/dev/sgl) inquiry</td>
<td></td>
</tr>
<tr>
<td>Inquiry about SCSI media device (Medium Changer = tape library)</td>
<td></td>
</tr>
</tbody>
</table>
### Archive Formats

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`ls</td>
<td>cpio -o &gt; archive.cpio`</td>
</tr>
<tr>
<td>`ls</td>
<td>cpio -oF archive.cpio`</td>
</tr>
<tr>
<td><code>cpio -id &lt; archive.cpio</code></td>
<td>Extract all files, recreating the directory structure</td>
</tr>
<tr>
<td><code>cpio -i -t &lt; archive.cpio</code></td>
<td>List the contents of a cpio archive file</td>
</tr>
<tr>
<td>`ls</td>
<td>cpio -o &gt; archive.cpio`</td>
</tr>
<tr>
<td>`ls</td>
<td>cpio -oF archive.cpio`</td>
</tr>
<tr>
<td><code>cpio -id &lt; archive.cpio</code></td>
<td></td>
</tr>
<tr>
<td><code>cpio -i -t &lt; archive.cpio</code></td>
<td></td>
</tr>
</tbody>
</table>

#### cpio

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>gzip file</code></td>
<td>Compress a file with gzip</td>
</tr>
<tr>
<td><code>gzip &lt; file &gt; file.gzip</code></td>
<td>Compress a file with gzip, leaving the original file intact</td>
</tr>
<tr>
<td><code>gunzip file.gz</code></td>
<td>Decompress a gzip-compressed file</td>
</tr>
<tr>
<td><code>gunzip -tv file.gz</code></td>
<td>Test the integrity of a gzip-compressed file</td>
</tr>
<tr>
<td><code>zcat file.gz</code></td>
<td>Read a gzip-compressed text file</td>
</tr>
<tr>
<td><code>zgrep pattern file.gz</code></td>
<td>grep for a gzip-compressed text file</td>
</tr>
<tr>
<td><code>zless file.gz</code></td>
<td>less for a gzip-compressed text file</td>
</tr>
<tr>
<td><code>zmore file.gz</code></td>
<td>more for a gzip-compressed text file</td>
</tr>
<tr>
<td><code>pigz file</code></td>
<td>Parallel, multicore-optimized gzip</td>
</tr>
</tbody>
</table>

#### gzip

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bzip2 file</code></td>
<td>Compress a file with bzip2</td>
</tr>
<tr>
<td><code>unzip file.bz2</code></td>
<td>Decompress a bzip2-compressed file</td>
</tr>
<tr>
<td><code>bzcat file.bz2</code></td>
<td>Read a bzip2-compressed text file</td>
</tr>
</tbody>
</table>

#### bzip2

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>7z a -t7z archive.7z dir/</code></td>
<td>Create a 7-Zip archive (has the highest compression ratio)</td>
</tr>
<tr>
<td><code>xz file</code></td>
<td>Compress a file with xz</td>
</tr>
<tr>
<td><code>unxz file.xz</code></td>
<td>Decompress a xz-compressed file</td>
</tr>
<tr>
<td><code>xzcat file.xz</code></td>
<td>Read a xz-compressed file</td>
</tr>
</tbody>
</table>

#### 7-Zip

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>lzma file</code></td>
<td>Compress a file with LZMA</td>
</tr>
<tr>
<td><code>ulzma file.lzma</code></td>
<td>Decompress a LZMA-compressed file</td>
</tr>
<tr>
<td><code>lzcat file.lzma</code></td>
<td>Read a LZMA-compressed file</td>
</tr>
</tbody>
</table>

#### LZMA

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>rar a archive.rar dir/</code></td>
<td>Create a RAR archive</td>
</tr>
<tr>
<td><code>unrar x archive.rar</code></td>
<td>Extract a RAR archive</td>
</tr>
</tbody>
</table>

#### rar

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>tar cf archive.tar dir/</code></td>
<td>Create a tarred archive (bundles multiple files in a single one)</td>
</tr>
<tr>
<td><code>tar czf archive.tar.gz dir/</code></td>
<td>Create a tarred gzip-compressed archive</td>
</tr>
<tr>
<td><code>tar xzf archive.tar.gz</code></td>
<td>Extract a tarred gzip-compressed archive</td>
</tr>
<tr>
<td><code>tar cJf archive.tar.bz2 dir/</code></td>
<td>Create a tarred bzip2-compressed archive</td>
</tr>
<tr>
<td><code>tar xJf archive.tar.bz2</code></td>
<td>Extract a tarred bzip2-compressed archive</td>
</tr>
<tr>
<td><code>tar cJf archive.tar.xz dir/</code></td>
<td>Create a tarred xz-compressed archive</td>
</tr>
<tr>
<td><code>tar xJf archive.tar.xz</code></td>
<td>Extract a tarred xz-compressed archive</td>
</tr>
<tr>
<td><code>tar tf archive.tar</code></td>
<td>List the contents of a tarred archive</td>
</tr>
</tbody>
</table>

#### tar

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>star -c -f=archive.star dir/</code></td>
<td>Create a star archive</td>
</tr>
<tr>
<td><code>star -x -f=archive.star</code></td>
<td>Extract a star archive</td>
</tr>
</tbody>
</table>

#### star
**Documentation**

**man command**  
Show the manpage for a command

**man 7 command**  
Show section 7 of the command manpage

**man man**  
Show information about manpages’ sections:  
1 - Executable programs or shell commands  
2 - System calls (functions provided by the kernel)  
3 - Library calls (functions within program libraries)  
4 - Special files  
5 - File formats and conventions  
6 - Games  
7 - Miscellaneous  
8 - System administration commands (usually only for root)  
9 - Kernel routines

**mandb**  
Generate or refresh the search database for manpage entries. This must be done after installing new packages, in order to obtain meaningful results from `apropos` or `man -k`

**apropos keyword**  
Show the commands whose manpage’s short description matches the keyword.  
Inverse of the `what is` command

**apropos -r regex**  
Show the commands whose manpage’s short description matches the regex

**man -k regex**  
Show the commands whose manpage’s full text matches the regex

**whatis command**  
Show the manpage’s short description for a command

**info command**  
Show the Info documentation for a command

**help**  
Show the list of available shell commands and functions

**help command**  
Show help about a shell command or function
### History

- **history**
  - Show the history of command lines executed up to this moment.
  - Commands prepended by a space will be executed but will not show up in the history.
  - After the user logs out from Bash, history is saved into `~/.bash_history`

- **!n**
  - Execute command number `n` in the command line history

- **history -c**
  - Clear the command line history

- **history -d n**
  - Delete command number `n` from the command line history

- **alias ls='ls -lap'**
  - Set up an alias for the `ls` command

- **alias**
  - Show defined aliases

- **unalias ls**
  - Remove the alias for the `ls` command

- **`
ls**
  - Run the non-aliased version of the `ls` command

- **/bin/ls**
  - Run the non-aliased version of the `ls` command

---

Almost all Linux commands accept the option `-v` (verbose), and some commands also accept the options `-vv` or `-vvv` (increasing levels of verbosity).

---

All Bash built-in commands, and many other commands, accept the flag `--` which denotes the end of options and the start of positional parameters:

- **grep -- -i file**
  - Search for the string "-i" in `file`

- **rm -- -rf**
  - Delete a file called "-rf"

---

- **cat /etc/debian_version**
  - (Debian)
  - Display Linux distribution name and version

- **cat /etc/fedora-release**
  - (Fedora)

- **cat /etc/redhat-release**
  - (Red Hat)

- **cat /etc/lsb-release**

- **lsb_release -- -a**

- **cat /etc/os-release**
cat file
Print a text file

cat file1 file2 > file3
Concatenate text files

cat file1 > file2
Copy file1 to file2. The cat command is able to operate on binary streams as well
> file2 < file1 cat
and therefore it works also with binary files (e.g. JPG images)

cat > file <<EOF
line 1
line 2
line 3
EOF
Create a **Here Document**, storing the lines entered in input to file

cat -etv <<< 'string'
Create a **Here String**, passing string as input to command

Print string, showing all invisible characters

tac file
Print or concatenate text files in opposite order line-wise, from last line to first line

rev file
Print a text file with every line reversed character-wise, from last char to first char

head file
Print the first 10 lines of a text file

head -n 10 file

head -n 10 file

head -f file

head -n +1 file1 file2 file3

Print each file with a filename header

multitail -i file1 -i file2
tail for multiple files at the same time (ncurses UI)

column file
Format a text file into columns

pr file
Format a text file for a printer

fmt -w 75 file
Format a text file so that each line has a max width of 75 characters

fold -w40 file
Wrap each line of a text file to 40 characters

nl file
Prepend line numbers to a text file

wc file
Print the number of lines, words, and bytes of a text file

join file1 file2
Join lines of two text files on a common field

paste file1 file2
Merge lines of text files

split -l 1 file
Split a text file into 1-line files; these will be named xaa, xab, xac, etc.

uniq file
Print the unique lines of a text file, omitting consecutive identical lines

sort file
Sort alphabetically the lines of a text file

shuf file
Shuffle randomly the lines of a text file

expand file
Convert tabs into spaces

unexpand file
Convert spaces into tabs

diff file1 file2
Compare two text files line by line and print the differences

cmp file1 file2
Compare two files and print the differences
Advanced text filters

- `cut -d: -f3 file`  
  Cut the lines of a file, considering : as the delimiter and printing only the 3rd field.

- `cut -d: -f1 /etc/passwd`  
  Print the list of user accounts in the system.

- `cut -c3-50 file`  
  Print character 3 to 50 of each line of a file.

- `sed 's/foo/bar/' file`  
  Stream Editor: Replace the first occurrence on a line of "foo" with "bar" in file, and print on stdout the result.

- `sed -i 's/foo/bar/' file`  
  Replace "foo" with "bar", overwriting the results in file.

- `sed 's/foo/bar/g' file`  
  Replace all occurrences of "foo" with "bar".

- `sed 's/foo/bar/7,13p' file`  
  Replace "foo" with the value of variable $var. The double quotes allow for variable expansion.

- `tr a-z A-Z < file`  
  Translate characters: Convert all lowercase into uppercase in a text file.

- `tr [:lower:] [:upper:] < file`  
  Delete all digits from a text file.

- `awk`  
  Interpreter for the AWK programming language, designed for text processing and data extraction.

- `grep foo file`  
  Print the lines of a file containing "foo".

- `grep -v foo file`  
  Print the lines of a file not containing "foo".

- `grep -e foo -e bar file`  
  Print the lines of a file containing "foo" or "bar".

- `egrep`  
  Print the lines of a file matching the given Extended Regex.

- `tail -f file | grep --line-buffered foo`  
  Output appended data as the text file grows, printing only the lines containing "foo".

- `stdbuf option command`  
  Run command with modified stdin, stdout, or stderr buffering.

- `rpl oldstring newstring file`  
  Replace strings in a file.

- `tidy`  
  Correct and tidy up the markup of HTML, XHTML, and XML files.

- `tidy -asxml -xml -indent -wrap 2000 \ -quiet --hide-comments yes file.xml`  
  Strip out comments from an XML file.

- `json_verify < file.json`  
  Validate the syntax of a JSON file.

- `json_reformat < file.json`  
  Pretty format a JSON file.

- `strings file`  
  Show all printable character sequences at least 4-characters long that are contained in file.

- `antiword file.doc`  
  Show text and images from a MS Word document.

- `catdoc file.doc`  
  Output plaintext from a MS Word document.
Regular expressions

^  Beginning of a line
$  End of a line
\< \> Word boundaries (beginning of line, end of line, space, or punctuation mark)
-  Any character except newline
\[abc\] Any of the characters specified
\[a-z\] Any of the characters in the specified range
[^abc] Any character except those specified
*  Zero or more times the preceding regex
+  One or more times the preceding regex
?  Zero or one time the preceding regex
\{5\} Exactly 5 times the preceding regex
\{,10\} At most 10 times the preceding regex
\{5,10\} Between 5 and 10 times the preceding regex
\1  The regex either before or after the vertical bar
\( ) Grouping, to be used for back-references. \1 expands to the 1\textsuperscript{st} match, \2 to the 2\textsuperscript{nd}, and so on until \9

The symbols above are used in POSIX EREs (Extended Regular Expressions). In POSIX BREs (Basic Regular Expressions), the symbols ? + { | ( ) need to be escaped (by adding a backslash character \ in front of them).
### File management

- **cp file file2**
  - Copy a file

- **cp file dir/**
  - Copy a file to a directory

- **cp -ar /dir1/. /dir2/**
  - Copy a directory recursively

- **mv file file2**
  - Rename a file

- **mv file dir/**
  - Move a file to a directory

- **rm file**
  - Delete a file

- **pv file > file2**
  - Copy a file, monitoring the progress of data through a pipe

- **touch file**
  - Change access timestamp and modify timestamp of a file as now. If the file does not exist, it is created

- **mktemp**
  - Create a temporary file or directory, using `tmp.XXXXXXXXXX` as filename template

- **stat file**
  - Display file or filesystem status

- **stat -c %A file**
  - Display file permissions

- **stat -c %s file**
  - Display file size, in bytes

- **shred /dev/hda**
  - Securely wipe the contents of a device

- **shred -u file**
  - Securely delete a file

- **fdupes dir**
  - Examines a directory for duplicate files in it. To consider files a duplicate, first compares file sizes and MD5 signatures, then compares the file contents byte-by-byte

- **tmpwatch**
  - Remove files which have not been accessed for some time

- **lsof**
  - List all open files

- **lsof -u user**
  - List all files currently open by user

- **lsof -l**
  - List open files and their sockets (equivalent to `netstat -ap`)

- **lsof -l:80**
  - List connections of local processes on port 80

- **lsof -i@10.0.0.3**
  - List connections of local processes to remote host 10.0.0.3

- **lsof -i@10.0.0.3:80**
  - List connections of local processes to remote host 10.0.0.3 on port 80

- **lsof -c mysql**
  - List all files opened by `mysql`, the MySQL daemon

- **lsof file**
  - List all processes using a specific file

- **lsof +L1**
  - List all processes using an unlinked file. These processes, until killed or restarted, hold the file open preventing it from being deleted (and thus freeing disk space)

- **lslocks**
  - List information about all currently held file locks

- **unlink file**
  - Remove a link to a file (equivalent to `rm`)

- **od file**
  - Dump a file into octal (or other formats)

- **hexdump options file**
  - Dump a file into hexadecimal (or other formats e.g. octal, decimal, ASCII)

- **xxd options file**
  - Convert a file from binary to hexadecimal, or vice versa
# Directory management

- `cd directory` Change to the specified directory  
- `cd -` Change to the previously used directory  
- `pwd` Print the current working directory  
- `ls` List the contents of the current directory  
- `dir` List only directories contained on the current directory  
- `vdir` List files, sorted by version number  
- `ls -d */` Create a directory  
- `mkdir dir` Create a directory  
- `mkdir -m 755 dir` Create a directory with mode 755  
- `mkdir -p /dir1/dir2/dir3` Create a directory, creating also the parent directories if they don’t exist  
- `rmdir dir` Delete a directory (which must be empty)  
- `tree` List directories and their contents in hierarchical format  
- `pushd dir` Add `dir` to the top of the directory stack and make it the current working directory  
- `popd` Remove the top directory from the directory stack and change to the new top directory  
- `dirs` Display the directory stack (i.e. the list of remembered directories)  
- `dirname file` Output the directory path in which `file` is located, stripping any non-directory suffix from the filename

### Bash directory shortcuts

<table>
<thead>
<tr>
<th>Shortcut</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>Current directory</td>
</tr>
<tr>
<td>..</td>
<td>Parent directory</td>
</tr>
<tr>
<td>~</td>
<td>Home directory of current user</td>
</tr>
<tr>
<td>~user</td>
<td>Home directory of user</td>
</tr>
<tr>
<td>~-</td>
<td>Previously used directory</td>
</tr>
</tbody>
</table>

### File-naming wildcards (globbing)

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>Matches zero or more characters</td>
</tr>
<tr>
<td>?</td>
<td>Matches one character</td>
</tr>
<tr>
<td>[abc]</td>
<td>Matches a, b, or c</td>
</tr>
<tr>
<td>[!abc]</td>
<td>Matches any character except a, b, or c</td>
</tr>
<tr>
<td>[a-z]</td>
<td>Matches any character between a and z</td>
</tr>
</tbody>
</table>

### Brace expansion

- `cp foo.{txt,bak}` Copy file “foo.txt” to “foo.bak”  
- `touch foo_{a,b,c}` Create files "foo_a", "foo_b", "foo_c"
In Linux, everything is (displayed as) a file. File descriptors are automatically associated to any process launched.

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Type</th>
<th>Default device</th>
<th>Device file</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standard input (stdin)</td>
<td>Input text stream</td>
<td>Keyboard</td>
<td>/dev/stdin</td>
</tr>
<tr>
<td>1</td>
<td>Standard output (stdout)</td>
<td>Output text stream</td>
<td>Terminal</td>
<td>/dev/stdout</td>
</tr>
<tr>
<td>2</td>
<td>Standard error (stderr)</td>
<td>Output text stream</td>
<td>Terminal</td>
<td>/dev/stderr</td>
</tr>
</tbody>
</table>

mail user@email < file
Redirect file to the stdin of command mail (in this case, send via e-mail the contents of file to the email address user@email). Redirection is handled by the shell, not by the command invoked. The space after the redirect operator is optional.

ls > file
Redirect the stdout of command ls to file (in this case, write on file the contents of the current directory). This overwrites file if it already exists, unless the Bash noclobber option is set (via `set -o noclobber`).

ls >| file
Redirect the stdout of command ls to file, even if noclobber is set.

ls >> file
Append the stdout of command ls to file.

ls 2> file
Redirect the stderr of command ls to file (in this case, write any error encountered by the command ls to file).

ls 2>> file
Append the stderr of command ls to file.

ls 2>/dev/null
Silence any error coming from the command ls.

cat <file1 >file2
cat >|file2
<file1 >file2 cat
Redirect file1 to the stdin and file2 to the stdout of the command cat (in this case, copy file1 to file2).

sort file | sponge file
Sort the contents of file and write the output to the file itself. sort file > file would not produce the desired result, because the stdout destination is created (and therefore the content of the preexisting file is deleted) before the sort command is run.

ls 2>>|1
Redirect both stdout and stderr of command ls to file.
ls >> file and ls >| file also work on some systems but are not recommended, because they are not POSIX standard.

> file
Create an empty file. If the file exists, its content will be deleted.

ls | tee file
tee reads from stdin and writes both to stdout and file (in this case, writes the contents of the current directory to screen and to file at the same time).

ls | tee -a file
tee reads from stdin and appends both to stdout and file.
```bash
while read -r line
do
  echo "Hello $line"
done < file

read MYVAR
read -n 8 MYVAR
read -t 60 MYVAR
read -s MYVAR

echo $MYVAR
echo -n "message"
printf "message"
echo -e '\a'

pv -qL10 <<< "message"
```

Process a text file line by line, reading from file. If file is /dev/stdin, reads from standard input instead

Read a variable from standard input
Read only max 8 chars from standard input
Read a variable from standard input, timing out after one minute
Read a variable from standard input without echoing to terminal (silent mode)

Print a variable on screen
Print message onscreen without a trailing line feed

Produce an alert sound (BEL sequence)

Print message onscreen, one character at a time
Any application, program, or script that runs on the system is a process. Signals are used for inter-process communication. Each process has a unique PID (Process ID) and a PPID (Parent Process ID); when a process spawns a child, the process PID is assigned to the child's PPID.

The /sbin/init process, run at bootup, has PID 1. It is the ancestor of all processes and becomes the parent of any orphaned process. It is also unkillable; should it die, the kernel will panic.

When a child process dies, its status becomes EXIT_ZOMBIE and a SIGCHLD is sent to the parent. The parent should then call the wait() system call to read the dead process’ exit status and other info; until that moment, the child process remains a zombie.

```
ps -ef (UNIX options)  List all processes
ps aux (BSD options)  
pstree PID  Display all processes in hierarchical format.
The process tree is rooted at PID, or at init if PID is omitted
pidof process  Show PID of process

kill -9 1138  Send a signal 9 (SIGKILL) to process 1138, hence killing it
killall -9 sshd  Kill processes whose name is "sshd"
pgrep sshd  Show processes whose name is "sshd"
ps -ef | grep "[s]shd"  
pgrep -u root sshd  Show processes whose name is "sshd" and are owned by root
pkill -9 -u root sshd  Kill processes whose name is "sshd" and are owned by root
xkill  Kill a process by its X GUI resource. Pops up a cursor to select a window

jobs  List all jobs (i.e. processes whose parent is a Bash shell)
CTRL + Z  Suspend a job, putting it in the stopped state (send a SIGTSTP)
bg %1  Put job #1 in the background (send a SIGCONT)
fg %1  Resume job #1 in the foreground and make it the current job (send a SIGCONT)
kill %1  Kill job #1

To each process is associated a niceness value: the higher the niceness, the lower the priority. The niceness value ranges from -20 to 19, and a newly created process has a default niceness of 0. Unprivileged users can modify a process’ niceness only within the range from 1 to 19.

nice -n -5 command  Start a command with a niceness of -5. If niceness is omitted, a default value of 10 is used
renice -5 command  Change the niceness of a running command to -5

strace command  Trace the execution of command, intercepting and printing the system calls called by a process and the signals received by a process
ipcs  Show IPC facilities information (shared memory, message queues, and semaphores)
pmap PID  Display the memory map of process PID

():(){ :|:& };:  Fork bomb: starts a process that continually replicates itself, slowing down or crashing the system because of resource starvation

{ command }& pid=$!; sleep n; kill -9 $pid  Run a command and kill it after n seconds
### Signals

The most frequently used signals are:

<table>
<thead>
<tr>
<th>Signal number</th>
<th>Signal name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SIGHUP</td>
<td>Used by many daemons to reload their configuration</td>
</tr>
<tr>
<td>2</td>
<td>SIGINT</td>
<td>Interrupt, stop</td>
</tr>
<tr>
<td>9</td>
<td>SIGKILL</td>
<td>Kill unconditionally (this signal cannot be ignored)</td>
</tr>
<tr>
<td>15</td>
<td>SIGTERM</td>
<td>Terminate gracefully</td>
</tr>
<tr>
<td>18</td>
<td>SIGCONT</td>
<td>Continue execution</td>
</tr>
<tr>
<td>20</td>
<td>SIGTSTP</td>
<td>Stop execution</td>
</tr>
</tbody>
</table>

The manpage `man 7 signal` lists all signal numbers and names.

- `kill -l` List all available signal names
- `kill -l n` Print the name of signal number `n`
- `trap action condition` Trap a signal
Resource monitoring

vmstat
Print a report about virtual memory statistics: processes, memory, paging, block I/O, traps, disks, and CPU activity

iostat
Print a report about CPU utilization, device utilization, and network filesystem. The first report shows statistics since the system boot; subsequent reports will show statistics since the previous report.

mpstat
Print a report about processor activities

vmstat 2 5
iostat 2 5
mpstat 2 5
Print the relevant report every 2 seconds, for 5 times

free
Show the amount of free and used memory in the system

top
Monitor processes in real-time

htop
Monitor processes in real-time (ncurses UI)

iostop
Display I/O usage by processes in the system

atop
Advanced system monitor that displays the load on CPU, RAM, disk, and network

powertop
Power consumption and power management diagnosis tool

uptime
Show how long the system has been up, how many users are connected, and the system load averages for the past 1, 5, and 15 minutes

time command
Execute command and, at its completion, write to stderr timing statistics about the run: elapsed real time between invocation and termination, user CPU time, system CPU time

sar
Show reports about system activity. Reports are generated from data collected via the cron job sysstat and stored in /var/log/sa/sn, where n is the day of the month

sar -n DEV
Show reports about network activity (received and transmitted packets per second)

sar -f /var/log/sa/s19 -s 06:00:00 -e 06:30:00
Show reports for system activity from 6 to 6:30 AM on the 19th of the month

sysbench
Multi-threaded benchmark tool able to monitor different OS parameters: file I/O, scheduler, memory allocation, thread implementation, databases

inxi
Debugging tool to rapidly and easily gather system information and configuration

<table>
<thead>
<tr>
<th>Linux monitoring tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>collectd</td>
</tr>
<tr>
<td>Nagios</td>
</tr>
<tr>
<td>MRTG</td>
</tr>
<tr>
<td>Cacti</td>
</tr>
<tr>
<td>Munin</td>
</tr>
<tr>
<td>Zabbix</td>
</tr>
<tr>
<td>Centreon</td>
</tr>
<tr>
<td>netdata</td>
</tr>
</tbody>
</table>
### Output of command `vmstat`

<table>
<thead>
<tr>
<th>proc</th>
<th>swpd</th>
<th>mem</th>
<th>buff</th>
<th>cache</th>
<th>si</th>
<th>so</th>
<th>bi</th>
<th>bo</th>
<th>in</th>
<th>cs</th>
<th>us</th>
<th>sy</th>
<th>id</th>
<th>wa</th>
<th>st</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>296724</td>
<td>267120</td>
<td>3393400</td>
<td>0</td>
<td>0</td>
<td>17</td>
<td>56</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>95</td>
</tr>
</tbody>
</table>

- **procs**
  - `r`: Number of runnable processes (running or waiting for run time)
  - `b`: Number of processes in uninterruptible sleep

- **memory**
  - `swpd`: Virtual memory used (swap)
  - `free`: Free memory (idle)
  - `buff`: Memory used as buffers
  - `cache`: Memory used as cache
  - `si`: Memory swapped in from disk
  - `so`: Memory swapped out to disk

- **swap**
  - `bi`: Blocks received in from a block device
  - `bo`: Blocks sent out to a block device

- **io**
  - `in`: Number of interrupts

- **system**
  - `cs`: Number of context switches
  - `us`: Time spent running user code (non-kernel)
  - `sy`: Time spent running system code (kernel)

- **cpu**
  - `id`: Time spent idle
  - `wa`: Time spent waiting for I/O
  - `st`: Time stolen from a virtual machine

### Output of command `free`

<table>
<thead>
<tr>
<th>Mem</th>
<th>Mem:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>16344088</td>
<td>2273312</td>
<td>11531400</td>
<td>776228</td>
<td>2539376</td>
<td>12935112</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swap:</td>
<td>1048572</td>
<td>0</td>
<td>1048572</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Swap:</th>
<th>Mem:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1504544</td>
<td>1491098</td>
<td>13021</td>
<td>0</td>
<td>91112</td>
<td>764542</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+/- buffers/cache:</td>
<td>635212</td>
<td>869498</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Swap:</td>
<td>2047686</td>
<td>7667</td>
<td>2040019</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mem</th>
<th>total</th>
<th>used</th>
<th>free</th>
<th>shared</th>
<th>buff/cache</th>
<th>available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mem:</td>
<td>16344088</td>
<td>2273312</td>
<td>11531400</td>
<td>776228</td>
<td>2539376</td>
<td>12935112</td>
</tr>
<tr>
<td>Swap:</td>
<td>1048572</td>
<td>0</td>
<td>1048572</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>+/- buffers/cache:</th>
<th>used</th>
<th>free</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mem:</td>
<td>635212</td>
<td>869498</td>
</tr>
<tr>
<td>Swap:</td>
<td>2047686</td>
<td>7667</td>
</tr>
</tbody>
</table>

### Notes

- **Mem**
  - `total`: Total configured amount of memory
  - `used`: Used memory
  - `free`: Unused memory
  - `shared`: Memory used by tmpfs, 0 if not available
  - `buff/cache`: Memory used by kernel buffers, page cache, and slabs
  - `available`: Memory available for new applications (without using swap)

- **Swap**
  - `total`: Total configured amount of swap space
  - `used`: Used swap space
  - `free`: Free swap space

* These are the true values indicating the free system resources available. All values are in Kb, unless options are used.
File permissions

Permission Octal value Command Effect on file Effect on directory
--- --- --- --- ---
Read
user: 400 chmod u+r Can open and read the file Can list directory content
  group: 40 chmod g+r
  others: 4 chmod o+r
Write
user: 200 chmod u+w Can modify the file Can create, delete, and rename files in the directory
  group: 20 chmod g+w
  others: 2 chmod o+w
Execute
user: 100 chmod u+x Can execute the file (binary or script) Can enter the directory, and search files within (by accessing a file’s inode)
  group: 10 chmod g+x
  others: 1 chmod o+x
SetUID (SUID) 4000 chmod u+s Executable is run with the privileges of the file’s owner No effect
SetGID (SGID) 2000 chmod g+s Executable is run with the privileges of the file’s group All new files and subdirectories inherit the directory’s group ID
Sticky 1000 chmod +t No effect Files inside the directory can be deleted or moved only by the file’s owner

chmod 711 file
chmod u=rw,x,go=x file
chmod u+wx file
chmod -x file

chmod -R g+x /path
find /path -type d 
-exec chmod g+x {} ";

chown user file
chown user:group file
chown :group file
chgrp group file

umask 022

Set read, write, and execute permission to user; set execute permission to group and others
Add write and execute permission to user
Remove execute permission from everybody (user, group, and others)
Set the group execute bit recursively on path and every dir and file underneath
Set the group execute bit recursively on path and every dir, but not file, underneath
Change the owner of the file to user
Change the owner of the file to user, and group ownership of the file to group
Change group ownership of the file to group

Linux default permissions are 0666 for files and 0777 for directories. These base permissions are ANDed with the inverted umask value to calculate the final permissions of a new file or directory.
### File attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>File can only be opened in append mode for writing</td>
</tr>
<tr>
<td>A</td>
<td>When file is accessed, its atime record is not modified</td>
</tr>
<tr>
<td>c</td>
<td>File is automatically compressed on-the-fly on disk by the kernel</td>
</tr>
<tr>
<td>C</td>
<td>File is not subject to copy-on-write updates. This applies only to filesystems which perform copy-on-write</td>
</tr>
<tr>
<td>d</td>
<td>File will not be backed up by the <code>dump</code> program</td>
</tr>
<tr>
<td>D</td>
<td>When directory is modified, changes are written synchronously on disk. Equivalent to <code>dirsync</code> mount option</td>
</tr>
<tr>
<td>e</td>
<td>File is using extents for mapping the blocks on disk</td>
</tr>
<tr>
<td>E</td>
<td>Compression error on file. This attribute is used by experimental compression patches</td>
</tr>
<tr>
<td>h</td>
<td>File stores its blocks in units of filesystem blocksize instead of in units of sectors, and is larger than 2 Tb</td>
</tr>
<tr>
<td>i</td>
<td>File is immutable i.e. cannot be modified, linked, or changed permissions</td>
</tr>
<tr>
<td>I</td>
<td>Directory is being indexed using hashed trees</td>
</tr>
<tr>
<td>j</td>
<td>All file data is written to the ext3 or ext4 journal before being written to the file itself</td>
</tr>
<tr>
<td>N</td>
<td>File has data stored inline within the inode itself</td>
</tr>
<tr>
<td>s</td>
<td>File will be securely wiped by zeroing when deleted</td>
</tr>
<tr>
<td>S</td>
<td>When file is modified, changes are written synchronously on disk. Equivalent to the <code>sync</code> mount option</td>
</tr>
<tr>
<td>t</td>
<td>File will not have EOF partial block fragment merged with other files. This applies only to filesystems with support for tail-merging</td>
</tr>
<tr>
<td>T</td>
<td>Directory is the top of directory hierarchies for the purpose of the Orlov block allocator</td>
</tr>
<tr>
<td>u</td>
<td>After file is deleted, it can be undeleted</td>
</tr>
<tr>
<td>X</td>
<td>Raw contents of compressed file can be accessed directly. This attribute is used by experimental compression patches</td>
</tr>
<tr>
<td>Z</td>
<td>Compressed file is dirty. This attribute is used by experimental compression patches</td>
</tr>
</tbody>
</table>

**chattr** `+attribute file`  
Add a file or directory attribute

**chattr** `-attribute file`  
Remove a file or directory attribute

**chattr** `=attribute file`  
Set a file or directory attribute, removing all other attributes

**lsattr** `file`  
List file or directory attributes

<table>
<thead>
<tr>
<th>Timestamp</th>
<th>Value tracked</th>
<th>Displayed via</th>
</tr>
</thead>
<tbody>
<tr>
<td>mtime</td>
<td>Time of last modification to file contents (data itself)</td>
<td><code>ls -l</code></td>
</tr>
<tr>
<td>ctime</td>
<td>Time of last change to file contents or file metadata (owner, group, or permissions)</td>
<td><code>ls -lc</code></td>
</tr>
<tr>
<td>atime</td>
<td>Time of last access to file for reading contents</td>
<td><code>ls -lu</code></td>
</tr>
</tbody>
</table>

The POSIX standard does not define a timestamp for file **creation**. Some filesystems (e.g. ext4, JFS, Btrfs) store this value, but currently there is no Linux kernel API to access it.
Access Control Lists (ACLs) provide a fine-grained set of permissions that can be applied to files and directories. An access ACL is set on an individual file or directory; a default ACL is set on a directory, and applies to all files and subdirs created inside it that don't have an access ACL. The final permissions are the intersection of the ACL with the chmod/umask value. A partition must have been mounted with the acl option in order to support ACLs on files.

```
setfacl -m u:user:permissions file
setfacl -m g:group:permissions file
setfacl -m m:permissions file
setfacl -m o:permissions file
setfacl -x u:user file
setfacl -x g:group file
```

The permissions are standard Unix permissions specified as any combination of `rwx`.

```
setfacl -m d:u:user:permissions dir  # Same as above, but set a default ACL instead of an access ACL.
setfacl -d -m u:user:permissions dir
```

```
getfacl file
getfacl file1 | setfacl --set-file=- file2
getfacl --access dir | setfacl -d -M- dir
```

```
chacl options
```

```
man acl
```

Set an access ACL on a file for an user
Set an access ACL on a file for a group
Set the effective rights mask on a file
Set the permissions on a file for other users
Remove an access ACL from a file for an user
Remove an access ACL from a file for a group
Display the access (and default, if any) ACL for a file
Copy the ACL of file1 and apply it to file2
Copy the access ACL of a directory and set it as default ACL
Change an ACL.
This command exists to provide compatibility with IRIX
Show the manpage about ACLs
A Linux directory contains a list of mappings between filenames and inodes. An inode is a structure containing all file metadata: file type, permissions, owner, group, size, access/change/modification/deletion times, number of links, attributes, ACLs, and address where the actual file content (data) is stored. An inode does not contain the name of the file; this information is stored in the directory where the file is located (i.e. referenced).

```
ls -i          Show a listing of the directory with the files' inode numbers

df -i          Report filesystem inode usage
```

<table>
<thead>
<tr>
<th>Hard link</th>
<th>Soft or symbolic link</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Definition</strong></td>
<td>A link to an already existing inode</td>
</tr>
<tr>
<td><strong>Command to create it</strong></td>
<td><code>ln file hardlink</code></td>
</tr>
<tr>
<td><strong>Link is still valid if the original file is moved or deleted</strong></td>
<td>Yes (because the link still references the inode to which the original file pointed)</td>
</tr>
<tr>
<td><strong>Can link to a file in another filesystem</strong></td>
<td>No (because inode numbers make sense only within a determinate filesystem)</td>
</tr>
<tr>
<td><strong>Can link to a directory</strong></td>
<td>No</td>
</tr>
<tr>
<td><strong>Link permissions</strong></td>
<td>Reflect the original file's permissions, even when these are changed</td>
</tr>
<tr>
<td><strong>Link attributes</strong></td>
<td><code>-</code> (regular file)</td>
</tr>
<tr>
<td><strong>Inode number</strong></td>
<td>The same as the original file</td>
</tr>
</tbody>
</table>
find /path -name "foo*"
find /path -name "foo*" -print
find / -name "foo*" -exec chmod 700 {} \
find / -name "foo*" -ok chmod 700 {} \
find / -size +128M
find / -type f -ctime +10
find / -type f -perm -4000
find / -type f -newermt "May 4 2:55" -delete
find . -type f -print -exec cat {} \
find . ! -name "*.gz" -type f -exec gzip {} \
find / -xdev -type f -exec gzip {} \
-exec ls -lah {} \\;

locate file
slocate file
updatedb

which command
which -a command
whereis command
whereis -b command
whereis -s command
whereis -m command
type command
file file

Find all files and dirs, in the directory tree rooted at /path, whose name starts with "foo"
Find all files and dirs whose name start with "foo" and apply permission 700 to all of them
Find all files and dirs whose name start with "foo" and apply permission 700 to all of them, asking for confirmation
Find all files larger than 128 Mb
Find all files last changed more than 10 days ago
Find all files with SUID set (a possible security risk, because a shell with SUID root is a backdoor)
Find and delete all files newer than the specified timestamp.
Using -delete is preferable to using -exec rm {} \\;
Print all files, in the current directory and under, prepending them with a filename header
Find all files, in the current directory and under, which do not have the gz extension, and compress them
Find all files larger than 100 Mb in the current filesystem only and display detailed information about them

Locate file by searching the file index /etc/updatedb.conf, not by actually walking the filesystem. The search is fast but will only yield results relative to the last rebuild of the file index
Rebuild the file index

Locate a binary executable command within the PATH
Locate all matches of a command, not only the first one
Locate the binary, source, and manpage files for a command
Locate the binary files for a command
Locate the source files for a command
Locate the manpage files for a command

Determine if a command is a program or a built-in (i.e. an internal feature of the shell)

Analyze the content of a file or directory, and display the kind of file (e.g. executable, text file, program text, swap file)
The scope of variables is the current shell only, while environment variables are visible within the current shell as well as within all subshells and Bash child processes spawned by the shell. Environment variables are set in /etc/environment in the form variable=value. By convention, variable names are lowercase while environment variable names are uppercase.

- `set` Display all variables
- `env` Display all environment variables
- `export VAR` Export a variable, making it an environment variable
- `VAR=value
  {VAR=value}
  let "VAR=value"
  readonly VAR=value` Set a variable.
  Note that there are no spaces around the equal sign
  Set a variable, making its value unchangeable
- `command $VAR
  command ${VAR}HELLO
  command "$VAR"` Pass a variable as argument to command.
  If other characters follow the variable name, it is necessary to specify the boundaries of the variable name via {} to make it unambiguous.
  It is recommended to double quote the variable when referencing it, to prevent interpretation of special characters (except `\ $ `) and word splitting (in case the variable value contains whitespaces), both of which will have unintended results
- `VAR=${(5+37)}
VAR=${5+37}
VAR=${(VAR2 + 42)}
VAR=`expr $VAR2 + 42`
`VAR=`date
VAR=$(date)` Evaluate a numeric expression and assign the result to another variable
  Assign to a variable the output resulting from a command
- `for i in /path/*/ do
  echo "Filename: $i"
  done` Loop and operate through all the output tokens (in this case, files in the path).
  Note: the equivalent construct for i in ${ls /path/} is unnecessary and harmful, because filenames containing whitespaces or glob characters will have unintended results
- `unset VAR` Delete a variable
- `set ${VAR:=value}
VAR=${VAR:-value}` Set a variable, only if it is not already set (i.e. does not exist) or is null
- `echo ${VAR:=message}
echo ${VAR:+message}
echo ${VAR,,}` If variable exists and is not null, print its value, otherwise print message
  If variable exists and is not null, print message, otherwise print nothing
  Print a string variable in lowercase
- `TOKENS=($STRING)
echo ${TOKENS[n]}
echo ${TOKENS[*]}` String tokenizer. Splits a string stored in the variable STRING into tokens, according to the content of the shell variable $IFS, and stores them in the array TOKENS
  Print the token number n
  Print all tokens
Shell operations

Bash built-in variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>$0</code></td>
<td>Script name</td>
</tr>
<tr>
<td><code>$n</code></td>
<td><code>n</code>th argument passed to the script or function</td>
</tr>
<tr>
<td><code>$8</code></td>
<td>All arguments passed to the script or function; each argument is a separate word</td>
</tr>
<tr>
<td><code>$*</code></td>
<td>All arguments passed to the script or function, as a single word</td>
</tr>
<tr>
<td><code>$#</code></td>
<td>Number of arguments passed to the script or function</td>
</tr>
<tr>
<td><code>$$</code></td>
<td>Exit status of the last recently executed command</td>
</tr>
<tr>
<td><code>$(PIPESTATUS[n])</code></td>
<td>Exit status of the <code>n</code>th command in the executed pipeline</td>
</tr>
<tr>
<td><code>$0</code></td>
<td>PID of the script in which this variable is called</td>
</tr>
<tr>
<td><code>$!</code></td>
<td>PID of the last recently executed background command</td>
</tr>
<tr>
<td><code>$SHLVL</code></td>
<td>Deepness level of current shell, starting with 1</td>
</tr>
<tr>
<td><code>$IFS</code></td>
<td>Internal Field Separator; defines what are the token separators for strings (e.g. for word splitting after expansion). By default it has the value &quot;space, tab, newline&quot;</td>
</tr>
<tr>
<td><code>$RANDOM</code></td>
<td>Pseudorandom integer value between 0 and 32767</td>
</tr>
</tbody>
</table>

Bash shell options

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>set -option</code></td>
<td>Enable a Bash option</td>
</tr>
<tr>
<td><code>set -o longoption</code></td>
<td>Enable a Bash option</td>
</tr>
<tr>
<td><code>set +option</code></td>
<td>Disable a Bash option</td>
</tr>
<tr>
<td><code>set +o longoption</code></td>
<td>Disable a Bash option</td>
</tr>
<tr>
<td><code>set -o</code></td>
<td>Show the status of all Bash options</td>
</tr>
<tr>
<td><code>set -v</code></td>
<td>Print shell input lines as they are read</td>
</tr>
<tr>
<td><code>set -x</code></td>
<td>Print command traces before execution of each command (debug mode)</td>
</tr>
<tr>
<td><code>set -e</code></td>
<td>Exit the script if a command fails (recommended option)</td>
</tr>
<tr>
<td><code>set -u</code></td>
<td>Treat expansion of unset variables as an error</td>
</tr>
<tr>
<td><code>set -o nounset</code></td>
<td>Treat expansion of unset variables as an error</td>
</tr>
</tbody>
</table>

To run a script with a Bash option enabled, do one of the following:

- Run the script with `bash -option script.sh`
- Specify the shebang line in the script as `#!/bin/bash -option`
- Add the command `set -option` at the beginning of the script

Bash shell event

<table>
<thead>
<tr>
<th>Event</th>
<th>Files run</th>
</tr>
</thead>
</table>
| When a login shell is launched | `/etc/profile
/etc/profile.d/*.sh
~/.bash_profile
~/.bash_login
~/.profile` | The shell executes the system-wide profile files, then the first of the 3 user files that exists and is readable |
| When a login shell exits | `~/.bash_logout`                                                                                     |
| When a non-login shell is launched | `/etc/bash.bashrc
/etc/bashrc
~/.bashrc`                                                                                               |
Shell scripting

Bash shell scripts must start with the shebang line `#!/bin/bash` indicating the location of the script interpreter.

<table>
<thead>
<tr>
<th>Script execution</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>source script.sh</code></td>
<td>Script execution takes place in the same shell. Variables defined and exported in the script are seen by the shell when the script exits.</td>
</tr>
<tr>
<td><code>bash script.sh</code> (file must be executable)</td>
<td>Script execution spawns a new shell</td>
</tr>
</tbody>
</table>

- `command &` Execute `command` in the background
- `command1; command2` Execute `command 1` and then `command 2`
- `command1 && command2` Execute `command 2` only if `command 1` executed successfully (exit status = 0)
- `command1 || command2` Execute `command 2` only if `command 1` did not execute successfully (exit status > 0)
- `(command1 || command2)` Group commands together for evaluation priority
- Run `command` in a subshell. This is used to isolate `command`'s effects, as variable assignments and other changes to the shell environment operated by `command` will not remain after `command` completes

- `exit` Terminate a script
- `exit n` Terminate a script with the specified exit status number `n`. By convention, a 0 exit status is used if the script executed successfully, non-zero otherwise
- `/bin/true` Do nothing and return immediately a status code of 0 (indicating success)
- `/bin/false` Do nothing and return immediately a status code of 1 (indicating failure)
- `if command then echo "Success" else echo "Failure" fi` Run a command, then evaluate whether it exited successfully or failed

- `function myfunc { commands }` Define a function. A function must be defined before it can be used in a Bash script. An advantage of functions over aliases is that functions can be passed arguments
- `myfunc() { commands }` Call a function
- `typeset -f` Show functions defined in the current Bash session

- `getopts` Parse positional parameters in a shell script
- `expect` Dialogue with interactive programs according to a script, analyzing what can be expected from the interactive program and replying accordingly
- `zenity` Display GTK+ graphical dialogs for user messages and input
**Command execution**

- `watch command`: Execute `command` every 2 seconds.
- `watch -d -n 1 command`: Execute `command` every second, highlighting the differences in the output.
- `timeout 30s command`: Execute `command` and kill it after 30 seconds.
- `command | ts`: Prepend a timestamp to each line of the output of `command`.
- `sleep 5`: Pause for 5 seconds.
- `sleep ${($RANDOM % 60) + 1}s`: Sleep for a random time between 1 and 60 seconds.
- `sleep infinity`: Pause forever.
- `usleep 5000`: Pause for 5000 microseconds.
- `yes`: Output endlessly the string "y".
- `yes string`: Output endlessly `string`.
- `yes | fsck /dev/sda`: Automatically answer yes every time `fsck` asks for confirmation before fixing errors.
- `script file`: Generate a typescript of a terminal session. Forks a subshell and starts recording on `file` everything that is printed on terminal; the typescript ends when the user exits the subshell.
- `xargs command`: Call `command` multiple times, one for each argument found on stdin.
- `ls foo* | xargs cat`: Print via `cat` the content of every file whose name starts by "foo".
- `parallel command`: Run `command` in parallel. This is used to operate on multiple inputs, similarly to `xargs`.
Perform a test; if it results true, `command` is executed

test "$MYVAR" operator "value" && command
[ "$MYVAR" operator "value" ] && command
if [ "$MYVAR" operator "value" ]; then command; fi

<table>
<thead>
<tr>
<th>Test operators</th>
<th>File operators</th>
<th>Expression operators</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Integer operators</strong></td>
<td><strong>Equal to</strong></td>
<td>-e or -a Exists</td>
</tr>
<tr>
<td>-eq</td>
<td></td>
<td>-d Is a directory</td>
</tr>
<tr>
<td>-ne</td>
<td>Not equal to</td>
<td>-b Is a block special file</td>
</tr>
<tr>
<td>-lt</td>
<td>Less than</td>
<td>-c Is a character special file</td>
</tr>
<tr>
<td>-le</td>
<td>Less than or equal to</td>
<td>-f Is a regular file</td>
</tr>
<tr>
<td>-gt</td>
<td>Greater than</td>
<td>-r Is readable</td>
</tr>
<tr>
<td>-ge</td>
<td>Greater than or equal to</td>
<td>-w Is writable</td>
</tr>
</tbody>
</table>

**String operators**

| -z            | Is zero length |
| -n or nothing | Is non-zero length |
| = or ==      | Is equal to |
| !=           | Is not equal to |
| <            | Is alphabetically before |
| >            | Is alphabetically after |

**Evaluation operators**

| -            | Plus |
| +=           | Minus |
| *=           | Multiplied by |
| /=           | Divided by |
| %=           | Remainder |

```
expr "$MYVAR" = "39 + 3"
expr string : regex
expr string : \(regex\)
```

Evaluate an expression (in this case, assigns the value 42 to the variable)

Return the length of the substring matching the regex

Evaluate an expression (in this case, assigns the value 42 to the variable)

Return the substring matching the regex
## Flow control

### Tests

- **if** `test 1` then
  - `command block 1`
- **elif** `test 2` then
  - `command block 2`
- else
  - `command block 3`
- **fi**

- **case** `$STRING` in
  - `pattern1`
    - `command block 1`
  - `pattern2`
    - `command block 2`
  - `*`
    - `command block default`
- **esac**

### Loops

- **while** `[test]` do
  - `[command block]`
- **do**
- **done**

  The command block executes as long as test is true

- i=0
- while `[ $i -le 7 ]`
  - do
    - echo $i
    - let i++
  - done

- **until** `[test]`
  - do
    - `[command block]`
  - done

  The command block executes as long as test is false

- i=0
- until `[ $i -gt 7 ]`
  - do
    - echo $i
    - let i++
  - done

- **for** `item` in `[list]`
  - do
    - `[command block]`
  - done

  The command block executes for each item in list

- for `i` in 0 1 2 3 4 5 6 7
  - do
    - echo $i
  - done

- for `i` in `{0..7}`
  - do
    - echo $i
  - done

- start=0
- end=7
- for `i` in `$(seq $start $end)`
  - do
    - echo $i
  - done

- for ((i = start; i <= end; i++))
  - do
    - echo $i
  - done

- **break**
  - Exit a loop

- **continue**
  - Jump to the next iteration
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vi</td>
<td>Vi, text editor</td>
</tr>
<tr>
<td>vim</td>
<td>Vi Improved, an advanced text editor</td>
</tr>
<tr>
<td>gvim</td>
<td>Vim with GUI</td>
</tr>
<tr>
<td>vimdiff</td>
<td>Compare two text files in Vim</td>
</tr>
<tr>
<td>pico</td>
<td>Pico, simple text editor</td>
</tr>
<tr>
<td>nano</td>
<td>Nano, simple text editor (a GNU clone of Pico)</td>
</tr>
<tr>
<td>emacs</td>
<td>GNU Emacs, a GUI text editor</td>
</tr>
<tr>
<td>gedit</td>
<td>GUI text editor</td>
</tr>
<tr>
<td>ed</td>
<td>Line-oriented text editor</td>
</tr>
<tr>
<td>hexedit</td>
<td>Hexadecimal and ASCII editor</td>
</tr>
<tr>
<td>more</td>
<td>Text pager (obsolete)</td>
</tr>
<tr>
<td>less</td>
<td>Text pager</td>
</tr>
<tr>
<td>most</td>
<td>Text pager with advanced features (screen split, binary viewer, etc.)</td>
</tr>
<tr>
<td>Key</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>g</td>
<td>Go to the first line in the file</td>
</tr>
<tr>
<td>ng</td>
<td>Go to line number n</td>
</tr>
<tr>
<td>G</td>
<td>Go to the last line in the file</td>
</tr>
<tr>
<td>F</td>
<td>Go to the end of the file, and move forward automatically as the file grows</td>
</tr>
<tr>
<td>C</td>
<td>Stop moving forward</td>
</tr>
<tr>
<td>-N</td>
<td>Show line numbers</td>
</tr>
<tr>
<td>-n</td>
<td>Don't show line numbers</td>
</tr>
<tr>
<td>=</td>
<td>Show information about the file</td>
</tr>
<tr>
<td>l</td>
<td>Show current and total line number, byte, and percentage of the file read</td>
</tr>
<tr>
<td>/pattern</td>
<td>Search pattern forward</td>
</tr>
<tr>
<td>?pattern</td>
<td>Search pattern backwards</td>
</tr>
<tr>
<td>&amp;pattern</td>
<td>Display only lines matching pattern</td>
</tr>
<tr>
<td>n</td>
<td>Search next occurrences forward</td>
</tr>
<tr>
<td>N</td>
<td>Search next occurrences backwards</td>
</tr>
<tr>
<td>:n</td>
<td>When reading multiple files, go to the next file</td>
</tr>
<tr>
<td>:p</td>
<td>When reading multiple files, go to the previous file</td>
</tr>
<tr>
<td>R</td>
<td>Repaint the screen</td>
</tr>
<tr>
<td>V</td>
<td>Show version number</td>
</tr>
<tr>
<td>h</td>
<td>Help</td>
</tr>
<tr>
<td>q</td>
<td>Quit</td>
</tr>
</tbody>
</table>

**less +command file**

Open file for reading, applying command (see list above)

**less +F --follow-name file**

Move forward, attempting periodically to reopen file by name; useful to keep reading a logfile that is being rotated. Note that, by default, less continues to read the original input file even if it has been renamed.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ESC</code></td>
<td>Go to Command mode</td>
</tr>
<tr>
<td><code>i</code></td>
<td>Insert text before cursor and go to Insert mode</td>
</tr>
<tr>
<td><code>I</code></td>
<td>Insert text after line</td>
</tr>
<tr>
<td><code>a</code></td>
<td>Append text after cursor</td>
</tr>
<tr>
<td><code>A</code></td>
<td>Append text after line</td>
</tr>
<tr>
<td><code>v</code></td>
<td>Go to Visual mode, character-wise then use the arrow keys to select a block of text</td>
</tr>
<tr>
<td><code>v</code></td>
<td>Go to Visual mode, line-wise</td>
</tr>
<tr>
<td><code>d</code></td>
<td>Delete selected block</td>
</tr>
<tr>
<td><code>gU</code></td>
<td>Switch block to uppercase</td>
</tr>
<tr>
<td><code>y</code></td>
<td>Copy (yank) selected block into buffer</td>
</tr>
<tr>
<td><code>gu</code></td>
<td>Switch block to lowercase</td>
</tr>
<tr>
<td><code>w</code></td>
<td>Move to next word</td>
</tr>
<tr>
<td><code>s</code></td>
<td>Move to end of line</td>
</tr>
<tr>
<td><code>b</code></td>
<td>Move to beginning of word</td>
</tr>
<tr>
<td><code>1G</code></td>
<td>Move to line 1 i.e. beginning of file</td>
</tr>
<tr>
<td><code>e</code></td>
<td>Move to end of word</td>
</tr>
<tr>
<td><code>G</code></td>
<td>Move to end of file</td>
</tr>
<tr>
<td><code>0</code></td>
<td>Move to beginning of line</td>
</tr>
<tr>
<td><code>z</code></td>
<td>Make current line the top line of the screen</td>
</tr>
<tr>
<td><code>ma</code></td>
<td>Mark position &quot;a&quot;. Marks a-z are local to current file, while marks A-Z are global to a specific file</td>
</tr>
<tr>
<td><code>'a</code></td>
<td>Go to mark &quot;a&quot;. If using a global mark, it also opens the specific file</td>
</tr>
<tr>
<td><code>y'a</code></td>
<td>Copy (yank) from mark &quot;a&quot; to current line, into the buffer</td>
</tr>
<tr>
<td><code>d'a</code></td>
<td>Delete from mark &quot;a&quot; to current line</td>
</tr>
<tr>
<td><code>p</code></td>
<td>Paste buffer after current line</td>
</tr>
<tr>
<td><code>yy</code></td>
<td>Copy current line</td>
</tr>
<tr>
<td><code>P</code></td>
<td>Paste buffer before current line</td>
</tr>
<tr>
<td><code>YYP</code></td>
<td>Duplicate current line</td>
</tr>
<tr>
<td><code>x</code></td>
<td>Delete current character</td>
</tr>
<tr>
<td><code>D</code></td>
<td>Delete from current character to end of line</td>
</tr>
<tr>
<td><code>X</code></td>
<td>Delete before current character</td>
</tr>
<tr>
<td><code>dd</code></td>
<td>Delete current line</td>
</tr>
<tr>
<td><code>7dd</code></td>
<td>Delete 7 lines. Almost any command can be prepended by a number to repeat it that number of times</td>
</tr>
<tr>
<td><code>u</code></td>
<td>Undo last command. Vi can undo the last command only, Vim is able to undo several commands</td>
</tr>
<tr>
<td><code>.</code></td>
<td>Repeat last text-changing command</td>
</tr>
<tr>
<td><code>/string</code></td>
<td>Search for <code>string</code> forward</td>
</tr>
<tr>
<td><code>n</code></td>
<td>Search for next match of <code>string</code></td>
</tr>
<tr>
<td><code>?string</code></td>
<td>Search for <code>string</code> backwards</td>
</tr>
<tr>
<td><code>N</code></td>
<td>Search for previous match of <code>string</code></td>
</tr>
<tr>
<td><code>:s/s1/s2/</code></td>
<td>Replace the first occurrence of <code>s1</code> with <code>s2</code> in the current line</td>
</tr>
<tr>
<td><code>:s/s1/s2/g</code></td>
<td>Replace globally every occurrence of <code>s1</code> with <code>s2</code> in the current line</td>
</tr>
<tr>
<td><code>:%s/s1/s2/g</code></td>
<td>Replace globally every occurrence of <code>s1</code> with <code>s2</code> in the whole file</td>
</tr>
<tr>
<td><code>:%s/s1/s2/gc</code></td>
<td>Replace globally every occurrence of <code>s1</code> with <code>s2</code> in the whole file, asking for confirmation</td>
</tr>
<tr>
<td><code>:5,40s/^/#/</code></td>
<td>Add a hash character at the beginning of each line, from line 5 to 40</td>
</tr>
<tr>
<td><code>!!program</code></td>
<td>Replace line with output from <code>program</code></td>
</tr>
<tr>
<td><code>:r file</code></td>
<td>Read <code>file</code> and insert it after current line</td>
</tr>
<tr>
<td><code>:X</code></td>
<td>Encrypt current document. Vi will automatically prompt for the password to encrypt and decrypt</td>
</tr>
<tr>
<td><code>:w file</code></td>
<td>Write to <code>file</code></td>
</tr>
<tr>
<td><code>:wq</code></td>
<td>Save changes and quit</td>
</tr>
<tr>
<td><code>:x</code></td>
<td>Quit (fails if there are unsaved changes)</td>
</tr>
<tr>
<td><code>ZZ</code></td>
<td>Abandon all changes and quit</td>
</tr>
</tbody>
</table>

**vi -R file**  
Open `file` in read-only mode

**cat file | vi -**  
Open `file` in read-only mode (this is done by having Vi read from stdin)
<table>
<thead>
<tr>
<th>Option</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>ai</td>
<td>Turn on auto indentation</td>
</tr>
<tr>
<td>all</td>
<td>Display all options</td>
</tr>
<tr>
<td>ap</td>
<td>Print a line after the commands d c J m :s t u</td>
</tr>
<tr>
<td>aw</td>
<td>Automatic write on commands :n ! e # ^ :rew ^} :tag</td>
</tr>
<tr>
<td>bf</td>
<td>Discard control characters from input</td>
</tr>
<tr>
<td>dir=tmpdir</td>
<td>Set tmpdir as directory for temporary files</td>
</tr>
<tr>
<td>eb</td>
<td>Precede error messages with a bell</td>
</tr>
<tr>
<td>ht=8</td>
<td>Set terminal tab as 8 spaces</td>
</tr>
<tr>
<td>ic</td>
<td>Ignore case when searching</td>
</tr>
<tr>
<td>lisp</td>
<td>Modify brackets for Lisp compatibility</td>
</tr>
<tr>
<td>list</td>
<td>Show tabs and EOL characters</td>
</tr>
<tr>
<td>set listchars=tab:&gt;-&gt;</td>
<td>Show tab as &gt; for the first char and as – for the following chars</td>
</tr>
<tr>
<td>magic</td>
<td>Allow pattern matching with special characters</td>
</tr>
<tr>
<td>mesg</td>
<td>Enable UNIX terminal messaging</td>
</tr>
<tr>
<td>nu</td>
<td>Show line numbers</td>
</tr>
<tr>
<td>opt</td>
<td>Speed up output by eliminating automatic Return</td>
</tr>
<tr>
<td>para=LI1PLPPPQPbP</td>
<td>Set macro to start paragraphs for { } operators</td>
</tr>
<tr>
<td>prompt</td>
<td>Prompt : for command input</td>
</tr>
<tr>
<td>re</td>
<td>Simulate smart terminal on dumb terminal</td>
</tr>
<tr>
<td>remap</td>
<td>Accept macros within macros</td>
</tr>
<tr>
<td>report</td>
<td>Show the largest size of changes on status line</td>
</tr>
<tr>
<td>ro</td>
<td>Make file readonly</td>
</tr>
<tr>
<td>scroll=12</td>
<td>Set screen size as 12 lines</td>
</tr>
<tr>
<td>shell=/bin/bash</td>
<td>Set shell escape to /bin/bash</td>
</tr>
<tr>
<td>showmode</td>
<td>Show current mode on status line</td>
</tr>
<tr>
<td>slow</td>
<td>Postpone display updates during inserts</td>
</tr>
<tr>
<td>sm</td>
<td>Show matching parentheses when typing</td>
</tr>
<tr>
<td>sw=8</td>
<td>Set shift width to 8 characters</td>
</tr>
<tr>
<td>tags=/usr/lib/tags</td>
<td>Set path for files checked for tags</td>
</tr>
<tr>
<td>term</td>
<td>Print terminal type</td>
</tr>
<tr>
<td>terse</td>
<td>Print terse messages</td>
</tr>
<tr>
<td>timeout</td>
<td>Eliminate 1-second time limit for macros</td>
</tr>
<tr>
<td>tl=3</td>
<td>Set significance of tags beyond 3 characters (0 = all)</td>
</tr>
<tr>
<td>ts=8</td>
<td>Set tab stops to 8 for text input</td>
</tr>
<tr>
<td>wa</td>
<td>Inhibit normal checks before write commands</td>
</tr>
<tr>
<td>warn</td>
<td>Display the warning message &quot;No write since last change&quot;</td>
</tr>
<tr>
<td>window=24</td>
<td>Set text window as 24 lines</td>
</tr>
<tr>
<td>wm=0</td>
<td>Set automatic wraparound 0 spaces from right margin</td>
</tr>
</tbody>
</table>

Options can also be permanently set by including them in ~/.exrc (Vi) or ~/.vimrc (Vim)
SHOW DATABASES;
USE CompanyDatabase;
SELECT DATABASE();
DROP DATABASE CompanyDatabase;

SHOW TABLES;
CREATE TABLE customers (
cusid INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(32), lastname VARCHAR(32), dob DATE,
city VARCHAR(24), zipcode VARCHAR(5));
CREATE TABLE payments (
payid INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
date DATE, fee INT, bill VARCHAR(128), cusid INT,
CONSTRAINT FK1 FOREIGN KEY (cusid) REFERENCES customers(cusid));
INSERT INTO customers (firstname,lastname,dob)
VALUES ('Arthur','Dent',1959-08-01), ('Trillian','',1971-03-19);
DELETE FROM customers WHERE firstname LIKE 'Zaphod';
UPDATE customers SET city = 'London' WHERE zipcode = 'L1 42HG';
CREATE INDEX lastname_index ON customers(lastname);
ALTER TABLE customers ADD INDEX lastname_index (lastname);

DESCRIBE customers;
SHOW CREATE TABLE customers;
SHOW INDEXES FROM customers;
DROP TABLE customers;
ALTER TABLE customers MODIFY city VARCHAR(32);
CREATE VIEW cust_view AS
SELECT * FROM customers WHERE city != 'London';

COMMIT;
ROLLBACK;
START TRANSACTION;
BEGIN;

If no database has been selected for use, tables must be referenced by `databasename.tablename`. 
`SELECT * FROM customers;

SELECT firstname, lastname FROM customers LIMIT 5;

SELECT firstname, lastname FROM customers LIMIT 1000,5;
SELECT firstname, lastname FROM customers OFFSET 1000 LIMIT 5;

SELECT firstname, lastname FROM customers WHERE zipcode = 'L1 42HG';
SELECT firstname, lastname FROM customers WHERE zipcode IS NOT NULL;
SELECT * FROM customers ORDER BY lastname, firstname;
SELECT * FROM customers ORDER by zipcode DESC;

SELECT firstname, lastname, TIMESTAMPDIFF(YEAR,dob,CURRENT_DATE) AS age FROM customers;
SELECT city, COUNT(*) FROM customers GROUP BY city;

SELECT cusid, SUM(fee) FROM payments GROUP BY cusid;
SELECT cusid, AVG(fee) FROM payments GROUP BY cusid
HAVING AVG(fee)<50;

SELECT MAX(fee) FROM payments;
SELECT COUNT(*) FROM customers;
SELECT cusid FROM payments t1 WHERE fee = (SELECT MAX(t2.fee) FROM payments t2 WHERE t1.cusid=t2.cusid);
SELECT @maxfee:=MAX(fee) FROM payments;
SELECT cusid FROM payments t1 WHERE fee = @maxfee;

SELECT * FROM customers WHERE firstname LIKE 'Trill%';
SELECT * FROM customers WHERE firstname REGEXP '^Art.*r$';

SELECT firstname, lastname FROM customers WHERE zipcode = 'L1 42HG' UNION
SELECT firstname, lastname FROM customers WHERE cusid > 4242001;
SELECT firstname, lastname FROM customers WHERE zipcode = 'L1 42HG'
INTERSECT
SELECT firstname, lastname FROM customers WHERE cusid > 4242001;
SELECT firstname, lastname FROM customers WHERE zipcode = 'L1 42HG'
EXCEPT
SELECT firstname, lastname FROM customers WHERE cusid > 4242001;

Select all columns from the customers table
Select first and last name of customers, showing 5 records only
Select first and last name of customers, skipping the first 1000 records and showing 5 records only
Select first and last name of customers whose zip code is "L1 42HG"
Select first and last name of customers with an existing zip code
Select customers in alphabetical order by last name, then first name
Select customers, sorting them by zip code in reverse order
Select first name, last name, and calculated age of customers
Show all cities, retrieving each unique output record only once
Show all cities and the number of customers in each city. NULL values are not counted
Show all fee payments grouped by customer ID, summed up
Show the average of fee payments grouped by customer ID, where this average is less than 50
Show the highest fee in the table
Show how many rows are in the table
Show the customer ID that pays the highest fee (via a subquery)
Show the customer ID that pays the highest fee (via a user set variable)
Show the customers which have same last name as other customers
Show the customer IDs that pay fees higher than the highest fee paid by customer ID 4242001

Select customers whose first name matches the expression:
% any number of chars, even zero
_ a single char
Select customers whose first name matches the regex

Select customers that satisfy any of the two requirements
Select customers that satisfy both of the two requirements
Select customers that satisfy the first requirement but not the second
<table>
<thead>
<tr>
<th>SQL</th>
<th>MySQL</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT customers.name, payments.bill FROM customers, payments WHERE customers.cusid = payments.cusid;</td>
<td>SELECT customers.name, payments.bill FROM customers [ JOIN</td>
<td>INNER JOIN</td>
</tr>
<tr>
<td>SELECT customers.name, payments.bill FROM customers NATURAL JOIN payments;</td>
<td>SELECT customers.name, payments.bill FROM customers [ JOIN</td>
<td>INNER JOIN</td>
</tr>
<tr>
<td>SELECT customers.name, payments.bill FROM customers JOIN payments USING (cusid);</td>
<td>SELECT customers.name, payments.bill FROM customers [ JOIN</td>
<td>INNER JOIN</td>
</tr>
<tr>
<td>SELECT customers.name, payments.bill FROM customers CROSS JOIN payments;</td>
<td>SELECT customers.name, payments.bill FROM customers JOIN payments;</td>
<td>Perform a cross join (aka Cartesian product) of two tables</td>
</tr>
<tr>
<td>SELECT customers.name, payments.bill FROM customers LEFT JOIN payments ON customers.cusid = payments.cusid;</td>
<td></td>
<td>Perform a left join (aka left outer join) of two tables, returning records matching the join condition and also records in the left table with unmatched values in the right table</td>
</tr>
<tr>
<td>SELECT customers.name, payments.bill FROM customers RIGHT JOIN payments ON customers.cusid = payments.cusid;</td>
<td></td>
<td>Perform a right join (aka right outer join) of two tables, returning records matching the join condition and also records in the right table with unmatched values in the left table</td>
</tr>
</tbody>
</table>
MySQL is the most used open source RDBMS (Relational Database Management System). It runs on TCP port 3306. On RHEL 7 it is replaced by its fork MariaDB, but the names of the client and of most tools remain unchanged.

**mysqld_safe**

Start the MySQL server (mysqld) with safety features such as restarting the server if errors occur and logging runtime information to the error logfile. This is the recommended command.

**mysql_install_db (deprecated)**

Initialize the MySQL data directory, create system tables, and set up an administrative account. To be run just after installing the MySQL server.

**mysql_secure_installation**

Set password for root, remove anonymous users, disable remote root login, and remove test database. To be run just after installing the MySQL server.

**mysql -u root -p**

Login to MySQL as root and prompt for the password.

**mysql -u root -p**

Login to MySQL as root with the specified password.

**mysql -u root -p -h host -P port**

Login to the specified remote MySQL host and port.

**mysql -u root -p -B 'SHOW DATABASES'**

Run a SQL command via MySQL. Flags are:
- **e** Run in batch mode
- **N** Do not print table header
- **B** Do not print table decoration characters

**mysqldump -u root -p --all-databases > dump.sql**

Backup all databases to a dump file.

**mysqldump -u root -p db > dump.sql**

Backup a database to a dump file.

**mysqldump -u root -p --databases db1 db2 > dump.sql**

Backup multiple databases to a dump file.

**mysqldump -u root -p db table1 table2 > dump.sql**

Backup some tables of a database to a dump file.

**mysql -u root -p < dump.sql**

Restore all databases from a dump file (which contains a complete dump of a MySQL server).

**mysql -u root -p db < dump.sql**

Restore a specific database from a dump file (which contains one database).

**mysql_upgrade -u root -p**

Check all tables in all databases for incompatibilities with the current version of MySQL.

**mysqlcheck**

Perform table maintenance. Each table is locked while is being processed. Options are:
- **--check** Check table for errors (default)
- **--analyze** Analyze table
- **--optimize** Optimize table
- **--repair** Repair table; can fix almost all problems except unique keys that are not unique

**mysqlcheck --check db table**

Check the specified table of the specified database.

**mysqlcheck --check --databases db1 db2**

Check the specified databases.

**mysqlcheck --check --all-databases**

Check all databases.
MySQL tools

mysqlslap  Tool for MySQL stress tests
mysqltuner.pl  Review the current MySQL installation configuration for performances and stability
mysqlreport  (obsolete)  Generate a user-friendly report of MySQL status values
mytop  Monitor MySQL processes and queries
innotop  Monitor MySQL InnoDB transactions

dbs="$(mysql -uroot -p$mysql_root_password -Bse'SHOW DATABASES;')"
for db in $dbs
do
  [operation on $db]
done

Perform an operation on each database name.
MySQL syntax

SELECT Host, User FROM mysql.user;

CREATE USER 'user'@'localhost' IDENTIFIED BY 'p4ssw0rd';

DROP USER 'user'@'localhost';

SET PASSWORD FOR 'user'@'localhost' = PASSWORD('p4ssw0rd');
SET PASSWORD FOR 'user'@'localhost' = '*7E684A3DF6273CD1B6DE53';

SHOW GRANTS FOR 'user'@'localhost';
GRANT ALL PRIVILEGES ON database.* TO 'user'@'localhost';
REVOKE ALL PRIVILEGES ON database.* FROM 'user'@'localhost';

GRANT SELECT ON database.* TO 'john'@'localhost' IDENTIFIED BY 'p4ssw0rd';
GRANT SELECT ON database.* TO 'john'@'localhost' IDENTIFIED BY PASSWORD '*7E684A3DF6273CD1B6DE53';

FLUSH PRIVILEGES;

SELECT * INTO OUTFILE 'file.csv'
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '\n' FROM database.table;

USE database; SOURCE dump.sql;

USE database; LOAD DATA LOCAL INFILE 'file.csv' INTO TABLE table;

DO SLEEP(n);
SELECT SLEEP(n);

SET PROFILING=1;
SHOW PROFILE;

statement;
statement\g
statement\G

SELECT /*!99999 comment*/ * FROM database.table;
SELECT /*!v statement*/ * FROM database.table;

\c

%! command

TEE logfile
SHOW VARIABLES;
SHOW SESSION VARIABLES;
SHOW LOCAL VARIABLES;
SHOW GLOBAL VARIABLES;
SHOW VARIABLES LIKE '%query%';
SHOW VARIABLES LIKE 'hostname';
SELECT @@hostname;

SET sort_buffer_size=10000;
SET SESSION sort_buffer_size=10000;
SET LOCAL sort_buffer_size=10000;
SET @@session.sort_buffer_size=10000;
SET @@local.sort_buffer_size=10000;
SET GLOBAL sort_buffer_size=10000;
SET @@global.sort_buffer_size=10000;

SHOW STATUS;
SHOW SESSION STATUS;
SHOW LOCAL STATUS;
SHOW GLOBAL STATUS;
SHOW STATUS LIKE '%wsrep%';

SHOW WARNINGS;
SHOW ERRORS;

SHOW TABLE STATUS;
SHOW ENGINE INNODB STATUS;
SELECT * FROM information_schema.processlist;
SHOW FULL PROCESSLIST;
SELECT * FROM information_schema.processlist
WHERE user='you';

SHOW CREATE TABLE table;
SHOW CREATE VIEW view;

SELECT VERSION();
SELECT CURDATE();
SELECT CURRENT_DATE;
SELECT CURTIME();
SELECT CURRENT_TIME;
SELECT NOW();
SELECT USER();

\s
SELECT table_schema AS "Name",
SUM(data_length+index_length)/1024/1024 AS "Size in Mb"
FROM information_schema.tables GROUP BY table_schema;

Display the sizes of all databases in the system (counting data + indexes)

SELECT table_schema AS "Name",
SUM(data_length+index_length)/1024/1024 AS "Size in Mb"
FROM information_schema.tables WHERE table_schema='database';

Display the size of database

SELECT table_name AS "Name",
ROUND(((data_length)/1024/1024),2) AS "Data size in Mb",
ROUND(((index_length)/1024/1024),2) AS "Index size in Mb"
FROM information_schema.TABLES WHERE table_schema='database'
ORDER BY table_name;

Display data and index size of all tables of database

SELECT table_name, table_rows
FROM information_schema.tables WHERE table_schema='database';

Print an estimate of the number of rows of each table of database

SELECT SUM(data_length+index_length)/1024/1024 AS "InnoDB Mb"
FROM information_schema.tables WHERE engine='InnoDB';

Display the amount of InnoDB data in all databases

SELECT table_name, engine
FROM information_schema.tables WHERE table_schema = 'database';

Print name and engine of all tables in database

SELECT CONCAT('KILL ',id,';')
FROM information_schema.processlist WHERE user='user'
INTO OUTFILE '/tmp/killuser'; SOURCE /tmp/killuser;

Kill all connections belonging to user

SELECT COUNT(1) SlaveThreadCount
FROM information_schema.processlist WHERE user='system user';

Distinguish between master and slave server; returns 0 on a master, >0 on a slave

SELECT ROUND(SUM(CHAR_LENGTH(field)<40)*100/COUNT(*),2)
FROM table;

Display the percentage of rows on which the string field is shorter than 40 chars

SELECT CHAR_LENGTH(field) AS Length, COUNT(*) AS Occurrences
FROM table GROUP BY CHAR_LENGTH(field);

Display all different lengths of string field and the number of times they occur

SELECT MAX(CHAR_LENGTH(field)) FROM table;

Display the longest string stored in field

SHOW FULL TABLES IN database WHERE table_type LIKE 'VIEW';

Display the list of views in database

SELECT "Table 1" AS `set`, t1.* FROM table1 t1 WHERE
ROW(t1.col1, t1.col2, t1.col3) NOT IN (SELECT * FROM table2)
UNION ALL
SELECT "Table 2" AS `set`, t2.* FROM table2 t2 WHERE
ROW(t2.col1, t2.col2, t2.col3) NOT IN (SELECT * FROM table1)

Display the differences between the contents of two tables table1 and table2 (assuming the tables are composed of 3 columns each)
MySQL operations

How to resync a master-slave replication

1. On the master, on terminal 1:
   
   ```
   mysql -uroot -p
   RESET MASTER;
   FLUSH TABLES WITH READ LOCK;
   SHOW MASTER STATUS;
   ```

   Note the values of MASTER_LOG_FILE and MASTER_LOG_POS; these values will need to be copied on the slave.

2. On the master, on terminal 2:
   
   ```
   mysqldump -uroot -p --all-databases > /root/dump.sql
   ```

   It is not necessary to wait until the dump completes.

3. On the master, on terminal 1:
   
   ```
   UNLOCK TABLES;
   ```

4. Transfer the dump file from the master to the slave

5. On the slave:
   
   ```
   mysql -uroot -p
   STOP SLAVE;
   SOURCE /root/dump.sql;
   RESET SLAVE;
   CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin.nnnnnn', MASTER_LOG_POS=mm;
   START SLAVE;
   SHOW SLAVE STATUS;
   ```

How to recover the MySQL root password

1. Stop the MySQL server

2. Restart the MySQL server skipping the grant tables
   
   ```
   mysqld_safe --skip-grant-tables --skip-networking &
   ```

3. Connect to the MySQL server passwordlessly
   
   ```
   mysql -uroot
   ```

4. Reload the grant tables
   
   ```
   FLUSH PRIVILEGES;
   ```

5. Change the root password
   
   ```
   SET PASSWORD FOR 'root'@'localhost' = PASSWORD('s3cr3t');
   ```

6. Stop the MySQL server and restart it normally
PostgreSQL (aka Postgres) is an open source object-relational database. By default it listens for connections on TCP port 5432.

```
\list
\l
\list+
\l+
\connect database
\c database
\q
```

List all databases
List all databases, displaying database size and description
Connect to database
Quit

**How to set up PostgreSQL with a database owned by user**

1. Set up PostgreSQL
   `postgresql-setup initdb`
2. Change the password of the postgres shell user
   `passwd postgres`
3. Create the `user` shell user
   `useradd user`
4. Switch to the postgres shell user and connect to PostgreSQL
   `su - postgres
   psql -U postgres`
5. Create the `user` PostgreSQL user
   `CREATE ROLE user WITH LOGIN;
   \password user
   \q`
6. Create a database owned by `user`
   `createdb -E utf8 -l C -T template0 database -O user`
7. Switch to the postgres shell user and connect to PostgreSQL
   `su - postgres
   psql -U postgres`
8. Grant the necessary privileges
   `GRANT ALL PRIVILEGES ON DATABASE database TO user;
   \q`
9. Verify that `user` can login to PostgreSQL
   `su - user
   psql -U user -W`
The X Window System (aka X11 or X) is a windowing system for Linux and UNIX-like OSes, providing a basic framework for GUI applications via a client-server model. A display manager provides a login screen to enter an X session and introduces the user to the desktop environment (e.g. GNOME, KDE, CDE, Enlightenment).

<table>
<thead>
<tr>
<th>Display Manager</th>
<th>Configuration files</th>
<th>Display Manager greeting screen</th>
</tr>
</thead>
<tbody>
<tr>
<td>xdm X Display Manager</td>
<td>/etc/x11/xdm/Xaccess</td>
<td>Control inbound requests from remote hosts</td>
</tr>
<tr>
<td></td>
<td>/etc/x11/xdm/Xresources</td>
<td>Configuration settings for X applications and the login screen</td>
</tr>
<tr>
<td></td>
<td>/etc/x11/xdm/Xservers</td>
<td>Association of X displays with local X server software, or with X terminals via XDMCP</td>
</tr>
<tr>
<td></td>
<td>/etc/x11/xdm/Xsession</td>
<td>Script launched by xdm after login</td>
</tr>
<tr>
<td></td>
<td>/etc/x11/xdm/Xsetup_0</td>
<td>Script launched before the graphical login screen</td>
</tr>
<tr>
<td></td>
<td>/etc/x11/xdm/xdm-config</td>
<td>Association of all xdm configuration files</td>
</tr>
<tr>
<td>gdm GNOME Display Manager</td>
<td>/etc/gdm/gdm.conf or /etc/gdm/custom.conf</td>
<td>Configured via gdmsetup</td>
</tr>
<tr>
<td>kdm KDE Display Manager</td>
<td>/etc/kde/kdm/kdmrc</td>
<td>Configured via kdm_config</td>
</tr>
</tbody>
</table>

/etc/init.d/xdm start
/etc/init.d/gdm start
/etc/init.d/kdm start

Start the appropriate Display Manager

xorgconfig (Debian)
Xorg -configure (Red Hat)
xorgcfg (Debian)
system-config-display (Red Hat)
x -version
xdpyinfo
xwininfo
xhost + 10.3.3.3
xhost - 10.3.3.3
switchdesk gde
gnome-shell --version

Configure X (text mode)
Configure X (graphical mode)
Show which version of X is running
Display information about the X server
Display information about windows
Add or remove 10.3.3.3 to the list of hosts allowed to make X connections to the local machine
Switch to the GDE Display Manager at runtime
Show which version of GNOME is running

/etc/X11/xorg.conf
~/.Xresources
$DISPLAY

Configuration file for X
Configuration settings for X applications, in the form
program*resource: value

Environment variable defining the display name of the X server, in the form
hostname:displaynumber.screennumber

The following line in /etc/inittab instructs init to launch XDM at runlevel 5:

x:5:respawn:/usr/X11R6/bin/xdm -nodaemon

The following lines in /etc/sysconfig/desktop define GNOME as the default Display Environment and Display Manager:
desktop="gde"
displaymanager="gdm"
xdotool
    X automation tool

xdotool getwindowfocus
    Get the ID of the currently focused window (if run in command line, it is
    the terminal where this command is typed)

xdotool selectwindow
    Pop up an X cursor and get the ID of the window selected by it

xdotool key --window 12345678 Return
    Simulate a \[\text{RETURN}\] keystroke inside window ID 12345678

xprop
    X property displayer. Pops up a cursor to select a window

xprop | grep WM_CLASS
    Get process name and GUI application name of the selected window

xrandr
    Xrandr -q
    Show screen(s) size and resolution

xrandr --output eDP1 --right-of VGA1
    Extend the screen on an additional VGA physical screen situated to the left

xsel
    Manipulate the X selection (primary, secondary, and clipboard)

xsel -b < file
    Copy the contents of a file to the X clipboard

xsel -b -a < file
    Append the contents of a file to the X clipboard

xsel -b -o
    Output onscreen the contents of the X clipboard

    Copy the contents of a file to the X clipboard

    Catalog the newly installed fonts in the new directory

mkfontdir
    Dynamically add new installed fonts in /usr/local/fonts to the X server

xset fp+ /usr/local/fonts
    Start the X font server

xfs
    Install fonts and build font information cache
### X keysim codes

#### Main

<table>
<thead>
<tr>
<th>Key</th>
<th>Main</th>
<th>Latin 1</th>
<th>Latin 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>BackSpace</td>
<td>f608</td>
<td>space 0020</td>
<td>questiondown 00bf</td>
</tr>
<tr>
<td>Tab</td>
<td>f609</td>
<td>exclam 0021</td>
<td>Agrave 00c0</td>
</tr>
<tr>
<td>Linefeed</td>
<td>f60a</td>
<td>quotedbl 0022</td>
<td>Acute 00c1</td>
</tr>
<tr>
<td>Clear</td>
<td>f60b</td>
<td>numbersign 0023</td>
<td>Acircumflex 00c2</td>
</tr>
<tr>
<td>Return</td>
<td>f60d</td>
<td>dollar 0024</td>
<td>Atilde 00c3</td>
</tr>
<tr>
<td>Pause</td>
<td>f613</td>
<td>percent 0025</td>
<td>Adieresis 00c4</td>
</tr>
<tr>
<td>Scroll_Lock</td>
<td>f614</td>
<td>amperandslash 0026</td>
<td>Aring 00c5</td>
</tr>
<tr>
<td>Sys_Req</td>
<td>f615</td>
<td>apostrophe 0027</td>
<td>AE 00c6</td>
</tr>
<tr>
<td>Escape</td>
<td>f61b</td>
<td>quoteright 0027</td>
<td>Ccedilla 00c7</td>
</tr>
<tr>
<td>Delete</td>
<td>fff</td>
<td>parenleft 0028</td>
<td>Egrave 00c8</td>
</tr>
</tbody>
</table>

#### Cursor control

<table>
<thead>
<tr>
<th>Key</th>
<th>Main</th>
<th>Latin 1</th>
<th>Latin 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td>f55</td>
<td>plus 002b</td>
<td>Edieresis 00cb</td>
</tr>
<tr>
<td>Left</td>
<td>f51</td>
<td>comma 002c</td>
<td>Igrave 00cc</td>
</tr>
<tr>
<td>Up</td>
<td>f52</td>
<td>minus 002d</td>
<td>Lacute 00cd</td>
</tr>
<tr>
<td>Right</td>
<td>f53</td>
<td>period 002e</td>
<td>Icircumflex 00cf</td>
</tr>
<tr>
<td>Down</td>
<td>f54</td>
<td>slash 002f</td>
<td>Idieresis 00cf</td>
</tr>
<tr>
<td>Prior</td>
<td>f55</td>
<td>0 - 9 0030 - 0039</td>
<td>ETH 00d0</td>
</tr>
<tr>
<td>Page_Up</td>
<td>f55</td>
<td>colon 003a</td>
<td>Eth 00d0</td>
</tr>
<tr>
<td>Next</td>
<td>f56</td>
<td>semicolon 003b</td>
<td>Ndieresis 00d1</td>
</tr>
<tr>
<td>Page_Down</td>
<td>f56</td>
<td>less 003c</td>
<td>Ograve 00d6</td>
</tr>
<tr>
<td>End</td>
<td>f57</td>
<td>equal 003d</td>
<td>Oacute 00d3</td>
</tr>
<tr>
<td>Begin</td>
<td>f58</td>
<td>greater 003e</td>
<td>Ocircumflex 00d4</td>
</tr>
</tbody>
</table>

#### Misc functions

<table>
<thead>
<tr>
<th>Key</th>
<th>Main</th>
<th>Latin 1</th>
<th>Latin 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select</td>
<td>f60</td>
<td>A - 2 0041 - 005a</td>
<td>multiply 00d7</td>
</tr>
<tr>
<td>Print</td>
<td>f61</td>
<td>bracketleft 005b</td>
<td>Oslash 00d8</td>
</tr>
<tr>
<td>Execute</td>
<td>f62</td>
<td>backslash 005c</td>
<td>Ooblique 00d8</td>
</tr>
<tr>
<td>Insert</td>
<td>f63</td>
<td>bracketright 005d</td>
<td>Ugrave 00d9</td>
</tr>
<tr>
<td>Undo</td>
<td>f65</td>
<td>aslasicircum 005e</td>
<td>Uacute 00da</td>
</tr>
<tr>
<td>Redo</td>
<td>f66</td>
<td>underscore 005f</td>
<td>Ucircumflex 00db</td>
</tr>
<tr>
<td>Menu</td>
<td>f67</td>
<td>grave 0060</td>
<td>Udieresis 00dc</td>
</tr>
<tr>
<td>Find</td>
<td>f68</td>
<td>a - x 0061 - 007a</td>
<td>Ythorn 00de</td>
</tr>
<tr>
<td>Cancel</td>
<td>f69</td>
<td>braceleft 007b</td>
<td>Yegrave 00de</td>
</tr>
<tr>
<td>Help</td>
<td>f6a</td>
<td>bar 007c</td>
<td>Yacute 00d4</td>
</tr>
<tr>
<td>Break</td>
<td>f6b</td>
<td>braceright 007d</td>
<td>Ygrave 00d5</td>
</tr>
<tr>
<td>Mode_switch</td>
<td>f7e</td>
<td>asciscitalite 007e</td>
<td>Yacute 00e1</td>
</tr>
<tr>
<td>script_switch</td>
<td>f7e</td>
<td>nobreakspace 00a0</td>
<td>Ycircumflex 00be</td>
</tr>
<tr>
<td>Num_Lock</td>
<td>fff</td>
<td>exclamdown 00a1</td>
<td>Ytildescript 00d3</td>
</tr>
<tr>
<td>Shift_L</td>
<td>ffe1</td>
<td>cent 00a2</td>
<td>Yaelig 00d4</td>
</tr>
<tr>
<td>Shift_R</td>
<td>ffe2</td>
<td>sterling 00a3</td>
<td>Ycircumflex 00e0</td>
</tr>
<tr>
<td>Control_L</td>
<td>ffe3</td>
<td>currency 00a4</td>
<td>Yaelig 00ea</td>
</tr>
<tr>
<td>Control_R</td>
<td>ffe3</td>
<td>yen 00a5</td>
<td>Ycircumflex 00ec</td>
</tr>
<tr>
<td>Caps_Lock</td>
<td>ffe5</td>
<td>brokenbar 00a6</td>
<td>Yaelig 00f0</td>
</tr>
<tr>
<td>Shift_Lock</td>
<td>ffe6</td>
<td>section 00a7</td>
<td>Ycircumflex 00f0</td>
</tr>
<tr>
<td>Meta_L</td>
<td>ffe7</td>
<td>diacritical 00a8</td>
<td>Ycircumflex 00f1</td>
</tr>
<tr>
<td>Meta_R</td>
<td>ffe8</td>
<td>copyright 00a9</td>
<td>Ycircumflex 00f2</td>
</tr>
<tr>
<td>Alt_L</td>
<td>ffe9</td>
<td>copyright 00a9</td>
<td>Ycircumflex 00f3</td>
</tr>
<tr>
<td>Alt_R</td>
<td>ffea</td>
<td>registered 00ae</td>
<td>Ycircumflex 00f4</td>
</tr>
<tr>
<td>Super_L</td>
<td>ffeb</td>
<td>macron 00af</td>
<td>Ycircumflex 00f5</td>
</tr>
<tr>
<td>Super_R</td>
<td>ffec</td>
<td>degree 00b0</td>
<td>Ycircumflex 00f6</td>
</tr>
<tr>
<td>Hyper_L</td>
<td>ffed</td>
<td>plusminus 00b1</td>
<td>Ycircumflex 00f7</td>
</tr>
<tr>
<td>Hyper_R</td>
<td>ffee</td>
<td>twosuperior 00b2</td>
<td>Ycircumflex 00f8</td>
</tr>
</tbody>
</table>

#### Modifiers

<table>
<thead>
<tr>
<th>Key</th>
<th>Main</th>
<th>Latin 1</th>
<th>Latin 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caps_Lock</td>
<td>ffe5</td>
<td>plus 00b2</td>
<td>Ycircumflex 00f9</td>
</tr>
<tr>
<td>Shift_Lock</td>
<td>ffe6</td>
<td>minus 00b3</td>
<td>Ycircumflex 00fa</td>
</tr>
<tr>
<td>Meta_L</td>
<td>ffe7</td>
<td>acute 00b4</td>
<td>Ycircumflex 00fb</td>
</tr>
<tr>
<td>Meta_R</td>
<td>ffe8</td>
<td>acute 00b5</td>
<td>Ycircumflex 00fc</td>
</tr>
<tr>
<td>Alt_L</td>
<td>ffe9</td>
<td>acute 00b6</td>
<td>Ycircumflex 00fd</td>
</tr>
<tr>
<td>Alt_R</td>
<td>ffea</td>
<td>acute 00b7</td>
<td>Ycircumflex 00fe</td>
</tr>
<tr>
<td>Super_L</td>
<td>ffeb</td>
<td>acute 00b8</td>
<td>Ycircumflex 00ff</td>
</tr>
<tr>
<td>Super_R</td>
<td>ffec</td>
<td>acute 00b9</td>
<td>Ycircumflex 00f0</td>
</tr>
<tr>
<td>Hyper_L</td>
<td>ffed</td>
<td>acute 00ba</td>
<td>Ycircumflex 00f1</td>
</tr>
<tr>
<td>Hyper_R</td>
<td>ffee</td>
<td>acute 00bb</td>
<td>Ycircumflex 00f2</td>
</tr>
</tbody>
</table>

### This table is derived from keysymdef.h which defines keysym codes (i.e. characters or functions associated with each key in the X Window System) as XK_key and its hex value. The key can be passed as argument to the xdotoool key command.
### User accounts

<table>
<thead>
<tr>
<th>Login name</th>
<th>Hashed password (obsolete), or x if password is in /etc/shadow</th>
<th>UID – User ID</th>
<th>GID – Default Group ID</th>
<th>GECOS field – Information about the user: Full name, Room number, Work phone, Home phone, Other</th>
<th>Home directory of the user</th>
<th>Login shell (if set to /sbin/nologin or /bin/false, user will be unable to log in)</th>
</tr>
</thead>
<tbody>
<tr>
<td>root</td>
<td>:x:0:0: /root:/bin/bash</td>
<td>0</td>
<td>0</td>
<td>/root</td>
<td>/bin/bash</td>
<td></td>
</tr>
<tr>
<td>bin</td>
<td>:x:1:1: /bin:/bin/bash</td>
<td>1</td>
<td>1</td>
<td>/bin</td>
<td>/bin/bash</td>
<td></td>
</tr>
<tr>
<td>jdoe</td>
<td>:x:500:10: John Doe,,555-1234,,:/home/jdoe:/bin/bash</td>
<td>500</td>
<td>10</td>
<td>/home/jdoe</td>
<td>/bin/bash</td>
<td></td>
</tr>
</tbody>
</table>

### User passwords

<table>
<thead>
<tr>
<th>Login name</th>
<th>Hashed password (if account is disabled, ! or !! if no password is set, prefixed by ! if the account is locked). Composed of the following subfields separated by $:</th>
<th>Date of last password change (in number of days since 1 January 1970)</th>
<th>Days before password may be changed; if 0, user can change the password at any time</th>
<th>Days after which password must be changed</th>
<th>Days before password expiration that user is warned</th>
<th>Days after password expiration that account is disabled</th>
<th>Date of account disabling (in number of days since 1 January 1970)</th>
<th>Reserved field</th>
</tr>
</thead>
<tbody>
<tr>
<td>bin</td>
<td>$6$yiH1otQ$KxeEUKHEK8e3jCUdw9Rxy3wu53:15580:0:99999:7:::15766:</td>
<td>15637:0:99999:7:::</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>jdoe</td>
<td>!$6$YOiH1otQ$KxeEUKHEK8e3jCUdw9Rxy3wu53:15580:0:99999:7:::15766:</td>
<td>15580:0:99999:7:::15766:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Group accounts

<table>
<thead>
<tr>
<th>Group name</th>
<th>Encrypted password, or x if password is in /etc/gshadow</th>
<th>GID – Group ID</th>
<th>Group members (if this is not their Default Group)</th>
</tr>
</thead>
<tbody>
<tr>
<td>root</td>
<td>:x:0:root</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>jdoe</td>
<td>:x:530:jdoe, asmith</td>
<td>530</td>
<td>jdoe, asmith</td>
</tr>
</tbody>
</table>

### Group passwords

<table>
<thead>
<tr>
<th>Group name</th>
<th>Encrypted password, or ! if no password set (default)</th>
<th>Group administrators</th>
<th>Group members</th>
</tr>
</thead>
<tbody>
<tr>
<td>root</td>
<td>:x:root:root</td>
<td></td>
<td></td>
</tr>
<tr>
<td>jdoe</td>
<td>!::</td>
<td></td>
<td></td>
</tr>
<tr>
<td>staff</td>
<td>$0cfz7IplhW19i::root:jdoe</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

/`/etc/shadow` and `/etc/gshadow` are mode 000 and therefore readable only by the root user.
User management

**useradd** 
- **-m** jdoe: Create a user account, creating and populating his homedir from /etc/skel
- **-mc** "John Doe" jdoe: Create a user account, specifying his full name
- **-ms** /bin/ksh jdoe: Create a user account, specifying his login shell
- **-D**: Show default values for user account creation, as specified in /etc/login.defs and /etc/default/useradd

**usermod** 
- **-c** "Jason Doe" jdoe: Modify the GECOS field of a user account
- **-L** jdoe: Lock a user account
- **-U** jdoe: Unlock a user account

Most options for **usermod** and **useradd** are the same.

**userdel** 
- **-r** jdoe: Delete a user and his homedir

**chfn** jdoe: Change the GECOS field of a user

**chsh** jdoe: Change the login shell of a user

**passwd** jdoe: Change the password of a user
- **-l** jdoe: Lock a user account
- **-S** jdoe: Show information about a user account: username, account status (L=locked, P=password, NP=no password), date of last password change, min age, max age, warning period, inactivity period in days

**chage** 
- **-E** 2022-02-14 jdoe: Change the password expiration date; account will be locked at that date
- **-d** 13111 jdoe: Change the date (in number of days since 1 January 1970) of last password change
- **-d** 0 jdoe: Force the user to change password at his next login
- **-M** 30 jdoe: Change the max number of days during which a password is valid
- **-m** 7 jdoe: Change the min number of days between password changes
- **-W** 15 jdoe: Change the number of days before password expiration that the user will be warned
- **-I** 3 jdoe: Change the number of days after password expiration before the account is locked
- **-l** jdoe: List password aging information for a user

**groupadd** geeks: Create a group

**groupmod** 
- **-n** nerds geeks: Change a group name
- **-d** 0 jdoe: Delete a group

**gpasswd** geeks: Set or change the password of a group
- **-a** jdoe geeks: Add a user to a group
- **-d** jdoe geeks: Delete a user from a group
- **-A** jdoe geeks: Add a user to the list of administrators of the group

**adduser** (Debian): User-friendly front-end commands for user and group management

**deluser**

**addgroup** (Debian)

**delgroup**

**system-config-users** (Red Hat): GUI for user and group management
On a system, every user is identified by a numeric UID (User ID), and every group by a numeric GID (Group ID).

UID 0 is assigned to the superuser.
UIDs from 0 to 99 should be reserved for static allocation by the system and not be created by applications.
UIDs from 100 to 499 should be reserved for dynamic allocation by the superuser and post-install scripts.
UIDs for user accounts start from 500 (Red Hat) or 1000 (SUSE, Debian).

* as recommended by the Linux Standard Base core specifications

A process has an effective, saved, and real UID and GID.

| Effective UID | Used for most access checks, and as the owner for files created by the process. An unprivileged process can change its effective UID only to either its saved UID or its real UID. |
| Saved UID     | Used when a process running with elevated privileges needs to temporarily lower its privileges. The process changes its effective UID (usually root) to an unprivileged one, and its privileged effective UID is copied to the saved UID. Later, the process can resume its elevated privileges by resetting its effective UID back to the saved UID. |
| Real UID      | Used to identify the real owner of the process and affect the permissions for sending signals. An unprivileged process can signal another process only if the sender’s real or effective UID matches the receiver’s real or saved UID. Child processes inherit the credentials from the parent, so they can signal each other. |

/etc/login.defs Definition of default values (UID and GID ranges, mail directory, account validity, password encryption method, and so on) for user account creation

whoami Print your username (as effective UID)

id Print your real and effective UID and GID, and the groups you are a member of

id -u Print your effective UID

id user Print UID, GID, and groups information about a user
**su and sudo**

**runuser -u user command**

- Run command as user. Can be launched only by root

**su user**

- Run a shell as user

**su**

- Run a shell as root

**su root**

- Pass a single command to the shell

**su -**

- Ensure that the spawned shell is a login shell, hence running login scripts and setting the correct environment variables. Recommended option

**su -c "fdisk -l"**

- Run a shell as root

**sudo -u user command**

- Run command as user

**sudo command**

- Run command as root

**sudo -uroot command**

- Login on an interactive shell as root

**sudo su -**

- List the allowed commands for the current user

**sudo -i**

- Run again the last command, but this time as root

**sudo -l**

- Edit a protected file. It is recommended to use this instead of allowing users to sudo text editors as root, which will cause security problems if the editor spawns a shell

**sudoedit /etc/passwd**

- Edit /etc/sudoers, the configuration file that specifies access rights to sudo

**visudo**

- Sudo commands are logged via syslog on /var/log/auth.log (Debian) or /var/log/secure (Red Hat).

**gksu -u root -l**

- GUI front-ends to su and sudo used to run an X Window command or application as root. Pops up a requester prompting the user for root's password
Terminals

Make /dev/tty the foreground terminal

Lock the virtual console (terminal)

Print your terminal device (e.g. /dev/tty1, /dev/pts/1)

Change or display terminal line settings

Disable XON/XOFF flow control

Prevent a process from terminating (receiving a SIGHUP) when its parent Bash dies.

When a Bash shell is terminated cleanly via `exit`, its jobs will become child of the Bash's parent and will continue running. When a Bash shell is killed instead, it issues a SIGHUP to his children which will terminate

Screen manager that multiplexes a single virtual VT100/ANSI terminal between multiple processes or shells. When the connection to a terminal is lost (e.g. because the terminal is closed manually, the user logs out, or the remote SSH session goes into timeout), a SIGHUP is sent to the shell and from there to all running child processes which are therefore terminated. The `screen` command starts an interactive shell screen session, to which the user will be able to reattach later

Start a screen session with the specified session name

Start the specified command in a screen session; session will end when the command exits

Show the list of detached screen sessions

Resume a detached screen session

Resume the last detached screen session

Detach a remote screen session and reattach your current terminal to it

Send a command to the window manager:
0 ... 9  Switch between screen sessions
  c  Create a new screen session
  ?  Show help

How to detach an already running job that was not started in a screen session

1. `CTRL Z`  Suspend the job
2. `bg`  Send the job to background
3. `jobs`  Show the number (let's assume is n) of the backgrounded job
4. `disown -h %n`  Mark job n so it will not receive a SIGHUP from its parent shell
or
1. `screen`  Start a screen session
2. `reptyr pid`  Attach the job with process ID pid to the new terminal (screen session)

Once done this, when the terminal is closed, the job will not be killed.
write user
 echo "Message" | write user

Write interactively a message to the terminal of user (which must be logged in)
Write a message to the terminal of user (which must be logged in)

wall
 echo "Message" | wall

Write interactively a message to the terminal of all logged in users
Write a message to the terminal of all logged in users

talk user

Open an interactive chat session with user (which must be logged in)

mesg y
 chmod g+w $(tty)

Allow the other users to message you via write, wall, and talk
Disallow the other users to message you via write, wall, and talk

mesg n
 chmod g-w $(tty)

mesg

Display your current message permission status

mesg works by enabling/disabling the group write permission of your terminal device, which is owned by system group tty. The root user is always able to message users.
cron is a job scheduler, allowing the repeated execution of commands specified in crontab files. The crond daemon checks the crontab files every minute and runs the command as the specified user at the specified times. It is not necessary to restart crond after the modification of a crontab file, as the changes will be reloaded automatically.

If /etc/cron.allow exists, only users listed therein can access the service. If /etc/cron.deny exists, all users except those listed therein can access the service. If none of these files exist, all users can access the service.

/etc/crontab
/etc/cron.d/*
/etc/cron.hourly/
/etc/cron.daily/
/etc/cron.weekly/
/etc/cron.monthly/
/var/spool/cron/
/var/spool/cron/user

CRONTAB

The crond daemon also runs anacron jobs, which allow the execution of periodic jobs on a machine that is not always powered on, such as a laptop. Only the superuser can schedule anacron jobs, which have a granularity of one day (vs one minute for cron jobs).

/var/spool/anacron/jobid

CRONTAB

The crond daemon also runs anacron jobs, which allow the execution of periodic jobs on a machine that is not always powered on, such as a laptop. Only the superuser can schedule anacron jobs, which have a granularity of one day (vs one minute for cron jobs).

/var/spool/anacron/jobid

CRONTAB
at is used for scheduled execution of commands that must run only once. Execution of these commands is the duty of the 
atd daemon.

If /etc/at.allow exists, only users listed therein can access the service.
If /etc/at.deny exists, all users except those listed therein can access the service.
If none of these files exist, no user except the superuser can access the service.

at 5:00pm tomorrow script.sh
at -f listofcommands.txt 5:00pm tomorrow
echo "rm file" | at now+2 minutes

Execute a command once at the specified time (absolute or relative)

at -l
atq
at -d 3
atrm 3

List the scheduled jobs
Remove job number 3 from the list
Utilities

bc
factor
units

cal

banner
figlet
toilet
lolcat

fortune

sensors
beep
speaker-test
on_ac_power

ipcalc

pwgen
pwqgen

uuidgen
aspell
cloc

gnome-terminal

gconf

conky

gkrellm

Calculator
Print the prime factors of an integer number
Converter of quantities between different units
Calendar
Print a text in large letters made of the character #
Print a text in large letters, in a specific font
Print a text in large colorful letters, in a specific font
Print a text in rainbow coloring
Print a random aphorism, like those found in fortune cookies
Print sensor chips information (e.g. temperature)
Produce a beep from the machine's speakers
Speaker test tone generator for the ALSA (Advanced Linux Sound Architecture) framework
Return 0 (true) if machine is connected to AC power, 1 (false) if on battery. Useful for laptops
IP addresses calculator
Random password generator
Random password generator with controllable quality
Generator of UUIDs (random or time-based)
Spell checker
Count lines of source code

GNOME shell terminal

Highly configurable system monitor widget with integration for audio player, email, and news
System monitor widget
Localization

<table>
<thead>
<tr>
<th>Locale environment variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>LANG</td>
</tr>
<tr>
<td>LANGUAGE</td>
</tr>
<tr>
<td>LC_NUMERIC</td>
</tr>
<tr>
<td>LC_TIME</td>
</tr>
<tr>
<td>LC_COLLATE</td>
</tr>
<tr>
<td>LC_MONETARY</td>
</tr>
<tr>
<td>LC_MESSAGES</td>
</tr>
<tr>
<td>LC_PAPER</td>
</tr>
<tr>
<td>LC_NAME</td>
</tr>
<tr>
<td>LC_ADDRESS</td>
</tr>
<tr>
<td>LC_TELEPHONE</td>
</tr>
<tr>
<td>LC_MEASUREMENT</td>
</tr>
<tr>
<td>LC_IDENTIFICATION</td>
</tr>
<tr>
<td>LC_ALL</td>
</tr>
</tbody>
</table>

The values of these locale environment variables are in the format `language_territory.encoding` e.g. `en_US.UTF-8`. The list of supported locales is stored in `/usr/share/i18n/SUPPORTED`.

```bash
locale
```
Show locale environment variables

```bash
locale-gen it_IT.UTF-8
```
Generate a locale (in this case IT) by compiling a list of locale definition files.

```bash
apt-get install manpages-it language-pack-it
```
Install a different locale (in this case IT); this affects system messages and manpages.

```bash
iconv -f IS6937 -t IS8859 filein > fileout
```
Convert a text file from a codeset to another

ISO/IEC-8859 is a standard for 8-bit encoding of printable characters. The first 256 characters in ISO/IEC-8859-1 (Latin-1) are identical to those in Unicode. UTF-8 encoding can represent every character in the Unicode set, and was designed for backward compatibility with ASCII.
### System time

- **date**
  - `date`: Show current date and time
  - `date -d "9999 days ago"`: Calculate a date and show it
  - `date -d "1970/01/01 + 4242"`: Show current date in the format specified
  - `date +%F %T`: Show current date in Unix time format (seconds elapsed since 00:00:00 1/1/1970)
  - `date +%s`: Set the date
  - `date "20130305 23:30:00"`: Set the date, in the format `MMDDhhmmYYYY`

- **timedatectl**
  - `timedatectl`: Show current date and time
  - `timedatectl set-time 2013-03-05`: Set the date
  - `timedatectl list-timezones`: List all possible timezones

- **zdump GMT**
  - `zdump`: Show current date and time in the GMT timezone

- **tzselect tzconfig dpkg-reconfigure tzdata timedatectl set-timezone timezone**
  - (Debian) `tzselect tzconfig dpkg-reconfigure tzdata`: Set the timezone
  - (Red Hat) `timedatectl set-timezone timezone`: Set the timezone

- **/etc/timezone** (Debian)
  - **/etc/localtime** (Red Hat)
  - Timezone, a symlink to the appropriate timezone file in `/usr/share/zoneinfo/`

- **ntpd**
  - NTP daemon, keeps the clock in sync with Internet time servers
  - `ntpd -q`: Synchronize the time once and quit
  - `ntpd -g`: Force NTP to start even if clock is off by more than the panic threshold (1000 secs)
  - `ntpd -ng`: Start NTP as a non-daemon, force synchronization of the clock, and quit. The NTP daemon must not be running when this command is launched

- **ntpdate timeserver**
  - Synchronizes the clock with the specified time server
  - `ntpdate -b timeserver`: Brutally set the clock, without waiting for it to adjust slowly
  - `ntpdate -q timeserver`: Query the time server without setting the clock

  **The ntpdate command is deprecated; to synchronize the clock, use ntpd instead.**

- **chronyd chronyc**
  - Daemon for chrony, a versatile NTP client/server
  - Command line interface for the chrony daemon

- **hwclock --show hwclock --hctosys hwclock --localtime hwclock --systohc hwclock --utc**
  - Show the hardware clock
  - Set the system time from the hardware clock
  - Set the hardware clock from system time
  - Indicate that the hardware clock is kept in Coordinated Universal Time
  - Indicate that the hardware clock is kept in local time
syslog

**syslogd** (Ubuntu 14)

Daemon logging events from user processes

**rsyslogd**

Daemon logging events from kernel processes

---

**/etc/syslog.conf**

```
# facility.level action
*.info;mail.none;authpriv.none /var/log/messages
authpriv.* /var/log/secure
mail.* /var/log/maillog
*.alert
*.emerg
local5.* @10.7.7
local7.* /var/log/boot.log
```

<table>
<thead>
<tr>
<th>Facility Creator of the message</th>
<th>Level Severity of the message</th>
<th>Action Destination of the message</th>
</tr>
</thead>
<tbody>
<tr>
<td>auth or security†</td>
<td>emerg or panic† (highest)</td>
<td>file message is written into a log file</td>
</tr>
<tr>
<td>authpriv cron</td>
<td>alert</td>
<td>@host message is sent to a logger server host (via UDP port 514)</td>
</tr>
<tr>
<td>daemon kern lpr mail</td>
<td>crit</td>
<td>user1,user2,user3 message is sent to the specified users’ consoles</td>
</tr>
<tr>
<td>mark news syslog user uucp</td>
<td>err or error†</td>
<td>* message is sent to all logged-in users’ consoles</td>
</tr>
<tr>
<td>local0 ... local7 (custom)</td>
<td>warning or warn†</td>
<td>notice info debug (lowest) none (facility disabled)</td>
</tr>
</tbody>
</table>

† = deprecated

Facilities and levels are listed in the manpage man 3 syslog.

---

**logger** `-p auth.info "Message"`

Send a message to syslog with facility "auth" and priority "info"

**logrotate**

Rotate logs. It gzips, renames, and eventually deletes old logfiles according to the configuration files `/etc/logrotate.conf` and `/etc/logrotate.d/*`. It is usually scheduled as a daily cron job

/var/log/messages

Global system logfile

/var/log/dmesg

Kernel ring buffer information

/var/log/kern.log

Kernel log

/var/log/boot.log

Information logged during boot

/var/log/secure

Information about failed authentication and authorization (e.g. sshd failed logins)
### E-mail

**Mailbox formats**

<table>
<thead>
<tr>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
</table>
| **mbox** | Each mail folder is a single file, storing multiple email messages.  
Advantages: universally supported; fast search inside a mail folder.  
Disadvantages: issues with file locking; possible mailbox corruption. |
| **Maildir** | Each mail folder is a directory, and contains the subdirectories /cur, /new, and /tmp.  
Each email message is stored in its own file with a unique filename ID.  
The process that delivers an email message writes it to a file in the tmp/ directory, and then moves it to new/.  The moving is commonly done by hard linking the file to new/ and then unlinking the file from tmp/, which guarantees that a MUA will not see a partially written message as it never looks in tmp/.  
When the MUA finds mail messages in new/ it moves them to cur/.  
Advantages: fast location/retrieval/deletion of a specific mail message; no file locking needed; can be used with NFS.  
Disadvantages: some filesystems may not efficiently handle a large number of small files; searching text inside all mail messages is slower. |

---

```
- ~/.forward
/ etc/ aliases
/ etc/ mail/ aliases
/ var/ spool/ mail/ user
/ var/ log/ mail. log  (Debian)
/ var/ log/ maillog  (Red Hat)

mail
mailx

mailx -s "Subject" -S smtp="mailserver:25" \  
user@domain.com < messagefile
uuencode binaryfile | mail user@domain.com

mutt -a binaryfile -- user@domain.com < /dev/null
```

Send a mail message to *user@domain.com* via an external SMTP server *mailserver*

Send a binary file to *user@domain.com* (not recommended because many mailclients will display the received attachment inline)

Send a binary file to *user@domain.com* using the Mutt MUA
### SMTP commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HELO xyz.linux.org</td>
<td>Initiate the conversation and identify client host to server</td>
</tr>
<tr>
<td>EHLO xyz.linux.org</td>
<td>Like HELO, but tell server to use Extended SMTP</td>
</tr>
<tr>
<td>MAIL FROM: <a href="mailto:alice@linux.org">alice@linux.org</a></td>
<td>Specify mail sender</td>
</tr>
<tr>
<td>RCPT TO <a href="mailto:bob@foobar.com">bob@foobar.com</a></td>
<td>Specify mail recipient</td>
</tr>
<tr>
<td>RCPT TO <a href="mailto:carol@quux.net">carol@quux.net</a></td>
<td>Specify data to send. Ended with a dot on a single line</td>
</tr>
<tr>
<td>DATA</td>
<td>Disconnect</td>
</tr>
<tr>
<td>HELP</td>
<td>List all available commands</td>
</tr>
<tr>
<td>NOOP</td>
<td>Empty command</td>
</tr>
<tr>
<td>VRFY <a href="mailto:alice@linux.org">alice@linux.org</a></td>
<td>Verify the existence of an e-mail address (this command should not be implemented, for security reasons)</td>
</tr>
<tr>
<td>EXPN mailinglist</td>
<td>Check mailing list membership</td>
</tr>
</tbody>
</table>

```plaintext
220 smtp.example.com ESMTP Postfix (server)
HELO xyz.linux.org
250 Hello xyz.linux.org, glad to meet you
MAIL FROM: alice@linux.org
250 Ok
RCPT TO bob@foobar.com
250 Ok
RCPT TO carol@quux.net
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: Alice <alice@linux.org>
To: Bob <bob@foobar.com>
Cc: Carol <carol@quux.net>
Date: Wed, 13 August 2014 18:02:43 -0500
Subject: Test message

This is a test message.
.
250 OK id=1OjReS-0005kT-Jj
QUIT
221 Bye
```

### SMTP response codes

<table>
<thead>
<tr>
<th>first digit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Command accepted, but not processed until client sends confirmation</td>
</tr>
<tr>
<td>2</td>
<td>Command successfully completed</td>
</tr>
<tr>
<td>3</td>
<td>Command accepted, but not processed until client sends more information</td>
</tr>
<tr>
<td>4</td>
<td>Command failed due to temporary errors</td>
</tr>
<tr>
<td>5</td>
<td>Command failed due to permanent errors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>second digit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Syntax error or command not implemented</td>
</tr>
<tr>
<td>1</td>
<td>Informative response in reply to a request for information</td>
</tr>
<tr>
<td>2</td>
<td>Connection response in reply to a data transmission</td>
</tr>
<tr>
<td>5</td>
<td>Status response in reply to a mail transfer operation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>third digit</th>
<th>Specifies further the response</th>
</tr>
</thead>
<tbody>
<tr>
<td>211</td>
<td>System status or help reply</td>
</tr>
<tr>
<td>214</td>
<td>Help message</td>
</tr>
<tr>
<td>220</td>
<td>The server is ready</td>
</tr>
<tr>
<td>221</td>
<td>The server is ending the conversation</td>
</tr>
<tr>
<td>250</td>
<td>The requested action was completed</td>
</tr>
<tr>
<td>251</td>
<td>The specified user is not local, but the server will forward the mail message</td>
</tr>
<tr>
<td>354</td>
<td>Reply to the DATA command. After getting this, start sending the message body</td>
</tr>
<tr>
<td>421</td>
<td>The mail server will be shut down, try again later</td>
</tr>
<tr>
<td>450</td>
<td>The mailbox that you are trying to reach is busy, try again later</td>
</tr>
<tr>
<td>451</td>
<td>The requested action was not done. Some error occurred in the mail server</td>
</tr>
<tr>
<td>452</td>
<td>The requested action was not done. The mail server ran out of system storage</td>
</tr>
<tr>
<td>500</td>
<td>The last command contained a syntax error or the command line was too long</td>
</tr>
<tr>
<td>501</td>
<td>The parameters or arguments in the last command contained a syntax error</td>
</tr>
<tr>
<td>502</td>
<td>The last command is not implemented in the mail server</td>
</tr>
<tr>
<td>503</td>
<td>The last command was sent out of sequence</td>
</tr>
<tr>
<td>504</td>
<td>One of the parameters of the last command is not implemented by the server</td>
</tr>
<tr>
<td>550</td>
<td>The mailbox that you are trying to reach can't be found or you don't have access rights</td>
</tr>
<tr>
<td>551</td>
<td>The specified user is not local; part of message text will contain a forwarding address</td>
</tr>
<tr>
<td>552</td>
<td>The mailbox that you are trying to reach has run out of space, try again later</td>
</tr>
<tr>
<td>553</td>
<td>The mail address that you specified was not syntactically correct</td>
</tr>
<tr>
<td>554</td>
<td>The mail transaction has failed for unknown causes</td>
</tr>
</tbody>
</table>
Sendmail is an MTA distributed as a monolithic binary file. Previous versions used to run SUID root, which caused many security problems; recent versions run SGID smmsp, the group that has write access on the mail queue. Sendmail uses smrsh, a restricted shell, to run some external programs.

/submit.cf
 sendmail.cf

Sendmail local mail transfer configuration file
Sendmail MTA configuration file

/m4 /etc/mail/submit.mc > /etc/mail/submit.cf

Generate a .cf configuration file from an editable .mc text file.
_cf configuration files must not be edited by hand

/etc/mail/

access.db
local-host-names.db
virtusertable.db
mailertable.db
domaintable.db
genericstable.db
genericsdomain.db

Access control file to allow or deny access to systems or users
List of domains that must be considered as local accounts
Map for local accounts, used to distribute incoming email
Routing table, used to dispatch emails from remote systems
Domain table, used for transitions from an old domain to a new one
Map for local accounts, used to specify a different sender for outgoing mail
Local FQDN

makemap hash /etc/mail/access.db < /etc/mail/access

Generate a .db database file from an editable text file.
.db database files must not be edited by hand

Temporary mailqueue files (where nnn is the Message ID):

dfnnn
qfnnn
Qfnnn
hnnn
fnnn
tfnnn
lfnnn
nfnnn
xfnnn

Mail body
Message envelope with headers and routing information
Message envelope if abandoned
Message envelope if held / quarantined by a milter (i.e. mail filter)
Temporary file
Lock file
Backup file
Transcript of delivery attempts

newaliases
sendmail -bi
mailq
sendmail -bp
sendmail -bt
sendmail -q
hoststat
purgestat
mailstats
praliases

Update the aliases database. Must be run after any change to /etc/aliases
Examine the mail queue
Run Sendmail in test mode
Force a queue run
Print statistics about remote hosts usage
Clear statistics about remote host usage
Print statistics about the mailserver
Display email aliases
Exim is a free MTA, distributed under open source GPL license.

/etc/exim.conf
/usr/local/etc/exim/configure (FreeBSD)  Exim4 configuration file

exim4 -bp  Examine the mail queue
exim4 -M messageID  Attempt delivery of message
exim4 -Mrm messageID  Remove a message from the mail queue
exim4 -Mvh messageID  See the headers of a message in the mail queue
exim4 -Mvb messageID  See the body of a message in the mail queue
exim4 -Mvc messageID  See a message in the mail queue
exim4 -qf domain  Force a queue run of all queued messages for a domain
exim4 -Rff domain  Attempt delivery of all queued messages for a domain
exim4 -bV  Show version and other info

exinext  Give the times of the next queue run
exigrep  Search through Exim logfiles
exicyclog  Rotate Exim logfiles
Postfix is a fast, secure, easy to configure, open source MTA intended as a replacement for Sendmail. It is implemented as a set of small helper daemons, most of which run in a chroot jail with low privileges. The main ones are:

- **master**: Postfix master daemon, always running; starts the other daemons when necessary
- **nqmgr**: Queue manager for incoming and outgoing mail, always running
- **smtpd**: SMTP daemon for incoming mail
- **smtp**: SMTP daemon for outgoing mail
- **bounce**: Manager of bounce messages
- **cleanup**: Daemon that verifies the syntax of outgoing messages before they are handed to the queue manager
- **local**: Daemon that handles local mail delivery
- **virtual**: Daemon that handles mail delivery to virtual users

```
/var/spool/postfix/
```

- **incoming**: Incoming queue. All new mail entering the Postfix queue is written here by the cleanup daemon. Under normal conditions this queue is nearly empty
- **active**: Active queue. Contains messages ready to be sent. The queue manager places messages here from the incoming queue as soon as they are available
- **deferred**: Deferred queue. A message is placed here when all its deliverable recipients are delivered, and delivery failed for some recipients for a transient reason. The queue manager scans this queue periodically and puts some messages back into the active queue to retry sending

```
/var/spool/postfix/
```

- **bounce**: Message delivery status report about why mail is bounced (non-delivered mail)
- **defer**: Message delivery status report about why mail is delayed (non-delivered mail)
- **trace**: Message delivery status report (delivered mail)

**Command Line Tools**

- **postfix reload**: Reload configuration
- **postconf -e 'mydomain = example.org'**: Edit a setting in the Postfix configuration
- **postconf -l**: List supported mailbox lock methods
- **postconf -m**: List supported database types
- **postconf -v**: Increase logfile verbosity
- **postmap dbtype:textfile**: Manage Postfix lookup tables, creating a hashed map file of database type `dbtype` from `textfile`
- **postmap hash:/etc/postfix/transport**: Regenerate the transport database
- **postalias**: Convert `/etc/aliases` into the aliases database file `/etc/aliases.db`
- **postsuper**: Operate on the mail queue
- **postqueue**: Unprivileged mail queue manager
Postfix configuration

/etc/postfix/main.cf  Postfix main configuration file

mydomain = example.org  This system’s domain
myorigin = $mydomain  Domain from which all sent mail will appear to originate
myhostname = foobar.$mydomain  This system’s hostname
inet_interfaces = all  Network interface addresses that this system receives mail on.
proxy_interfaces = all  Value can also be localhost, all, or loopback-only
mynetworks = 10.3.3.0/24 !10.3.3.66  Networks the SMTP clients are allowed to connect from
mydestination = $myhostname, localhost, $mydomain, example.com,
   hash:/etc/postfix/otherdomains  Domains for which Postfix will accept received mail.
   Value can also be a lookup database file e.g. a hashed map
relayhost = 10.6.6.6  Relay host to which Postfix should send all mail for delivery,
   instead of consulting DNS MX records
relay_domains = $mydestination  Sources and destinations for which mail will be relayed.
   Can be empty if Postfix is not intended to be a mail relay
virtual_alias_domains = virtualex.org  Set up Postfix to handle mail for virtual domains too.
virtual_alias_maps = /etc/postfix/virtual  The /etc/postfix/virtual file is a hashed map, each line of
   or  the file containing the virtual domain email address and the
   virtual_alias_domains = hash:/etc/postfix/virtual  destination real domain email address:
mailbox_command = /usr/bin/procmail  jdoe@virtualex.org    john.doe@example.org
   ksmit@gmail.org   kim.smith
   The @virtualex.org in the last line is a catch-all specifying
   that all other email messages to the virtual domain are
delivered to the root user on the real domain

A line beginning with whitespace or tab is a continuation of the previous line.
A line beginning with a # is a comment.  A # not placed at the beginning of a line is not a comment delimiter.

/etc/postfix/master.cf  Postfix master daemon configuration file

# service type private unpriv chroot wakeup maxproc command + args
smtp    inet n  -      -      -      -      -       smtpd
pickup  fifo n   -      -     60     1      pickup
cleanup unix n   -      -     0      cleanup
qmgr    fifo n   -      -     300    1      qmgr
rewrite unix -    -      -      -      trivial-rewrite
bounce  unix -    -      -      -      bounce
defer   unix -    -      -      -      bounce
flush   unix n    -      1000?  0      flush
smtp    unix -    -      -      -      smtp
showq  unix n    -      -      -      showq
error   unix -    -      -      -      error
local   unix -    n      n      -      local
virtual unix -    n      n      -      virtual
lmtp    unix -    -      n      -      lmtp

service  Name of the service
type     Transport mechanism used by the service
private  Whether the service is accessible only by Postfix daemons and not by the whole system. Default is yes
unprivileged  Whether the service is unprivileged i.e. not running as root. Default is yes
chroot  Whether the service is chrooted. Default is yes
wakeup  How often the service needs to be woken up by the master daemon. Default is never
maxproc  Max number of simultaneous processes providing the service. Default is 50
command  Command used to start the service

The − indicates that an option is set to its default value.
Procmail is a regex-based MDA whose main purpose is to preprocess and sort incoming email messages. It is able to work both with the standard mbox format and the Maildir format.

To have all email processed by Procmail, the `~/.forward` file may be edited to contain:

```
"[exec /usr/local/bin/procmail || exit 75"
```

<table>
<thead>
<tr>
<th>Procmail recipes</th>
<th>/etc/procmailrc and ~/.procmailrc</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>/etc/procmailrc</strong></td>
<td>System-wide recipes</td>
</tr>
<tr>
<td><strong>~/.procmailrc</strong></td>
<td>User's recipes</td>
</tr>
<tr>
<td><strong>procmail -h</strong></td>
<td>List all Procmail flags for recipes</td>
</tr>
<tr>
<td><strong>formail</strong></td>
<td>Utility for email filtering and editing</td>
</tr>
<tr>
<td><strong>lockfile</strong></td>
<td>Utility for mailbox file locking</td>
</tr>
<tr>
<td><strong>mailstat</strong></td>
<td>Utility for generation of reports from Procmail logs</td>
</tr>
</tbody>
</table>

### Common parameters, nonspecific to Procmail

| **PATH**<br>$HOME/bin:/usr/bin:/bin:/usr/sbin:/sbin |
| **MAILDIR**<br>$HOME/Mail |
| **DEFAULT**<br>$MAILDIR/Inbox |
| **LOGFILE**<br>$HOME/.procmaillog |

### Flag: match headers (default) and use file locking (highly recommended when writing to a file or a mailbox in mbox format)

- **:0:**<br> * ^From: .* (alice|bob)@foobar.org | DEFAULT

  - **Condition:** match the header specifying the sender address
  - **Destination:** default mailfolder

### Conditions: match sender address and subject headers

- **:0:**<br> * ^From: .*owner@listserv\.com <br>  * ^Subject: .*Linux | $MAILDIR/Geekstuff1

### Flaq: file locking not necessary because using Maildir format

- **:0:**<br> * ^From: .*owner@listserv\.com <br>  * ^Subject: .*Linux | $MAILDIR/Geekstuff2/

### # Blacklisted by SpamAssassin

- **:0:**<br> * ^X-Spam-Status: Yes | /dev/null

  - **Flag:** file locking not necessary because blackholing to /dev/null
  - **Condition:** match SpamAssassin’s specific header
  - **Destination:** delete the message

### Flaq: match body of message instead of headers

- **:0B:**<br> * hacking $MAILDIR/Geekstuff

### Flaq: match either headers or body of message

- **:0HB:**<br> * hacking $MAILDIR/Geekstuff

### Condition: match messages larger than 256 Kb

- **:0:**<br> * > 256000 <br>  | /root/myprogram

  - **Condition:** match messages larger than 256 Kb
  - **Destination:** pipe message through the specified program

### Flaqs: use the pipe as a filter (modifying the message), and have Procmail wait that the filter finished processing the message

- **:0fw**<br> * ^From: .*@foobar\.org <br>  | /root/myprogram

### Flaq: copy the message and proceed with next recipe

- **:0c**<br> * ^Subject: .*administration ! secretary@domain.com <br>  | $MAILDIR/Forwarded

  - **Flag:** copy the message and proceed with next recipe
  - **Destination:** forward to specified email address, and (this is ordered by the next recipe) save in the specified mailfolder
The Courier MTA provides modules for ESMTP, IMAP, POP3, webmail, and mailing list services in a single framework. To use Courier, it is necessary first to launch the `courier-authlib` service, then launch the desired mail service e.g. `courier-imap` for the IMAP service.

```
/imusr/lib/courier-imap/etc/
or
/etc/courier/

imapi
d Courier IMAP daemon configuration

imapd-ssl
 Courier IMAPS daemon configuration

pop3d
 Courier POP3 daemon configuration

pop3d-ssl
 Courier POP3S daemon configuration

/usr/lib/courier-imap/share/

mkimapdcert
 Generate a certificate for the IMAPS service

mkpop3dcert
 Generate a certificate for the POP3 service

makealiases
 Create system aliases in `/usr/lib/courier/etc/aliases.dat`, which is
 made by processing a `/usr/lib/courier/etc/aliases/system` text file:

root          : postmaster
mailer-daemon : postmaster
MAILER-DAEMON : postmaster
uucp          : postmaster
postmaster    : admin

/usr/lib/courier-imap/etc/pop3d

Couriier POP configuration file

ADDRESS=0
 Address on which to listen. 0 means all addresses

PORT=127.0.0.1.900,192.168.0.1.900
 Port number on which connections are accepted. In this case, accept
 connections on port 900 on IP addresses 127.0.0.1 and 192.168.0.1

POP3AUTH="LOGIN CRAM-MD5 CRAM-SHA1"
 POP authentication advertising SASL (Simple Authentication and
 Security Layer) capability, with CRAM-MD5 and CRAM-SHA1

POP3AUTH_TLS="LOGIN PLAIN"
 Also advertise SASL PLAIN if SSL is enabled

MAXDAEMONS=40
 Maximum number of POP3 servers started

MAXPERIP=4
 Maximum number of connections to accept from the same IP address

PIDFILE=/var/run/courier/pop3d.pid
 PID file

TCPOPT="-nodnslookup -noidentlookup"
 Miscellaneous couriertcpd options. Should not be changed

LOGGEROPTS="-name=pop3d"
 Options for courierlogger

POP3_PROXY=0
 Enable or disable proxying

PROXY_HOSTNAME=myproxy
 Override value from `gethostname()` when checking if a proxy
 connection is required

DEDFDOMAIN="@example.com"
 Optional default domain. If the username does not contain the first
 character of DEFSIZEN, then it is appended to the username. If
 DEFSIZEN and DOMAINSEP are both set, then DEFSIZEN is appended
 only if the username does not contain any character from DOMAINSEP

POP3DSTART=YES
 Flag intended to be read by the system startup script

MAILDIRPATH=Maildir
 Maildir directory
```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDRESS=0</td>
<td>Address on which to listen. 0 means all addresses</td>
</tr>
<tr>
<td>PORT=127.0.0.1.900,192.168.0.1.900</td>
<td>Port number on which connections are accepted. In this case, accept connections on port 900 on IP addresses 127.0.0.1 and 192.168.0.1</td>
</tr>
<tr>
<td>AUTHSERVICE143=imap</td>
<td>Authenticate using a different service parameter depending on the connection's port. This only works with authentication modules that use the service parameter, such as PAM</td>
</tr>
<tr>
<td>MAXGAEMONS=40</td>
<td>Maximum number of IMAP servers started</td>
</tr>
<tr>
<td>MAXPERIP=20</td>
<td>Maximum number of connections to accept from the same IP address</td>
</tr>
<tr>
<td>PIDFILE=/var/run/courier/imapd.pid</td>
<td>PID file for courier tcpd</td>
</tr>
<tr>
<td>TCPDOPTS=&quot;-nodnslookup -noidentlookup&quot;</td>
<td>Miscellaneous courier tcpd options. Should not be changed</td>
</tr>
<tr>
<td>LOGGEROPTS=&quot;-name=imapd&quot;</td>
<td>Options for courier logger</td>
</tr>
<tr>
<td>DEFDOMAIN=&quot;@example.com&quot;</td>
<td>Optional default domain. If the username does not contain the first character of DEFDOMAIN, then it is appended to the username. If DEFDOMAIN and DOMAINSEP are both set, then DEFDOMAIN is appended only if the username does not contain any character from DOMAINSEP</td>
</tr>
<tr>
<td>IMAP_CAPABILITY=&quot;IMAP4rev1 UIDPLUS \ CHILDREN NAMESPACE THREAD=ORDEREDSUBJECT \ THREAD=REFERENCES SORT QUOTA IDLE&quot;</td>
<td>Specifies what most of the response should be to the CAPABILITY command</td>
</tr>
<tr>
<td>IMAP_KEYWORDS=1</td>
<td>Enable or disable custom IMAP keywords. Possible values are: 0 disable keywords, 1 enable keywords, 2 enable keywords with a slower algorithm</td>
</tr>
<tr>
<td>IMAP_ACL=1</td>
<td>Enable or disable IMAP ACL extension</td>
</tr>
<tr>
<td>SMAP_CAPABILITY=SMAP1</td>
<td>Enable the experimental Simple Mail Access Protocol extensions</td>
</tr>
<tr>
<td>IMAP_PROXY=0</td>
<td>Enable or disable proxying</td>
</tr>
<tr>
<td>IMAP_PROXY_FOREIGN=0</td>
<td>Proxying to non-Courier servers. Resends the CAPABILITY command after logging in to remote server. May not work with all IMAP clients</td>
</tr>
<tr>
<td>IMAP_IDLE_TIMEOUT=60</td>
<td>How often, in seconds, the server should poll for changes to the folder while in IDLE mode</td>
</tr>
<tr>
<td>IMAP_CHECK_ALL_FOLDERS=0</td>
<td>Enable or disable server check for mail in every folder</td>
</tr>
<tr>
<td>IMAP_USELOCKS=1</td>
<td>Enable or disable dot-locking to support concurrent multiple access to the same folder. Strongly recommended when using shared folders</td>
</tr>
<tr>
<td>IMAP_SHAREDINDEXFILE=&quot;/etc/courier/shared/index&quot;</td>
<td>Index of all accessible folders. This setting should normally not be changed</td>
</tr>
<tr>
<td>IMAP_TRASHFOLDERNAME=Trash</td>
<td>Trash folder</td>
</tr>
<tr>
<td>IMAP_EMPTYTRASH=Trash:7,Sent:30</td>
<td>Purge folders i.e. delete all messages from the specified folders after the specified number of days</td>
</tr>
<tr>
<td>IMAP_MOVE_EXPUNGE_TO_TRASH=0</td>
<td>Enable or disable moving expunged messages to the trash folder (instead of directly deleting them)</td>
</tr>
<tr>
<td>HEADERFROM=X-IMAP-Sender</td>
<td>Save the return address (SENDER) in the X-IMAP-Sender mail header. This header is added to the sent message, but not in the copy of the message saved in the folder</td>
</tr>
<tr>
<td>MAILDIRPATH=Maildir</td>
<td>Mail directory</td>
</tr>
</tbody>
</table>
Dovecot is an open source, security-hardened, fast, and efficient IMAP and POP3 server. It implements its own high-performance dbox mailbox format. By default, it uses PAM authentication. The script `mkcert.sh` can be used to create self-signed SSL certificates.

<table>
<thead>
<tr>
<th><strong>Dovecot configuration file</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>base_dir = /var/run/dovecot/</td>
<td>Base directory where to store runtime data</td>
</tr>
<tr>
<td>protocols = imaps pop3s</td>
<td>Protocols to serve. If Dovecot should use <code>dovecot-auth</code>, this can be set to <code>none</code></td>
</tr>
<tr>
<td>listen = *, [::]</td>
<td>Network interfaces on which to accept connections. In this case, listen to all IPv4 and IPv6 interfaces</td>
</tr>
<tr>
<td>disable_plaintext_auth = yes</td>
<td>If yes, disable LOGIN command and all other plaintext authentications unless SSL/TLS is used (LOGINDISABLED capability)</td>
</tr>
<tr>
<td>shutdown_clients = yes</td>
<td>If yes, kill all IMAP and POP3 processes when Dovecot master process shuts down; if no, Dovecot can be upgraded without forcing existing client connections to close</td>
</tr>
<tr>
<td>log_path = /dev/stderr</td>
<td>Log file to use for error messages, instead of sending them to syslog. In this case, log to stderr</td>
</tr>
<tr>
<td>info_log_path = /dev/stderr</td>
<td>Log file to use for informational and debug messages. Default value is the same as <code>log_path</code></td>
</tr>
<tr>
<td>syslog_facility = mail</td>
<td>Syslog facility to use, if logging to syslog</td>
</tr>
<tr>
<td>login_dir = /var/run/dovecot/login</td>
<td>Directory where the authentication process places authentication UNIX sockets. The login process needs to be able to connect to these sockets</td>
</tr>
<tr>
<td>login_chroot = yes</td>
<td>Chroot login process to the <code>login_dir</code></td>
</tr>
<tr>
<td>login_user = dovecot</td>
<td>User for the login process and for access control in the authentication process. This is not the user that will access mail messages</td>
</tr>
<tr>
<td>login_process_size = 64</td>
<td>Maximum login process size, in Mb</td>
</tr>
<tr>
<td>login_process_per_connection = yes</td>
<td>If yes, each login is processed in its own process (more secure); if no, each login process processes multiple connections (faster)</td>
</tr>
<tr>
<td>login_processes_count = 3</td>
<td>Number of login processes to keep for listening for new connections</td>
</tr>
<tr>
<td>login_max_processes_count = 128</td>
<td>Maximum number of login processes to create</td>
</tr>
<tr>
<td>login_max_connections = 256</td>
<td>Maximum number of connections allowed per each login process. This setting is used only if <code>login_process_per_connection = no</code>; once the limit is reached, the process notifies master so that it can create a new login process</td>
</tr>
<tr>
<td>login_greeting = Dovecot ready.</td>
<td>Greeting message for clients</td>
</tr>
<tr>
<td>login_trusted_networks = \</td>
<td>Trusted network ranges (usually IMAP proxy servers). Connections from these IP addresses are allowed to override their IP addresses and ports, for logging and authentication checks. <code>disable_plaintext_auth</code> is also ignored for these networks</td>
</tr>
<tr>
<td>10.7.7.0/24 10.8.8.0/24</td>
<td></td>
</tr>
<tr>
<td>mbox_read_locks = fcntl</td>
<td>Locking methods to use for locking mailboxes in mbox format. Possible values are:</td>
</tr>
<tr>
<td>mbox_write_locks = dotlock fcntl</td>
<td>dotlock</td>
</tr>
<tr>
<td></td>
<td>dotlock_try</td>
</tr>
<tr>
<td></td>
<td>fcntl</td>
</tr>
<tr>
<td></td>
<td>flock</td>
</tr>
<tr>
<td></td>
<td>lockf</td>
</tr>
<tr>
<td>maildir_stat_dirs = no</td>
<td>Option for mailboxes in Maildir format. If no (default), the LIST command returns all entries in the mail directory beginning with a dot; if yes, returns only entries which are directories</td>
</tr>
<tr>
<td>dbox_rotate_size = 2048</td>
<td>Maximum and minimum file size, in Kb, of a mailbox in dbox format until it is rotated</td>
</tr>
<tr>
<td>dbox_rotate_min_size = 16</td>
<td></td>
</tr>
<tr>
<td>!include /etc/dovecot/conf.d/*.conf</td>
<td>Include configuration file</td>
</tr>
<tr>
<td>!include_try /etc/dovecot/extra.conf</td>
<td>Include optional configuration file, and do not report an error if file is not found</td>
</tr>
</tbody>
</table>
### Dovecot mailbox configuration

<table>
<thead>
<tr>
<th>/etc/dovecot.conf</th>
<th>Dovecot configuration file</th>
</tr>
</thead>
</table>
| mail_location = \ mbox:~/INBOX=/var/spool/mail/%u or mail_location = maildir:~/Maildir | Mailbox location, in mbox or Maildir format. Variables: 
  %u username
  %n user part in user@domain, same as %u if there is no domain 
  %d domain part in user@domain, empty if there is no domain 
  %h home directory |
| namespace shared { | Definition of a shared namespace, for accessing other users' mailboxes that have been shared. 
Private namespaces are for users' personal emails. 
Public namespaces are for shared mailboxes managed by root user |
  separator = / | Hierarchy separator to use. It should be the same for all namespaces, and depends on the underlying mail storage format |
  prefix = shared/%u/ | Prefix required to access this namespace; must be different for each. 
In this case, mailboxes are visible under shared/user@domain/; the variables %n, %d, and %u are expanded to the destination user |
  location = maildir:%h/Maildir:\ INDEX=/Maildir/shared/%u | Mailbox location for other users' mailboxes; it is in the same format as mail_location which is also the default for it. 
%variable and ~/ expand to the logged in user's data; 
%variable expands to the destination user's data |
  inbox = no | Define whether this namespace contains the INBOX. Note that there can be only one INBOX across all namespaces |
  hidden = no | Define whether the namespace is hidden i.e. not advertised to clients via NAMESPACE extension |
  subscriptions = no | Namespace handles its own subscriptions; if set to no, the parent namespace handles them and Dovecot uses the default namespace for saving subscriptions. If prefix is empty, this should be set to yes |
  list = children | Show the mailboxes under this namespace with LIST command, making the namespace visible for clients that do not support the NAMESPACE extension. 
In this case, lists child mailboxes but hide the namespace prefix; list the namespace only if there are visible shared mailboxes |
| mail_uid = 666 mail_gid = 666 | UID and GID used to access mail messages |
| mail_privileged_group = mail | Group to enable temporarily for privileged operations. Currently this is used only with INBOX when its initial creation or a dotlocking fails |
| mail_access_groups = tmpmail | Supplementary groups to with grant access for mail processes. Used typically to set up access to shared mailboxes |
| lock_method = fcntl | Locking method for index files. Can be fcntl, flock, or dotlock |
| first_valid_uid = 500 last_valid_uid = 0 | Valid UID range for users; default is 500 and above. This makes sure that users cannot login as daemons or other system users. Denying root login is hardcoded to Dovecot and cannot be bypassed |
| first_valid_gid = 1 last_valid_gid = 0 | Valid GID range for users; default is non-root. 
Users with invalid primary GID are not allowed to login |
| max_mail_processes = 512 | Maximum number of running mail processes. 
When this limit is reached, new users are not allowed to login |
| max_mail_processes = 512 | Maximum number of running mail processes. 
When this limit is reached, new users are not allowed to login |
| mail_process_size = 256 | Maximum mail process size, in Mb |
| valid_chroot_dirs = | List of directories under which chrooting is allowed for mail processes |
| mail_chroot = | Default chroot directory for mail processes. Usually not needed as Dovecot does not allow users to access files outside their mail directory |
| mailbox_idle_check_interval = 30 | Minimum time, in seconds, to wait between mailbox checks. 
When the IDLE command is running, mailbox is checked periodically for new mails or other changes |
### Dovecot POP/IMAP configuration

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/etc/dovecot.conf</td>
<td>Dovecot configuration file</td>
</tr>
<tr>
<td>protocol pop3</td>
<td>Block with options for the POP3 protocol</td>
</tr>
<tr>
<td>listen</td>
<td>Network interfaces on which to accept POP3 connections</td>
</tr>
<tr>
<td>login_executable</td>
<td>Location of the POP3 login executable</td>
</tr>
<tr>
<td>mail_executable</td>
<td>Location of the POP3 mail executable</td>
</tr>
<tr>
<td>pop3_no_flag_updates</td>
<td>If set to no, do not try to set mail messages non-recent or seen with POP3 sessions, to reduce disk I/O. With Maildir format do not move files from new/ to cur/; with mbox format do not write Status- headers</td>
</tr>
<tr>
<td>pop3_lock_session</td>
<td>Defines whether to keep the mailbox locked for the whole POP3 session</td>
</tr>
<tr>
<td>pop3_uidl_format</td>
<td>POP3 UIDL (Unique Mail Identifier) format to use</td>
</tr>
<tr>
<td>protocol imap</td>
<td>Block with options for the IMAP protocol</td>
</tr>
<tr>
<td>listen</td>
<td>Network interfaces on which to accept IMAP and IMAPS connections</td>
</tr>
<tr>
<td>ssl_listen</td>
<td>Location of the IMAP login executable</td>
</tr>
<tr>
<td>mail_executable</td>
<td>Location of the IMAP mail executable</td>
</tr>
<tr>
<td>mail_max_userip_connections</td>
<td>Maximum number of IMAP connections allowed for a user from each IP address</td>
</tr>
<tr>
<td>imap_idle_notify_interval</td>
<td>Waiting time, in seconds, between &quot;OK Still here&quot; notifications when client is IDLE</td>
</tr>
<tr>
<td>ssl</td>
<td>SSL/TLS support. Possible values are yes, no, required</td>
</tr>
<tr>
<td>ssl_cert_file</td>
<td>Location of the SSL certificate</td>
</tr>
<tr>
<td>ssl_key_file</td>
<td>Location of private key</td>
</tr>
<tr>
<td>ssl_key_password</td>
<td>Password of private key, if it is password-protected. Since /etc/dovecot.conf is usually world-readable, it is better to place this setting into a root-owned 0600 file instead and include it via the setting !include_try /etc/dovecot/dovecot-passwd.conf. Alternatively, Dovecot can be started with dovecot -p p4ssw0rd</td>
</tr>
<tr>
<td>ssl_ca_file</td>
<td>List of trusted SSL certificate authorities. This file contains CA certificates followed by CRLs</td>
</tr>
<tr>
<td>ssl_verify_client_cert</td>
<td>Request client to send a certificate</td>
</tr>
<tr>
<td>ssl_cipher_list</td>
<td>List of SSL ciphers to use</td>
</tr>
<tr>
<td>verbose_ssl</td>
<td>Show protocol level SSL errors</td>
</tr>
</tbody>
</table>
### Dovecot authentication

<table>
<thead>
<tr>
<th>/etc/dovecot.conf</th>
<th>Dovecot configuration file</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>auth_executable</strong> = /usr/libexec/dovecot/dovecot-auth</td>
<td>Location of the authentication executable</td>
</tr>
<tr>
<td><strong>auth_process_size</strong> = 256</td>
<td>Max authentication process size, in Mb</td>
</tr>
<tr>
<td><strong>auth_username_chars</strong> = abcd...VWXYZ01234567890.-@</td>
<td>List of allowed characters in the username. If the username entered by the user contains a character not listed in here, the login automatically fails. This is to prevent a user exploiting any potential quote-escaping vulnerabilities with SQL/LDAP databases</td>
</tr>
<tr>
<td><strong>auth_realms</strong> =</td>
<td>List of realms for SASL authentication mechanisms that need them. If empty, multiple realms are not supported</td>
</tr>
<tr>
<td><strong>auth_default_realm</strong> = example.org</td>
<td>Default realm/domain to use if none was specified</td>
</tr>
<tr>
<td><strong>auth_anonymous_username</strong> = anonymous</td>
<td>Username to assign to users logging in with ANONYMOUS SASL mechanism</td>
</tr>
<tr>
<td><strong>auth_verbose</strong> = no</td>
<td>Defines whether to log unsuccessful authentication attempts and the reasons why they failed</td>
</tr>
<tr>
<td><strong>auth_debug</strong> = no</td>
<td>Define whether to enable more verbose logging (e.g. SQL queries) for debugging purposes</td>
</tr>
<tr>
<td><strong>auth_failure_delay</strong> = 2</td>
<td>Delay before replying to failed authentications, in seconds</td>
</tr>
<tr>
<td><strong>auth default {</strong></td>
<td>Accepted authentication mechanisms</td>
</tr>
<tr>
<td>mechanisms = plain login cram-md5</td>
<td>Deny login to the users listed in /etc/dovecot.deny (this file contains one user per line)</td>
</tr>
<tr>
<td>passdb passwd-file {</td>
<td>PAM authentication block. Enables authentication matching (username and remote IP address) for PAM</td>
</tr>
<tr>
<td>args = /etc/dovecot.deny</td>
<td>System users e.g. NSS or /etc/passwd</td>
</tr>
<tr>
<td>denial = yes</td>
<td>Shadow passwords for system users, e.g. NSS or /etc/passwd</td>
</tr>
<tr>
<td>}</td>
<td>PAM-like authentication for OpenBSD</td>
</tr>
<tr>
<td>passdb pam {</td>
<td>SQL database</td>
</tr>
<tr>
<td>args = cache_key=%u%r dovecot</td>
<td>LDAP database</td>
</tr>
<tr>
<td>}</td>
<td>Export the authentication interface to other programs. Master socket provides access to userdb information, and is typically used to give Dovecot’s local delivery agent access to userdb so it can find mailbox locations. The default user/group is the one who started dovecot-auth (i.e. root). The client socket is generally safe to export to everyone. Typical use is to export it to the SMTP server so it can do SMTP AUTH lookups using it</td>
</tr>
<tr>
<td>passdb passwd {</td>
<td></td>
</tr>
<tr>
<td>blocking = yes</td>
<td></td>
</tr>
<tr>
<td>args =</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>passdb shadow {</td>
<td></td>
</tr>
<tr>
<td>blocking = yes</td>
<td></td>
</tr>
<tr>
<td>args =</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>passdb bsdauth {</td>
<td></td>
</tr>
<tr>
<td>cache_key = %u</td>
<td></td>
</tr>
<tr>
<td>args =</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>passdb sql {</td>
<td></td>
</tr>
<tr>
<td>args = /etc/dovecot/dovecot-sql.conf</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>passdb ldap {</td>
<td></td>
</tr>
<tr>
<td>args = /etc/dovecot/dovecot-ldap.conf</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>socket listen {</td>
<td></td>
</tr>
<tr>
<td>master {</td>
<td></td>
</tr>
<tr>
<td>path = /var/run/dovecot/auth-master</td>
<td></td>
</tr>
<tr>
<td>mode = 0600</td>
<td></td>
</tr>
<tr>
<td>user =</td>
<td></td>
</tr>
<tr>
<td>group =</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>client {</td>
<td></td>
</tr>
<tr>
<td>path = /var/run/dovecot/auth-client</td>
<td></td>
</tr>
<tr>
<td>mode = 0660</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
</tbody>
</table>
FTP (File Transfer Protocol) is a client-server unencrypted protocol for file transfer. Secure alternatives are FTPS (FTP secured with SSL/TLS) and SFTP (SSH File Transfer Protocol). It can operate either in active or in passive mode.

### Active mode (default)
1. Client connects to FTP server on port 21 (control channel) and sends second unprivileged port number
2. Server acknowledges
3. Server connects from port 20 (data channel) to client’s second unprivileged port number
4. Client acknowledges

### Passive mode (more protocol-compliant, because it is the client that initiates the connection)
1. Client connects to FTP server on port 21 and requests passive mode via the PASV command
2. Server acknowledges and sends unprivileged port number via the PORT command
3. Client connects to server's unprivileged port number
4. Server acknowledges

<table>
<thead>
<tr>
<th>FTP servers</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Very Secure FTP</strong></td>
<td>Hardened and high-performance FTP implementation. The vsftpd daemon operates with multiple processes that run as a non-privileged user in a chrooted jail</td>
</tr>
<tr>
<td><strong>Pure-FTP</strong></td>
<td>Free and easy-to-use FTP server</td>
</tr>
<tr>
<td><code>pure-ftpd</code></td>
<td>Pure-FTP daemon</td>
</tr>
<tr>
<td><code>pure-ftpwho</code></td>
<td>Show clients connected to the Pure-FTP server</td>
</tr>
<tr>
<td><code>pure-mrtginfo</code></td>
<td>Show connections to the Pure-FTP server as a MRTG graph</td>
</tr>
<tr>
<td><code>pure-statsdecode</code></td>
<td>Show Pure-FTP log data</td>
</tr>
<tr>
<td><code>pure-pw</code></td>
<td>Manage Pure-FTP virtual accounts</td>
</tr>
<tr>
<td><code>pure-pwconvert</code></td>
<td>Convert the system user database to a Pure-FTP virtual accounts database</td>
</tr>
<tr>
<td><code>pure-quotacheck</code></td>
<td>Manage Pure-FTP quota database</td>
</tr>
<tr>
<td><code>pure-uploadscript</code></td>
<td>Run a command on the Pure-FTP server to process an uploaded file</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FTP clients</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ftp</strong></td>
<td>Standard FTP client</td>
</tr>
<tr>
<td><code>ftp ftpserver.domain.com</code></td>
<td>Connect to an FTP server</td>
</tr>
<tr>
<td><strong>lftp</strong></td>
<td>Sophisticated FTP client with support for HTTP and BitTorrent</td>
</tr>
<tr>
<td><code>lftp ftpserver.domain.com</code></td>
<td>Connect to an FTP server and try an anonymous login</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/etc/vsftpd/vsftpd.conf</td>
<td>Very Secure FTP server configuration file</td>
</tr>
<tr>
<td>listen=NO</td>
<td>Run vsftpd in standalone mode (i.e. not via inetd)?</td>
</tr>
<tr>
<td>local_enable=YES</td>
<td>Allow local system users (i.e. in /etc/passwd) to log in?</td>
</tr>
<tr>
<td>chroot_local_user=YES</td>
<td>Chroot local users in their home directory?</td>
</tr>
<tr>
<td>write_enable=YES</td>
<td>Allow FTP commands that write on the filesystem (i.e. STOR, DELE, RNFR, RNTO, MKD, RMD, APPE, and SITE)?</td>
</tr>
<tr>
<td>anonymous_enable=YES</td>
<td>Allow anonymous logins? If yes, anonymous and ftp are accepted as logins</td>
</tr>
<tr>
<td>anon_root=/var/ftp/pub</td>
<td>Directory to go after anonymous login</td>
</tr>
<tr>
<td>anon_upload_enable=YES</td>
<td>Allow anonymous uploads?</td>
</tr>
<tr>
<td>chown_uploads=YES</td>
<td>Change ownership of anonymously uploaded files?</td>
</tr>
<tr>
<td>chown_username=ftp</td>
<td>User to whom set ownership of anonymously uploaded files</td>
</tr>
<tr>
<td>anon_world_readable_only=NO</td>
<td>Allow anonymous users to only download world-readable files?</td>
</tr>
<tr>
<td>ssl_enable=YES</td>
<td>Enable SSL?</td>
</tr>
<tr>
<td>force_local_data_ssl=NO</td>
<td>Encrypt local data?</td>
</tr>
<tr>
<td>force_local_logins_ssl=YES</td>
<td>Force encrypted authentication?</td>
</tr>
<tr>
<td>allow_anon_ssl=YES</td>
<td>Allow anonymous users to use SSL?</td>
</tr>
<tr>
<td>ssl_tlsv1=YES</td>
<td>Allowed SSL/TLS versions</td>
</tr>
<tr>
<td>ssl_tlsv2=NO</td>
<td>Location of certificate file</td>
</tr>
<tr>
<td>ssl_tlsv3=NO</td>
<td>Location of private key file</td>
</tr>
</tbody>
</table>

rsa_cert_file=/etc/pki/tls/certs/vsftpd.pem
rsa_private_key_file=/etc/pki/tls/certs/vsftpd.pem
In Linux, printers are managed by `cupsd`, the CUPS (Common Unix Printing System) daemon. Printers are administered via a web interface on the URL `http://localhost:631`.

```
/etc/cups/cupsd.conf
/etc/cups/printers.conf
/etc/printcap
/var/spool/cups/
/var/log/cups/error_log
/var/log/cups/page_log

Start the CUPS service

gnome-cups-manager
cupsenable printer0
cupsdisable printer0
cupsaccept printer0
cupsreject -r "Message" printer0
cupstestppd LEXC510.ppd
cupsaddsmb printer0
cups-config --cflags

cups-config --datadir

cups-config --ldflags

cups-config --libs

cups-config --serverbin

cups-config --serverroot

cups-config --serverroot

lpstat
lpadmin
lpadmin -p printer0 -P LEXC750.ppd
lp -d printer0 file

lpq
lpq -P printer0
lpq user
lp -P printer0 jobnumber
lp -P printer0 user
lp -P printer0 -
lpc

a2ps file.txt
ps2pdf file.ps
mpage file.ps
gv file.ps
```

- `CUPS` configuration file
- Database of available local CUPS printers
- Database of printer capabilities, for old printing applications
- Printer spooler for data awaiting to be printed
- CUPS error log
- Information about printed pages
- `CUPS` configuration file
- Start the CUPS service
- Run the CUPS Manager graphical application
- Enable a CUPS printer
- Disable a CUPS printer
- Accept a job sent on a printer queue
- Reject a job sent on a printer queue, with an informational message
- Test the conformance of a PPD file to the format specification
- Export a printer to Samba (for use with Windows clients)
- Show the necessary compiler options
- Show the default CUPS data directory
- Show the necessary linker options
- Show the necessary libraries to link to
- Show the default CUPS binaries directory that stores filters and backends
- Show the default CUPS configuration file directory
- Show CUPS status information
- Administer CUPS printers
- Specify a PPD (Adobe PostScript Printer Description) file to associate to a printer
- Print a file on the specified printer
- View the default print queue
- View a specific print queue
- View the print queue of a specific user
- Delete a specific job from a printer queue
- Delete all jobs from a specific user from a printer queue
- Delete all jobs from a printer queue
- Manage print queues
- Convert a text file to PostScript
- Convert a file from PostScript to PDF
- Print a PostScript document on multiple pages per sheet on a PostScript printer
- View a PostScript document (the `gv` software is a derivation of GhostView)
An IPv4 address is 32-bit long, and is represented divided in four octets (dotted-quad notation), e.g. 193.22.33.44.

There are approximately $4 \times 10^{16}$ total possible IPv4 addresses.

IPv4 classful addressing is obsolete and has been replaced by CIDR (Classless Inter-Domain Routing).

An IPv6 address is 128-bit long, and is represented divided in eight 16-bit groups (4 hex digits). Leading zeros in each group can be deleted. A single chunk of one or more adjacent 0000 groups can be deleted. e.g. 2130:0000:0000:0007:0040:15bc:235f which can also be written as 2130::7:40:15bc:235f.

There are approximately $3 \times 10^{38}$ total possible IPv6 addresses.

The IANA (Internet Assigned Numbers Authority) manages the allocation of IPv4 and IPv6 addresses, assigning large blocks to RIRs (Regional Internet Registries) which in turn allocate addresses to ISPs (Internet Service Providers) and other local registries. These address blocks can be searched via a WHOIS query to the appropriate RIR, which is:

- AFRINIC for Africa
- ARIN for US, Canada, and Antarctica
- APNIC for Asia and Oceania
- LACNIC for Latin America
- RIPE NCC for Europe, Middle East, and Russia
### VLSM chart - Last octet subnetting (CIDR notation)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>.00</td>
<td>.128</td>
<td>.192</td>
<td>.224</td>
<td>.240</td>
<td>.248</td>
<td>.252</td>
</tr>
<tr>
<td>.01</td>
<td>.129</td>
<td>.193</td>
<td>.225</td>
<td>.241</td>
<td>.249</td>
<td>.253</td>
</tr>
<tr>
<td>.02</td>
<td>.130</td>
<td>.194</td>
<td>.226</td>
<td>.242</td>
<td>.250</td>
<td>.254</td>
</tr>
<tr>
<td>.03</td>
<td>.131</td>
<td>.195</td>
<td>.227</td>
<td>.243</td>
<td>.251</td>
<td>.255</td>
</tr>
<tr>
<td>.04</td>
<td>.132</td>
<td>.196</td>
<td>.228</td>
<td>.244</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.05</td>
<td>.133</td>
<td>.197</td>
<td>.229</td>
<td>.245</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.06</td>
<td>.134</td>
<td>.198</td>
<td>.230</td>
<td>.246</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.07</td>
<td>.135</td>
<td>.199</td>
<td>.231</td>
<td>.247</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.08</td>
<td>.136</td>
<td>.200</td>
<td>.232</td>
<td>.248</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.09</td>
<td>.137</td>
<td>.201</td>
<td>.233</td>
<td>.249</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.10</td>
<td>.138</td>
<td>.202</td>
<td>.234</td>
<td>.250</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.11</td>
<td>.139</td>
<td>.203</td>
<td>.235</td>
<td>.251</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.12</td>
<td>.140</td>
<td>.204</td>
<td>.236</td>
<td>.252</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.13</td>
<td>.141</td>
<td>.205</td>
<td>.237</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.14</td>
<td>.142</td>
<td>.206</td>
<td>.238</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.15</td>
<td>.143</td>
<td>.207</td>
<td>.239</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.16</td>
<td>.144</td>
<td>.208</td>
<td>.240</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.17</td>
<td>.145</td>
<td>.209</td>
<td>.241</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.18</td>
<td>.146</td>
<td>.210</td>
<td>.242</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.19</td>
<td>.147</td>
<td>.211</td>
<td>.243</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.20</td>
<td>.148</td>
<td>.212</td>
<td>.244</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.21</td>
<td>.149</td>
<td>.213</td>
<td>.245</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.22</td>
<td>.150</td>
<td>.214</td>
<td>.246</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.23</td>
<td>.151</td>
<td>.215</td>
<td>.247</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.24</td>
<td>.152</td>
<td>.216</td>
<td>.248</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.25</td>
<td>.153</td>
<td>.217</td>
<td>.249</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.26</td>
<td>.154</td>
<td>.218</td>
<td>.250</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.27</td>
<td>.155</td>
<td>.219</td>
<td>.251</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.28</td>
<td>.156</td>
<td>.220</td>
<td>.252</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.29</td>
<td>.157</td>
<td>.221</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.30</td>
<td>.158</td>
<td>.222</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.31</td>
<td>.159</td>
<td>.223</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.32</td>
<td>.160</td>
<td>.224</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.33</td>
<td>.161</td>
<td>.225</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.34</td>
<td>.162</td>
<td>.226</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.35</td>
<td>.163</td>
<td>.227</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.36</td>
<td>.164</td>
<td>.228</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.37</td>
<td>.165</td>
<td>.229</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.38</td>
<td>.166</td>
<td>.230</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.39</td>
<td>.167</td>
<td>.231</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.40</td>
<td>.168</td>
<td>.232</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.41</td>
<td>.169</td>
<td>.233</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.42</td>
<td>.170</td>
<td>.234</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.43</td>
<td>.171</td>
<td>.235</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.44</td>
<td>.172</td>
<td>.236</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.45</td>
<td>.173</td>
<td>.237</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.46</td>
<td>.174</td>
<td>.238</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.47</td>
<td>.175</td>
<td>.239</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.48</td>
<td>.176</td>
<td>.240</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.49</td>
<td>.177</td>
<td>.241</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.50</td>
<td>.178</td>
<td>.242</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.51</td>
<td>.179</td>
<td>.243</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.52</td>
<td>.180</td>
<td>.244</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.53</td>
<td>.181</td>
<td>.245</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.54</td>
<td>.182</td>
<td>.246</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.55</td>
<td>.183</td>
<td>.247</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.56</td>
<td>.184</td>
<td>.248</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.57</td>
<td>.185</td>
<td>.249</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.58</td>
<td>.186</td>
<td>.250</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.59</td>
<td>.187</td>
<td>.251</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.60</td>
<td>.188</td>
<td>.252</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.61</td>
<td>.189</td>
<td>.253</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.62</td>
<td>.190</td>
<td>.254</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.63</td>
<td>.191</td>
<td>.255</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Each block of a column identifies a subnet, whose range of valid hosts addresses is [network address +1 — broadcast address -1] inclusive.
The network address of the subnet is the number shown inside a block.
The broadcast address of the subnet is the network address of the block underneath -1 or, for the bottom block, .255.
### Most common well-known ports

<table>
<thead>
<tr>
<th>Port number</th>
<th>Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>TCP FTP (data)</td>
</tr>
<tr>
<td>21</td>
<td>TCP FTP (control)</td>
</tr>
<tr>
<td>22</td>
<td>TCP SSH</td>
</tr>
<tr>
<td>23</td>
<td>TCP Telnet</td>
</tr>
<tr>
<td>25</td>
<td>TCP SMTP</td>
</tr>
<tr>
<td>53</td>
<td>TCP/UDP DNS</td>
</tr>
<tr>
<td>67</td>
<td>UDP BOOTP/DHCP (server)</td>
</tr>
<tr>
<td>68</td>
<td>UDP BOOTP/DHCP (client)</td>
</tr>
<tr>
<td>80</td>
<td>TCP HTTP</td>
</tr>
<tr>
<td>110</td>
<td>TCP POP3</td>
</tr>
<tr>
<td>119</td>
<td>TCP NNTP</td>
</tr>
<tr>
<td>123</td>
<td>UDP NTP</td>
</tr>
<tr>
<td>139</td>
<td>TCP/UDP Microsoft NetBIOS</td>
</tr>
<tr>
<td>143</td>
<td>TCP IMAP</td>
</tr>
<tr>
<td>161</td>
<td>UDP SNMP</td>
</tr>
<tr>
<td>443</td>
<td>TCP HTTPS (HTTP over SSL/TLS)</td>
</tr>
<tr>
<td>465</td>
<td>TCP SMTP over SSL</td>
</tr>
<tr>
<td>993</td>
<td>TCP IMAPS (IMAP over SSL)</td>
</tr>
<tr>
<td>995</td>
<td>TCP POP3S (POP3 over SSL)</td>
</tr>
</tbody>
</table>

1-1023: privileged ports, used server-side  
1024-65535: unprivileged ports, used client-side

/etc/services lists all well-known ports.  
Many network services are run by the xinetd super server.

### ISO/OSI and TCP/IP protocol stack models

<table>
<thead>
<tr>
<th>Layer</th>
<th>ISO/OSI</th>
<th>TCP/IP</th>
<th>Standards</th>
<th>Data transmission unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Application</td>
<td>Application</td>
<td>HTTP, SMTP, POP ...</td>
<td>Message</td>
</tr>
<tr>
<td>6</td>
<td>Presentation</td>
<td>Transport</td>
<td>TCP, UDP</td>
<td>Segment (TCP), Datagram (UDP)</td>
</tr>
<tr>
<td>5</td>
<td>Session</td>
<td>Internet</td>
<td>IPv4, IPv6, ICMP ...</td>
<td>Packet</td>
</tr>
<tr>
<td>4</td>
<td>Transport</td>
<td>Network Access</td>
<td>Ethernet, Wi-Fi, PPP ...</td>
<td>Frame</td>
</tr>
<tr>
<td>3</td>
<td>Network</td>
<td></td>
<td></td>
<td>Bit</td>
</tr>
<tr>
<td>2</td>
<td>Data Link</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Physical</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Network configuration commands

- `ip a`  
  Display configuration of all network interfaces

- `ip addr`  
  Display configuration of `eth0`

- `ip addr add dev eth0 10.1.1.3/24`  
  Configure IP address of `eth0`

- `ip link show eth0`  
  Display configuration of `eth0`

- `ifconfig eth0 hw ether 45:67:89:ab:cd:ef`  
  Configure MAC address of `eth0`

- `ip link set eth0 up`  
  Activate `eth0`

- `ip link set eth0 down`  
  Shut down `eth0`

- `dhclient eth0`  
  Request an IP address via DHCP

- `pump -i eth0`  
  (SUSE)

- `dhcpcd eth0`  
  (SUSE)

- `ip neigh`  
  Show the ARP cache table (containing mappings of MAC to IP addresses)

- `arp -a`  
  Show the ARP cache entry for a host

- `ip neigh show 10.1.1.4`  
  Add a new ARP entry for a host

- `arp 10.1.1.4`  
  Delete an ARP entry

- `ip neigh add 10.1.1.5 lladdr 01:23:45:67:89:ab dev eth0`  
  Delete the ARP table for all interfaces

- `ip neigh del 10.1.1.5 dev eth0`  
  (RHEL 7)

- `ip neigh flush all`  
  (RHEL 7)

- `hostname`  
  Get the hostname

- `hostname -f`  
  Get the FQDN (Fully Qualified Domain Name)

- `hostnamectl set-hostname --static "mybox"`  
  Set the hostname

- `hostnamectl`  
  (RHEL 7)

- `/etc/init.d/networking restart`  
  Restart network services

- `/etc/init.d/network restart`  
  (Debian)

- `ethtool option device`  
  Query or control network driver and hardware settings

- `ethtool eth0`  
  View hardware settings of `eth0`
Network configuration files

/etc/hosts
Mappings between IP addresses and hostnames, for name resolution

127.0.0.1  localhost.localdomain  localhost
10.2.3.4   myhost.domain.org      myhost

/etc/nsswitch.conf
Sources that must be used by various system library lookup functions

passwd:  files nisplus nis
shadow:  files nisplus nis
group:   files nisplus nis
hosts:   files dns nisplus nis

/etc/host.conf
Sources for name resolution, for systems before glibc2.
Obsolete, superseded by /etc/nsswitch.conf

order hosts,bind
multi on

/etc/resolv.conf
Domain names that must be appended to bare hostnames, and DNS servers that will be used for name resolution

search domain1.org domain2.org
nameserver  192.168.3.3
nameserver  192.168.4.4

/etc/networks
Mappings between network addresses and names

loopback  127.0.0.0
mylan     10.2.3.0

/etc/services
List of service TCP/UDP port numbers

/etc/protocols
List of available protocols

/sys/class/net
List of all network interfaces in the system
### Red Hat network configuration

**Network configuration file** *(file: /etc/sysconfig/network)*

- `ADDRESS=10.2.3.4`
- `NETMASK=255.255.255.0`
- `GATEWAY=10.2.3.254`
- `HOSTNAME=mylinuxbox.example.org`
- `NETWORKING=yes`

**Configuration file for eth0.** *(file: /etc/sysconfig/network-scripts/ifcfg-eth0)*

- `DEVICE=eth0`
- `TYPE=Ethernet`
- `BOOTPROTO=none`
- `ONBOOT=yes`
- `NM_CONTROLLED=no`
- `IPADDR=10.2.3.4`
- `NETMASK=255.255.255.0`
- `GATEWAY=10.2.3.254`
- `DNS1=8.8.8.8`
- `DNS2=4.4.4.4`
- `USERCTL=no`

**Multiple configuration files for a single eth0 interface, which allows binding multiple IP addresses to a single NIC** *(files: /etc/sysconfig/network-scripts/ifcfg-eth0:0, /etc/sysconfig/network-scripts/ifcfg-eth0:1, /etc/sysconfig/network-scripts/ifcfg-eth0:2)*

**Static route configuration for eth0** *(file: /etc/sysconfig/network-scripts/route-eth0)*

- `default 10.2.3.4 dev eth0`
- `10.7.8.0/24 via 10.2.3.254 dev eth0`
- `10.7.9.0/24 via 10.2.3.254 dev eth0`

**Ethernet frame types.** *(file: /etc/ethertypes)*

**List and configuration of all network interfaces** *(file: /etc/network/interfaces)*

- `allow-hotplug eth0`
- `iface eth0 inet static`
  - `address 10.2.3.4`
  - `netmask 255.255.255.0`
  - `gateway 10.2.3.254`
  - `dns-domain example.com`
  - `dns-nameservers 8.8.8.8 4.4.4.4`

**Hostname of the local machine** *(file: /etc/hostname)*

**ARP mappings** *(file: /etc/ethers)*
In RHEL7 the network configuration is managed by the NetworkManager daemon. A **connection** is a network configuration that applies to a **device** (aka network interface). A device can be included in multiple connections, but only one of them may be active at a time. The configuration for **connection** is stored in the file `/etc/sysconfig/network-scripts/ifcfg-connection`. Although it is possible to set up networking by editing these configuration files, it is much easier to use the command `nmcli`.

- **nmcli device status**
  - Show all network devices

- **nmcli device disconnect iface**
  - Disconnects the device `iface`.
  - This command should be used instead of `nmcli connection down` because if **connection** is set to autoconnect, Network Manager will bring the connection (and the device) up again short time later.

- **nmcli connection show**
  - Show all connections.
  - Connections with an empty device entry are inactive

- **nmcli connection show --active**
  - Show active connections

- **nmcli connection show connection**
  - Show the configuration of **connection**

- **nmcli connection add con-name connection \ type ethernet ifname iface ipv4.method manual \ ipv4.addresses 10.0.0.13/24 ipv4.gateway 10.0.0.254**
  - Configure a new **connection** that uses the Ethernet interface `iface` and assigns it an IPv4 address and gateway

- **nmcli connection modify connection options**
  - Modify the configuration of **connection**

- **nmcli connection up connection**
  - Brings up a **connection**

- **nmcli connection reload**
  - Reload any manual change made to the files `/etc/sysconfig/network-scripts/ifcfg-*`

*The manpage* `man nmcli-examples` contains many examples of network configuration.
Teaming and bridging

Network teaming allows binding together two or more network interfaces to increase throughput or provide redundancy. RHEL7 implements network teaming via the teamd daemon.

How to set up a teaming connection

1. `nmcli connection add type team con-name teamcon ifname teamif \ config '{"runner":{"name":"loadbalance"}}'` Set up a team connection `teamcon` and a team interface `teamif` with a runner (in JSON code) for automatic failover

2. `nmcli connection modify teamcon ipv4.method manual \ ipv4.addresses 10.0.0.14/24 ipv4.gateway 10.0.0.254` Assign manually an IP address and gateway

3. `nmcli connection add type team-slave ifname iface \ master teamcon` Add an existing device `iface` as a slave of team `teamcon`. The slave connection will be automatically named `team-slave-iface`

4. Repeat the previous step for each slave interface.

    teamdctl teamif state Show the state of the team interface `teamif`
    teamnl teamif command Debug a team interface `teamif`

A network bridge emulates a hardware bridge, i.e. a Layer 2 device able to forward traffic between networks based on MAC addresses.

How to set up a bridge connection

1. `nmcli connection add type bridge con-name brcon ifname brif` Set up a bridge connection `brcon` and a bridge interface `brif`

2. `nmcli connection modify brcon ipv4.method manual \ ipv4.addresses 10.0.0.15/24 ipv4.gateway 10.0.0.254` Assign manually an IP address and gateway

3. `nmcli connection add type bridge-slave ifname iface \ master brcon` Add an existing device `iface` as a slave of bridge `brcon`. The slave connection will be automatically named `bridge-slave-iface`

4. Repeat the previous step for each slave interface.

    brctl show brif Display information about the bridge interface `brif`

The manpage `man teamd.conf` lists many examples of team configurations and runners. The manpage `man nmcli-examples` contains, among others, examples of teaming and bridging configuration.
Wireless networking

iwlist wlan0 scan
List all wireless devices in range, with their quality of signal and other information

iwlist wlan0 freq
Display transmission frequency settings

iwlist wlan0 rate
Display transmission speed settings

iwlist wlan0 txpower
Display transmission power settings

iwlist wlan0 key
Display encryption settings

iwgetid wlan0 option
Print NWID, ESSID, AP/Cell address or other information about the wireless network that is currently in use

iwconfig wlan0
Display configuration of wireless interface wlan0

iwconfig wlan0 option
Configure wireless interface wlan0

iw dev wlan0 station dump
On a wireless card configured in AP Mode, display information (e.g. MAC address, tx/rx, bitrate, signal strength) about the clients

rfkill list
List installed wireless devices

rfkill unblock n
Enable wireless device number n

hcidump -i device
Display raw HCI (Host Controller Interface) data exchanged with a Bluetooth device
Network tools

- **dig example.org**
  - Perform a DNS lookup for the specified domain or hostname.
  - Returns information in BIND zone file syntax; uses an internal resolver and hence does not honor `/etc/resolv.conf`

- **host example.org**
  - Perform a DNS lookup for the specified domain or hostname.
  - Does honor `/etc/resolv.conf`

- **nslookup example.org** (deprecated)

- **dig @nameserver -t MX example.org**
  - Perform a DNS lookup for the MX record of the specified domain, querying `nameserver`

- **dig example.org any**
  - Get all DNS records for a domain

- **dig -x a.b.c.d**
  - Perform a reverse DNS lookup for the IP address `a.b.c.d`

- **whois example.org**
  - Query the WHOIS service for an Internet resource, usually a domain name

- **ping host**
  - Test if a remote host can be reached and measure the round-trip time to it. This is done by sending an ICMP Echo Request datagram and awaiting an ICMP Echo Response

- **fping -a host1 host2 host3**
  - Ping multiple hosts in parallel and report which ones are alive

- **bing host1 host2**
  - Calculate point-to-point throughput between two hosts

- **traceroute host**
  - Print the route, hop by hop, packets trace to a remote host. This is done by sending a sequence of ICMP Echo Request datagrams with increasing TTL values, starting with TTL=1, and expecting ICMP Time Exceeded datagrams

- **tracepath host**
  - Simpler `traceroute`

- **mtr host**
  - `traceroute` and `ping` combined

- **redir --laddr=ip1 --lport=port1 \\ --caddr=ip2 --cport=port2**
  - Redirect all connections coming to local IP address `ip1` and port `port1`, to remote IP address `ip2` and port `port2`

- **telnet host port**
  - Establish a telnet connection to the specified host and port number. If port is omitted, uses default port 23

- **wget --no-clobber --html-extension \\ --page-requisites --convert-links \\ --recursive --domains example.org \\ --no-parent www.example.org/path**
  - Download a whole website `www.example.org/path`

- **curl www.example.org/file.html -o myfile.html**
  - Download a file via HTTP and save it locally under another name

- **curl -u user:password 'ftp://ftpserver/path/file'**
  - Download a file via FTP, after logging in to the server

- **curl -XPUT webserver -d 'data'**
  - Send an HTTP PUT command with `data` to `webserver`
## Network monitoring

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>netstat</code></td>
<td>Display network connections</td>
</tr>
<tr>
<td><code>netstat --tcp</code></td>
<td>Display active TCP connections</td>
</tr>
<tr>
<td><code>netstat -t</code></td>
<td>Display only listening sockets</td>
</tr>
<tr>
<td><code>netstat -l</code></td>
<td>Display all listening and non-listening sockets</td>
</tr>
<tr>
<td><code>netstat -a</code></td>
<td>Display network connections, without resolving hostnames or portnames</td>
</tr>
<tr>
<td><code>netstat -p</code></td>
<td>Display network connections, with PID and name of program to which each socket belongs</td>
</tr>
<tr>
<td><code>netstat -i</code></td>
<td>Display network interfaces</td>
</tr>
<tr>
<td><code>netstat -s</code></td>
<td>Display protocol statistics</td>
</tr>
<tr>
<td><code>netstat -r</code></td>
<td>Display kernel routing tables (equivalent to <code>route -e</code>)</td>
</tr>
<tr>
<td><code>netstat -c</code></td>
<td>Display network connections continuously</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ss</code></td>
<td>Display socket statistics (similarly to <code>netstat</code>)</td>
</tr>
<tr>
<td><code>ss -t -a</code></td>
<td>Display all TCP sockets</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>nmap host</code></td>
<td>Scan for open TCP ports (TCP SYN scan) on remote host</td>
</tr>
<tr>
<td><code>nmap -sS host</code></td>
<td>Do a ping sweep (ICMP ECHO probes) on remote host</td>
</tr>
<tr>
<td><code>nmap -sP host</code></td>
<td>Scan for open UDP ports on remote host</td>
</tr>
<tr>
<td><code>nmap -sV host</code></td>
<td>Do a service and version scan on open ports</td>
</tr>
<tr>
<td><code>nmap -p 1-65535 host</code></td>
<td>Scan all ports (1-65535), not only the common ports, on remote host</td>
</tr>
<tr>
<td><code>nmap -O host</code></td>
<td>Find which operating system is running on remote host (OS fingerprinting)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>arp-scan</code></td>
<td>Scan all hosts on the current LAN. Uses ARP (Layer 2) packets and is therefore able to find even the hosts configured to drop all IP or ICMP traffic; for the same reason cannot scan hosts outside the same LAN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ngrep</code></td>
<td>Filter data payload of network packets matching a specified regex</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dhcpdump -i eth0</code></td>
<td>Sniff all DHCP packets on interface <code>eth0</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>nload</code></td>
<td>Display a graph of the current network usage</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>iptraf</code></td>
<td>IP LAN monitor (ncurses UI)</td>
</tr>
<tr>
<td><code>iptraf-ng</code></td>
<td>IP LAN monitor (ncurses UI)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>netserver</code></td>
<td>Run a network performance benchmark server</td>
</tr>
<tr>
<td><code>netperf</code></td>
<td>Do network performance benchmarks by connecting to a netserver</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>iperf -s</code></td>
<td>Run a network throughput benchmark server</td>
</tr>
<tr>
<td><code>iperf -c server</code></td>
<td>Perform network throughput tests in client mode, by connecting to an iperf server</td>
</tr>
</tbody>
</table>
Tcpdump is a packet analyzer (aka packet sniffer). A GUI equivalent is Wireshark, previously called Ethereal.

- `tcpdump -ni eth0`  
  Sniff all network traffic on interface eth0, suppressing DNS resolution
- `tcpdump ip host 10.0.0.2 tcp port 25`  
  Sniff network packets on TCP port 25 from and to 10.0.0.2
- `tcpdump ether host '45:67:89:ab:cd:ef'`  
  Sniff traffic from and to the network interface having MAC address 45:67:89:ab:cd:ef
- `tcpdump 'src host 10.0.0.2 and (tcp port 80 or tcp port 443)'`  
  Sniff HTTP and HTTPS traffic having as source host 10.0.0.2
- `tcpdump -ni eth0 not port 22`  
  Sniff all traffic on eth0 except that belonging to the SSH connection
- `tcpdump -vvnn -i eth0 arp`  
  Sniff ARP traffic on eth0, on maximum verbosity level, without converting host IP addresses and port numbers to names
- `tcpdump ip host 10.0.0.2 and not 10.0.0.9`  
  Sniff IP traffic between 10.0.0.2 and any other host except 10.0.0.9
Netcat is "the Swiss Army knife of networking", a very flexible generic TCP/IP client/server. Depending on the distribution, the binary is called `nc`, `ncat` (Red Hat), or `netcat` (SUSE).

```bash
nc -z 10.0.0.7 22
ncat 10.0.0.7 22
```

Scan for a listening SSH daemon on remote host 10.0.0.7

```bash
nc -l -p 25
```

Listen for connections on port 25 (i.e. mimic a SMTP server). Send any input received on stdin to the connected client and dump on stdout any data received from the client

```bash
nc 10.0.0.7 389 < file
```

Push the content of `file` to port 389 on remote host 10.0.0.7

```bash
echo "GET / HTTP/1.0\r\n\r\n" | nc 10.0.0.7 80
```

Connect to web server 10.0.0.7 and issue a HTTP GET

```bash
while true; do
  do nc -l -p 80 -q 1 < page.html; done
while true; do
  do echo "<html><body>Hello</body></html>" \
    | ncat -l -p 80; done
```

Start a minimal web server, serving the specified HTML page to any connected client

```bash
nc -v -n -z -w1 -r 10.0.0.7 1-1023
```

Run a TCP port scan against remote host 10.0.0.7. Probes randomly all privileged ports with a 1-second timeout, without resolving service names, and with verbose output

```bash
echo "" | nc -v -n -w1 10.0.0.7 1-1023
```

Retrieve the greeting banner of any network service that might be running on remote host 10.0.0.7
Host access control files used by the TCP Wrapper system.

Each file contains zero or more `daemon:client` lines. The first matching line is considered.

Access is granted when a `daemon:client` pair matches an entry in `/etc/hosts.allow`. Otherwise, access is denied when a `daemon:client` pair matches an entry in `/etc/hosts.deny`. Otherwise, access is granted.

<table>
<thead>
<tr>
<th><code>etc/hosts.allow</code> and <code>etc/hosts.deny</code> lines syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL: ALL</td>
<td>All services to all hosts</td>
</tr>
<tr>
<td>ALL: .example.edu</td>
<td>All services to all hosts of the example.edu domain</td>
</tr>
<tr>
<td>ALL: .example.edu EXCEPT host1.example.edu</td>
<td>All services to all hosts of example.edu, except host1</td>
</tr>
<tr>
<td>in.fingerd: .example.com</td>
<td>Finger service to all hosts of example.com</td>
</tr>
<tr>
<td>in.tftpd: LOCAL</td>
<td>TFTP to hosts of the local domain only</td>
</tr>
<tr>
<td>sshd: 10.0.0.3 10.0.0.4 10.1.1.0/24</td>
<td>SSH to the hosts and network specified</td>
</tr>
<tr>
<td>sshd: 10.0.1.0/24</td>
<td>SSH to 10.0.1.0/24</td>
</tr>
<tr>
<td>sshd: 10.0.1.1</td>
<td></td>
</tr>
<tr>
<td>sshd: 10.0.1.0/255.255.255.0</td>
<td></td>
</tr>
<tr>
<td>in.tftpd: ALL: spawn (/safe_dir/safe_finger -l @%h</td>
<td>Send a finger probe to hosts attempting TFTP and notify root user via email</td>
</tr>
<tr>
<td></td>
<td>/bin/mail -s %d-%h root) &amp;</td>
</tr>
<tr>
<td>portmap: ALL: (echo Illegal RPC request \</td>
<td>When a client attempts a RPC request via the portmapper (NFS access), echo a message to the terminal and notify the root user via email</td>
</tr>
<tr>
<td></td>
<td>from %h</td>
</tr>
</tbody>
</table>
### Output of command `route -en`

<table>
<thead>
<tr>
<th>Destination</th>
<th>Genmask</th>
<th>Flags</th>
<th>Metric</th>
<th>Ref</th>
<th>Use</th>
<th>Iface</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.3.0</td>
<td>0.0.0.0</td>
<td>U</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>eth0</td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>192.168.3.1</td>
<td>UG</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>eth0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Destination</th>
<th>network or host</th>
<th>gateway</th>
<th>Genmask</th>
<th>Flags</th>
<th>Metric</th>
<th>Ref</th>
<th>Use</th>
<th>Iface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>network mask</td>
<td>host</td>
<td>genmask</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>255.255.255.255</td>
<td>0.0.0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flags</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>route is up</td>
<td>route is up</td>
<td>route is up</td>
</tr>
<tr>
<td>G</td>
<td>use gateway</td>
<td>use gateway</td>
<td>use gateway</td>
</tr>
<tr>
<td>H</td>
<td>target is host</td>
<td>target is host</td>
<td>target is host</td>
</tr>
<tr>
<td>!</td>
<td>rejected route</td>
<td>rejected route</td>
<td>rejected route</td>
</tr>
<tr>
<td>D</td>
<td>dynamically installed by daemon</td>
<td>dynamically installed by daemon</td>
<td>dynamically installed by daemon</td>
</tr>
<tr>
<td>M</td>
<td>modified from routing daemon</td>
<td>modified from routing daemon</td>
<td>modified from routing daemon</td>
</tr>
<tr>
<td>R</td>
<td>reinstate route for dynamic routing</td>
<td>reinstate route for dynamic routing</td>
<td>reinstate route for dynamic routing</td>
</tr>
</tbody>
</table>

**ip route**

- `route -en`
- `route -F`
- `netstat -rn`

**ip route show cache**

- `route -C`

**ip route add default via 10.1.1.254**

- `route add default gw 10.1.1.254`

**ip route add 10.2.0.1 dev eth0**

- `route add 10.2.0.1 via 10.2.0.254`

**ip route add -host 10.2.0.1 gw 10.2.0.254**

**ip route add 10.2.0.0/16 via 10.2.0.254**

- `route add -net 10.2.0.0 netmask 255.255.0.0 gw 10.2.0.254`

**ip route delete 10.2.0.1 dev eth0**

- `route del -host 10.2.0.1 gw 10.2.0.254`

**ip route flush all**

- `ip route add default via 10.1.1.254`

- `Add a default gateway 10.1.1.254`

- `Add a route for a host 10.2.0.1`

- `Add a route for a network 10.2.0.0/16`

- `Delete a route for a host 10.2.0.1`

- `Delete the routing table for all interfaces`
The Netfilter framework provides firewalling capabilities in Linux. It is implemented by the user-space application programs `iptables` for IPv4 (which replaced `ipchains`, which itself replaced `ipfwadm`) and `ip6tables` for IPv6. `iptables` is implemented in the kernel and therefore does not have a daemon process or a service. The ability to track connection state is provided by the `ip_conntrack` kernel module.

In RHEL 7, `iptables` is replaced by the `firewalld` daemon. It is possible, but discouraged, to use `iptables` anyway by disabling `firewalld` and installing the package `iptables-services`, which provides a systemd interface for `iptables`. In Ubuntu, `iptables` is managed by the `ufw` (Uncomplicated Firewall) service.

```
/etc/sysconfig/iptables
```

Default file containing the firewall rules

```
iptables-restore < file
iptables-save > file
```

Load into `iptables` the firewall rules specified in the `file`

Save into `iptables` the firewall rules specified in the `file`

<table>
<thead>
<tr>
<th>iptables rules file</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>filter</code></td>
</tr>
<tr>
<td><code>INPUT ACCEPT [0:0]</code></td>
</tr>
<tr>
<td><code>FORWARD ACCEPT [0:0]</code></td>
</tr>
<tr>
<td><code>OUTPUT ACCEPT [0:0]</code></td>
</tr>
<tr>
<td>COMMIT</td>
</tr>
</tbody>
</table>

Delete all rules and open the firewall to all connections
Iptables uses **tables** containing sets of **chains**, which contain sets of **rules**. Each rule has a **target** (e.g. ACCEPT). The "filter" table contains chains INPUT, FORWARD, OUTPUT (built-in chains); this is the default table to which all iptables commands are applied, unless another table is specified via the `-t` option. The "nat" table contains chains PREROUTING, OUTPUT, POSTROUTING. The "mangle" table contains chains PREROUTING, OUTPUT.

When a packet enters the system, it is handed to the INPUT chain. If the destination is local, it is processed; if the destination is not local and IP forwarding is enabled, the packet is handed to the FORWARD chain, otherwise it is dropped. An outgoing packet generated by the system will go through the OUTPUT chain.

If NAT is in use, an incoming packet will pass at first through the PREROUTING chain, and an outgoing packet will pass last through the POSTROUTING chain.

```
iptables -A INPUT -s 10.0.0.6 -j ACCEPT
iptables -A INPUT -s 10.0.0.7 -j REJECT
iptables -A INPUT -s 10.0.0.8 -j DROP
iptables -A INPUT -s 10.0.0.9 -j LOG
iptables -D INPUT -s 10.0.0.9 -j LOG
iptables -D INPUT 42
iptables -F INPUT
iptables -F
```

Add a rule to accept all packets from 10.0.0.6
Add a rule to reject all packets from 10.0.0.7 and send back a ICMP response to the sender
Add a rule to silently drop all packets from 10.0.0.8
Add a rule to log (via syslog) all packets from 10.0.0.9
Delete a specific rule
Delete rule 42 of the INPUT chain
Flush all rules of the INPUT chain
Flush all rules, hence disabling the firewall
Delete all user-defined (not built-in) rules in the "mangle" table

```
iptables -L INPUT
iptables -L -n
iptables -N mychain
iptables -P INPUT DROP
```

List the rules of the INPUT chain
List all rules, without translating numeric values (IP addresses to FQDNs and port numbers to services)
Define a new chain
Define the chain policy target, which takes effect when no rule matches and the end of the rules list is reached

```
iptables -A OUTPUT -d 10.7.7.0/24 -j DROP
iptables -A FORWARD -i eth0 -o eth1 -j DROP
iptables -A INPUT -p 17 -j DROP
iptables -A INPUT -p udp -j DROP
iptables -A INPUT --sport 1024:65535 --dport 53 \ -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-request \ -m limit --limit 1/s -i eth0 -j ACCEPT
iptables -A INPUT --state ESTABLISHED \ -j ACCEPT
iptables -A INPUT --state NEW -j ACCEPT
iptables -A INPUT --state RELATED -j ACCEPT
iptables -A INPUT --state INVALID -j ACCEPT
```

Add a rule to drop all packets with destination 10.7.7.0/24
Add a rule to log all packets entering the system via eth0 and exiting via eth1
Add a rule to drop all incoming UDP traffic (protocol numbers are defined in /etc/protocols)
Add a rule to accept all packets coming from any unprivileged port and with destination port 53
Add a rule to accept incoming pings through eth0 at a maximum rate of 1 ping/second
Load the module for stateful packet filtering, and add a rule to accept all packets that are part of a communication already tracked by the state module
Add a rule to accept all packets that are not part of a communication already tracked by the state module
Add a rule to accept all packets that are related (e.g. ICMP responses to TCP or UDP traffic) to a communication already tracked by the state module
Add a rule to accept all packets that do not match any of the states above
**iptables NAT routing**

**SNAT (Source Network Address Translation)**

- `iptables -t nat -A POSTROUTING -s 10.0.0.0/24 -o eth1 -j SNAT --to-source 93.184.216.119`
  
  Map all traffic leaving the LAN to the external IP address 93.184.216.119

- `iptables -t nat -A POSTROUTING -s 10.0.0.0/24 -o eth1 -j SNAT --to-source 93.184.216.119:93.184.216.127`
  
  Map all traffic leaving the LAN to a pool of external IP addresses 93.184.216.119-127

- `iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE`
  
  Map all traffic leaving the LAN to the address dynamically assigned to eth1 via DHCP

**DNAT (Destination Network Address Translation)**

- `iptables -t nat -A PREROUTING -i eth1 -d 93.184.216.119 -j DNAT --to-destination 10.0.0.13`
  
  Allow the internal host 10.0.0.13 to be publicly reachable via the external address 93.184.216.119

**PAT (Port Address Translation)**

- `iptables -t nat -A PREROUTING -i eth1 -d 93.184.216.119 -p tcp --dport 80 -j DNAT --to-destination 10.0.0.13:8080`
  
  Make publicly accessible a webserver that is located in the LAN, by mapping port 8080 of the internal host 10.0.0.13 to port 80 of the external address 93.184.216.119

- `iptables -t nat -A PREROUTING -i eth0 -d ! 10.0.0.0/24 -p tcp --dport 80 -j REDIRECT --to-ports 3128`
  
  Redirect all outbound HTTP traffic originating from the LAN to a proxy running on port 3128 on the Linux box

**IP Forwarding**

- `sysctl -w net.ipv4.ip_forward=1`
- `echo 1 > /proc/sys/net/ipv4/ip_forward`

Enable IP forwarding; necessary to set up a Linux machine as a router. (This command causes other network options to be changed as well.)
In firewalld, a network interface (aka interface) or a subnet address (aka source) can be assigned to a specific zone. To determine to which zone a packet belongs, first the zone of the source is analyzed, then the zone of the interface; if no source or interface matches, the packet is associated to the default zone (which is "public", unless set otherwise). If the zone is not specified (via --zone=zone), the command is applied to the default zone.

By default, commands are temporary; adding the --permanent option to a command sets it as permanent, or shows permanent settings only.

Temporary commands are effective immediately but are canceled at reboot, firewall reload, or firewall restart. Permanent commands are effective only after reboot, firewall reload, or firewall restart.

### Firewalld zones (as obtained by firewall-cmd --get-zones)

<table>
<thead>
<tr>
<th>Zone</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>block</td>
<td>Rejects incoming connections with an ICMP HOST_PROHIBITED; allows only established connections</td>
</tr>
<tr>
<td>dmz</td>
<td>Used to expose services to the public; allows only specific incoming connections</td>
</tr>
<tr>
<td>drop</td>
<td>Drops all incoming packets; allows only outgoing connections</td>
</tr>
<tr>
<td>external</td>
<td>Used for routing and masquerading; allows only specific connections</td>
</tr>
<tr>
<td>home</td>
<td>Allows only specific incoming connections</td>
</tr>
<tr>
<td>internal</td>
<td>Used to define internal networks and allow only private network traffic</td>
</tr>
<tr>
<td>public</td>
<td>Allows only specific incoming connections. Default zone</td>
</tr>
<tr>
<td>trusted</td>
<td>Accepts all traffic</td>
</tr>
<tr>
<td>work</td>
<td>Used to define internal networks and allow only private network traffic</td>
</tr>
</tbody>
</table>

### System commands

- **systemctl status firewalld**
  - Check the status of the firewall
- **firewall-config**
  - Firewall management GUI
- **firewall-cmd --state**
- **firewall-cmd --reload**
  - Reload firewall configuration; this applies all permanent changes and cancels all temporary changes. Current connections are not terminated
- **firewall-cmd --complete-reload**
  - Reload firewall configuration, stopping all current connections
- **firewall-cmd --runtime-to-permanent**
  - Transform all temporary changes to permanent
- **firewall-cmd --list-all-zones**
  - List all zones and their full settings
- **firewall-cmd --get-default-zone**
  - Show the default zone
- **firewall-cmd --set-default-zone=home**
  - Set "home" as the default zone
- **firewall-cmd --get-active-zones**
  - Show the active zones i.e. zones bound to either an interface or a source
- **firewall-cmd --get-zones**
  - Show all available zones
- **firewall-cmd --get-zone-of-interface=eth0**
  - Show the zone assigned to eth0
- **firewall-cmd --new-zone=test**
  - Create a new zone called "test"
- **firewall-cmd --zone=home --change-interface=eth0**
  - Assign eth0 to the "home" zone
- **firewall-cmd --zone=home --list-all**
  - List temporary settings of the "home" zone
- **firewall-cmd --zone=home --list-all --permanent**
  - List permanent settings of the "home" zone
- **firewall-cmd --zone=home --add-source=10.1.1.0/24**
  - Assign 10.1.1.0/24 to the "home" zone i.e. route all traffic from that subnet to that zone
- **firewall-cmd --zone=home --list-sources**
  - List sources bound to the "home" zone
firewall-cmd --zone=trusted --add-service=ssh
firewall-cmd --zone=trusted --add-port=22/tcp
firewall-cmd --zone=trusted --add-service={ssh,http,https}
firewall-cmd --zone=trusted --list-services
firewall-cmd --zone=trusted --list-ports
firewall-cmd --get-services
firewall-cmd --get-icmptypes
firewall-cmd --add-icmp-block=echo-reply
firewall-cmd --query-icmp-block=echo-reply
firewall-cmd --list-icmp-block
firewall-cmd --add-rich-rule='richrule'
firewall-cmd --add-rich-rule='rule \
family=ipv4 source address=10.2.2.0/24 service name=tftp
log prefix=tftp level=info limit value=3/m accept'
firewall-cmd --list-rich-rules
firewall-cmd --zone=zone --add-masquerade
firewall-cmd --zone=zone --add-rich-rule='rule \
family=ipv4 source address=10.2.2.0/24 masquerade'
firewall-cmd --zone=zone --add-forward-port=
port=22:proto/tcp:toport=2222:toaddr=10.7.7.7

Add the SSH service to the "trusted" zone
Add the SSH, HTTP, and HTTPS services to the "trusted" zone
Show temporary and permanent services bound to the "trusted" zone
Show temporary and permanent ports open on the "trusted" zone
List all predefined services
Show all known types of ICMP messages
Block a specific ICMP message type
Tell if a specific ICMP message type is blocked
Show the list of blocked ICMP message types
Set up a rich rule (for more complex and detailed firewall configurations)
Set up a direct rule to allow tftp connections from subnet 10.2.2.0/24 and log them via syslog at a rate of 3 per minute
List all rich rules
Set up a direct rule (in iptables format)
Set up a direct rule to allow SSH connections
Set up a direct rule when firewalld is not running
Show all direct rules
Set up masquerading for hosts of zone; packets originating from zone will get the firewall's IP address on the "external" zone as source address
Set up masquerading only for those hosts of zone located in subnet 10.2.2.0/24
Set up port forwarding for hosts of zone; incoming connections to port 22 for hosts of zone will be forwarded to port 2222 on host 10.7.7.7

Predefined services are configured in /usr/lib/firewalld/services/service.xml.
User-defined services are configured in /etc/firewalld/services/service.xml.

User-defined direct rules are stored in /etc/firewalld/direct.xml.

The manpage man firewalld.richlanguage contains several examples of rich rules.
The manpage man firewalld.direct documents the syntax of direct rules.
SSH

```
ssh user@host
```

Connect to a remote host via SSH (Secure Shell) and login as user.
Options:
- `-v` `-vv` `-vvv` Increasing levels of verbosity
- `-p n` Use port n instead of standard port 22

```
ssh user@host /path/to/command
```

Execute a command on a remote host

```
sftp user@host
```

FTP-like tool for secure file transfer

```
scp /path1/file user@host:/path2/
scp user@host:/path1/file /path2/
scp user1@host1:/path1/file user2@host2:/path2/
```

Non-interactive secure file copy.
Can transfer files from local to remote, from remote to local, or between two remote hosts

```
autosh user@host
```

Connect to a remote host, monitoring the connection and restarting it automatically if it dies

```
sspass -p password ssh user@host
```

Connect to a remote host using the specified password

```
pssh -i -H "host1 host2 host3" /path/to/command
```

Execute a command in parallel on a group of remote hosts

```
ssh-keygen -t rsa -b 2048
```

Generate interactively a 2048-bit RSA key pair; will prompt for a passphrase

```
ssh-keygen -t dsa
```

Generate a DSA key pair

```
ssh-keygen -p -t rsa
```

Change passphrase of the private key

```
ssh-keygen -q -t rsa -f /path/to/keyfile -N '' -C ''
```

Generate a RSA key with no passphrase (for non-interactive use) and no comment

```
ssh-keygen -lf /path/to/keyfile.pub
```

View key length and fingerprint of a public key

```
ssh-agent
```

Echo to the terminal the environment variables that must be set in order to use the SSH Agent

```
eval `ssh-agent`
```

Start the SSH Agent daemon that caches decrypted private keys in memory; also shows the PID of ssh-agent and sets the appropriate environment variables. Once ssh-agent is started, the keys to cache must be added via the ssh-add command; cached keys will then be automatically used by any SSH tool e.g. ssh, sftp, scp

```
ssh-agent bash -c 'ssh-add /path/to/keyfile'
```

Start ssh-agent and cache the specified key

```
ssh-add
```

Add the default private keys to the ssh-agent cache

```
ssh-add /path/to/keyfile
```

Add a specific private key to the ssh-agent cache

```
ssh-copy-id user@host
```

Use locally available keys to authorize, via public key authentication, login of user on a remote host. This is done by copying the user’s local public key
`~/.ssh/id_rsa.pub` to `~/.ssh/authorized_keys` on the remote host
### SSH operations

#### SSH port forwarding (aka SSH tunneling)

```bash
ssh -L 2525:mail.foo.com:25 user@mail.foo.com
```

Establish a SSH encrypted tunnel from localhost to remote host mail.foo.com, redirecting traffic from local port 2525 to port 25 of remote host mail.foo.com. Useful if the local firewall blocks outgoing port 25. In this case, port 2525 is used to go out; the application must be configured to connect to localhost on port 2525 (instead of mail.foo.com on port 25)

```bash
ssh -L 2525:mail.foo.com:25 user@login.foo.com
```

Establish a SSH encrypted tunnel from localhost to remote host login.foo.com. Remote host login.foo.com will then forward, unencrypted, all data received over the tunnel on port 2525 to remote host mail.foo.com on port 25

#### SSH reverse forwarding (aka SSH reverse tunneling)

```bash
ssh -R 2222:localhost:22 user@login.foo.com
```

Establish a SSH encrypted reverse tunnel from remote host login.foo.com back to localhost, redirecting traffic sent to port 2222 of remote host login.foo.com back towards local port 22. Useful if the local firewall blocks incoming connections so remote hosts cannot connect back to local machine. In this case, port 2222 of login.foo.com is opened for listening and connecting back to localhost on port 22; remote host login.foo.com is then able to connect to the local machine on port 2222 (redirected to local port 22)

#### SSH as a SOCKS proxy

```bash
ssh -D 33333 user@login.foo.com
```

The application supporting SOCKS must be configured to connect to localhost on port 33333. Data is tunneled from localhost to login.foo.com, then unencrypted to destination

#### X11 Forwarding

```bash
ssh -X user@login.foo.com
```

Enable the local display to execute locally a X application stored on a remote host login.foo.com

### How to enable public key authentication

1. On remote host, set PubkeyAuthentication yes in /etc/ssh/sshd_config
2. On local machine, do ssh-copy-id you@remotehost (or copy your public key to the remote host by hand)

### How to enable host-based authentication amongst a group of trusted hosts

1. On all hosts, set HostBasedAuthentication yes in /etc/ssh/sshd_config
2. On all hosts, create /etc/ssh/hosts.equiv and enter in this file all trusted hostnames
3. Connect via SSH manually from your machine on each host so that all hosts’ public keys go into ~/.ssh/known_hosts
4. Copy ~/.ssh/known_hosts from your machine to /etc/ssh/ssh_known_hosts on all hosts

### How to enable X11 Forwarding

1. On remote host 10.2.2.2, set X11Forwarding yes in /etc/ssh/sshd_config, and make sure that xauth is installed
2. On local host 10.1.1.1, type ssh -X 10.2.2.2, then run on remote host the graphical application e.g. xclock &

It is also possible to enable X11 Forwarding via telnet (but this is insecure and obsolete, and therefore not recommended):

1. On remote host 10.2.2.2, type export DISPLAY=10.1.1.1:0.0
2. On local host 10.1.1.1, type xhost +
3. On local host 10.1.1.1, type telnet 10.2.2.2, then run on remote host the graphical application e.g. xclock &
SSH configuration

/etc/ssh/sshd_config
SSH server daemon configuration file
/etc/ssh/ssh_config
SSH client global configuration file
/etc/ssh/ssh_host_key
Host’s private key (should be mode 0600)
/etc/ssh/ssh_host_key.pub
Host’s public key
/etc/ssh/hosts.equiv
Names of trusted hosts for host-based authentication
/etc/ssh/ssh_known_hosts
Database of host public keys that were previously accepted as legitimate
/etc/ssh/ssh_config
SSH client global configuration file
/etc/ssh/ssh_host_key
User’s RSA or DSA private key, as generated by ssh-keygen
/etc/ssh/ssh_host_key.pub
User’s RSA or DSA public key, as generated by ssh-keygen
/etc/ssh/hosts.equiv
Names of trusted hosts for host-based authentication
/etc/ssh/known_hosts
Host public keys that were previously accepted as legitimate by the user
/etc/ssh/known_hosts
Host public keys that were previously accepted as legitimate by the user
/etc/ssh/known_hosts
Host public keys that were previously accepted as legitimate by the user
/etc/ssh/ssh_known_hosts
Host public keys that were previously accepted as legitimate by the user

<table>
<thead>
<tr>
<th>/etc/ssh/sshd_config</th>
<th>SSH server configuration file</th>
</tr>
</thead>
<tbody>
<tr>
<td>PermitRootLogin yes</td>
<td>Control superuser login via SSH. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>yes Superuser can login</td>
</tr>
<tr>
<td></td>
<td>no Superuser cannot login</td>
</tr>
<tr>
<td></td>
<td>without-password Superuser cannot login with password</td>
</tr>
<tr>
<td></td>
<td>forced-commands-only Superuser can only run commands in SSH command line</td>
</tr>
<tr>
<td>AllowUsers jdoe ksmith</td>
<td>List of users that can/cannot login via SSH, or * for everybody</td>
</tr>
<tr>
<td>DenyUsers jhacker</td>
<td>List of groups whose members can/cannot login via SSH, or * for all groups</td>
</tr>
<tr>
<td>AllowGroups geeks</td>
<td>Permit authentication via login and password</td>
</tr>
<tr>
<td>DenyGroups *</td>
<td>Permit authentication via public key</td>
</tr>
<tr>
<td>PasswordAuthentication yes</td>
<td>Permit authentication based on trusted hosts</td>
</tr>
<tr>
<td>PubKeyAuthentication yes</td>
<td>Specify protocols supported by SSH. Value can be 1 or 2 or both</td>
</tr>
<tr>
<td>HostbasedAuthentication yes</td>
<td>Allow X11 Forwarding</td>
</tr>
<tr>
<td>Protocol 1,2</td>
<td>X11Forwarding yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>/etc/ssh/sshd_config and ~/.ssh/config</th>
<th>SSH client configuration file</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host *</td>
<td>List of hosts to which the following directives will apply, or * for all hosts</td>
</tr>
<tr>
<td>StrictHostKeyChecking yes</td>
<td>Ask before adding new host keys to the ~/.ssh/known_hosts file, and refuse to connect if the key for a known host has changed. This prevents MITM attacks</td>
</tr>
<tr>
<td>GSSAPIAuthentication yes</td>
<td>Support authentication using GSSAPI</td>
</tr>
<tr>
<td>ForwardX11Trusted yes</td>
<td>Allow remote X11 clients to fully access the original X11 display</td>
</tr>
<tr>
<td>IdentityFile ~/.ssh/id_rsa</td>
<td>User identity file for authentication. Default values are: ~/.ssh/identity for protocol version 1</td>
</tr>
<tr>
<td></td>
<td>~/.ssh/id_rsa and ~/.ssh/id_dsa for protocol version 2</td>
</tr>
</tbody>
</table>
The X.509 standard defines the format of public key certificates and other related files. It includes cryptographic standards and protocols such as SSL/TLS, PKCS7, PKCS12, and OCSP. The Public Key Infrastructure X.509 (PKIX) is described in RFC 5280.

### X.509 file formats

<table>
<thead>
<tr>
<th>DER</th>
<th>Binary-encoded certificate</th>
</tr>
</thead>
</table>
| PEM     | ASCII-armored Base64-encoded certificate, included between these two lines:  
|         | \---------------BEGIN X.509\_FILE\_TYPE------  
|         | \---------------END X.509\_FILE\_TYPE------  

DER and PEM are also used as file extensions for different types of files; see below.

### X.509 file type extensions

<table>
<thead>
<tr>
<th>CRT</th>
<th>Certificate or certificate chain</th>
</tr>
</thead>
<tbody>
<tr>
<td>CER</td>
<td>Certificate Signing Request</td>
</tr>
<tr>
<td>CSR</td>
<td>Private key</td>
</tr>
<tr>
<td>KEY</td>
<td>Certificate Revocation List</td>
</tr>
<tr>
<td>DER</td>
<td>Certificate; DER-encoded</td>
</tr>
<tr>
<td>PEM</td>
<td>Certificate (including or not the private key), certificate chain, or Certificate Signing Request; PEM-encoded</td>
</tr>
</tbody>
</table>

### Other file type extensions

<table>
<thead>
<tr>
<th>P12</th>
<th>Certificate (including or not the private key), certificate chain, or Certificate Signing Request; bundled in a PKCS#12 archive file format</th>
</tr>
</thead>
</table>
openssl x509 -text -in cert.crt -noout
openssl req -text -in cert.csr -noout
openssl req -new -key cert.key -out cert.csr
openssl req -new -keyout cert.key -out cert.csr
openssl x509 -req -text -in cert.csr
openssl req -new -keyout cert.key -out cert.csr
openssl ca -config ca.conf -in cert.csr
openssl ca -config ca.conf -gencrl -revoke cert.crt
openssl ca -config ca.conf -gencrl -out list.crl
openssl ca -config ca.conf -gencrl -out cert.crt
openssl pkcs12 -export -in cert.pem -inkey cert.key -out cert.pfx
openssl pkcs12 -in cert.p12 -out cert.crt
openssl pkcs12 -in cert.p12 -out cert.key -nocerts -nodes
openssl pkey -text -in old.key -out new.key -noout
openssl pkey -text -in private.key -noout
openssl req -new -keyout cert.key -out cert.csr
openssl req -new -key cert.key -out cert.csr
openssl x509 -req -in cert.csr
openssl req -new -key cert.key -out cert.csr
openssl ca -config ca.conf -in cert.csr
openssl ca -config ca.conf -gencrl -revoke cert.crt
openssl ca -config ca.conf -gencrl -out list.crl
openssl pkcs12 -export -in cert.pem
openssl pkcs12 -in cert.p12
openssl pkcs12 -out cert.crt -nodes
openssl pkcs12 -out cert.key -cacerts
openssl pkcs12 -out cert.crt -cacerts.cat
openssl x509 -in cert.crt -outform DER -out cert.der
openssl x509 -in cert.crt
openssl x509 -text
openssl list-message-digest-commands
openssl list-cipher-commands
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CA.pl -newca</code></td>
<td>Create a Certification Authority hierarchy</td>
</tr>
<tr>
<td><code>CA.pl -newreq</code></td>
<td>Generate a Certificate Signing Request</td>
</tr>
<tr>
<td><code>CA.pl -newreq-nodes</code></td>
<td>Generate a Certificate Signing Request, creating also a key pair (unencrypted, for non-interactive use)</td>
</tr>
<tr>
<td><code>CA.pl -signreq</code></td>
<td>Sign a Certificate Signing Request</td>
</tr>
<tr>
<td><code>CA.pl -pkcs12 &quot;Certificate name&quot;</code></td>
<td>Generate a PKCS#12 certificate from a Certificate Signing Request</td>
</tr>
<tr>
<td><code>CA.pl -newcert</code></td>
<td>Generate a self-signed certificate</td>
</tr>
<tr>
<td><code>CA.pl -verify</code></td>
<td>Verify a certificate against the Certification Authority certificate for “demoCA”</td>
</tr>
</tbody>
</table>
gpg --gen-key  
Generate a key pair

gpg --import alice.asc  
Import Alice's public key alice.asc into your keyring

gpg --list-keys  
List the keys contained into your keyring

gpg --list-secret-keys  
List your private keys contained into your keyring

gpg --list-public-keys  
List the public keys contained into your keyring

gpg --export -o keyring.gpg  
Export your whole keyring to a file keyring.gpg

gpg --export-secret-key -a "You" -o private.key  
Export your private key to a file private.key

gpg --export-public-key -a "Alice" -o alice.pub  
Export Alice's public key to a file alice.pub

gpg --edit-key "Alice"  
Sign Alice's public key

gpg -e -u "You" -r "Alice" file  
Sign file (with your private key) and encrypt it to Alice (with Alice's public key)

gpg -d file.gpg -o file  
Decrypt file.gpg (with your own private key) and save the decrypted file to file

md5sum
sha1sum
sha224sum
sha256sum
sha384sum
sha512sum
shasum

Print or check the digest of a file generated by a specific hashing algorithm
OpenVPN is an open source software that implements a Virtual Private Network (VPN) between two endpoints. The encrypted VPN tunnel uses UDP port 1194.

```
openvpn --genkey --secret keyfile
```

Generate a shared secret keyfile for OpenVPN authentication. The keyfile must be copied on both server and client.

```
openvpn server.conf
openvpn client.conf
```

Start the VPN on the server side

Start the VPN on the client side

```
/etc/openvpn/server.conf
```

Server-side configuration file:

```
dev tun
ifconfig server_IP client_IP
keepalive 10 60
ping-timer-rem
persist-tun
persist-key
secret keyfile
```

```
/etc/openvpn/client.conf
```

Client-side configuration file:

```
remote server_public_IP
dev tun
ifconfig client_IP server_IP
keepalive 10 60
ping-timer-rem
persist-tun
persist-key
secret keyfile
```
## Key bindings - terminal

<table>
<thead>
<tr>
<th>Key</th>
<th>Alternate key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CTRL F</code></td>
<td></td>
<td>Move cursor forward one char</td>
</tr>
<tr>
<td><code>CTRL B</code></td>
<td></td>
<td>Move cursor backward one char</td>
</tr>
<tr>
<td><code>CTRL A</code></td>
<td><code>HOME</code></td>
<td>Move cursor to beginning of line</td>
</tr>
<tr>
<td><code>CTRL E</code></td>
<td><code>END</code></td>
<td>Move cursor to end of line</td>
</tr>
<tr>
<td><code>CTRL H</code></td>
<td><code>BACKSPACE</code></td>
<td>Delete char to the left of cursor</td>
</tr>
<tr>
<td><code>CTRL W</code></td>
<td></td>
<td>Delete word to the left of cursor</td>
</tr>
<tr>
<td><code>CTRL K</code></td>
<td></td>
<td>Delete all chars to the left of cursor</td>
</tr>
<tr>
<td><code>CTRL T</code></td>
<td></td>
<td>Delete all chars to the right of cursor</td>
</tr>
<tr>
<td><code>CTRL U</code></td>
<td><code>CTRL U</code></td>
<td>Swap current char with previous char</td>
</tr>
<tr>
<td><code>CTRL T</code></td>
<td></td>
<td>Swap current word with previous word</td>
</tr>
<tr>
<td><code>CTRL L</code></td>
<td></td>
<td>Scroll up the screen buffer</td>
</tr>
<tr>
<td><code>CTRL U</code></td>
<td></td>
<td>Scroll down the screen buffer</td>
</tr>
<tr>
<td><code>CTRL L</code></td>
<td></td>
<td>Clear screen (same as <code>clear</code>)</td>
</tr>
<tr>
<td><code>CTRL P</code></td>
<td><code>↑</code></td>
<td>Previous command in history</td>
</tr>
<tr>
<td><code>CTRL N</code></td>
<td><code>↓</code></td>
<td>Next command in history</td>
</tr>
<tr>
<td><code>CTRL R</code></td>
<td><code>ALT /</code></td>
<td>Reverse history search</td>
</tr>
<tr>
<td><code>TAB</code></td>
<td><code>ALT /</code></td>
<td>Autocomplete commands, filenames, and directory names</td>
</tr>
<tr>
<td><code>ALT </code></td>
<td></td>
<td>Autocomplete filenames and directory names only</td>
</tr>
<tr>
<td><code>CTRL J</code></td>
<td><code>RETURN</code></td>
<td>Expand the Bash alias currently entered on the command line</td>
</tr>
<tr>
<td><code>CTRL M</code></td>
<td></td>
<td>Line feed</td>
</tr>
<tr>
<td><code>CTRL S</code></td>
<td></td>
<td>Carriage return</td>
</tr>
<tr>
<td><code>CTRL Q</code></td>
<td></td>
<td>Pause transfer to terminal</td>
</tr>
<tr>
<td><code>CTRL Z</code></td>
<td></td>
<td>Forward history search (if XON/XOFF flow control is disabled)</td>
</tr>
<tr>
<td><code>CTRL D</code></td>
<td></td>
<td>Resume transfer to terminal</td>
</tr>
<tr>
<td><code>CTRL Q</code></td>
<td></td>
<td>Send a SIGTSTP to put the current job in background</td>
</tr>
<tr>
<td><code>CTRL Z</code></td>
<td></td>
<td>Send a SIGINT to stop the current process</td>
</tr>
<tr>
<td><code>CTRL D</code></td>
<td></td>
<td>Send a EOF to current process (if it's a shell, same as <code>logout</code>)</td>
</tr>
<tr>
<td><code>CTRL ALT DEL</code></td>
<td><code>CTRL ALT DEL</code></td>
<td>Send a SIGINT to reboot the machine (same as <code>shutdown -r now</code>); specified in <code>/etc/inittab</code> and <code>/etc/init/control-alt-delete</code></td>
</tr>
<tr>
<td><code>CTRL ALT F1 ... F6</code></td>
<td><code>CTRL ALT F1 ... F6</code></td>
<td>Switch between text consoles (same as <code>chvt n</code>)</td>
</tr>
</tbody>
</table>
### Key bindings - X

<table>
<thead>
<tr>
<th>Key</th>
<th>Alternate key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRL + ALT F7 ... F11</td>
<td></td>
<td>Switch between X Window consoles</td>
</tr>
<tr>
<td>CTRL + ALT + +</td>
<td></td>
<td>Increase X Window screen resolution</td>
</tr>
<tr>
<td>CTRL + ALT + -</td>
<td></td>
<td>Decrease X Window screen resolution</td>
</tr>
<tr>
<td>CTRL + TAB</td>
<td></td>
<td>Switch between X Window tasks</td>
</tr>
<tr>
<td>CTRL + ALT + ;</td>
<td>CTRL + ALT + L</td>
<td>Switch to next workspace</td>
</tr>
<tr>
<td>CTRL + ALT + ;</td>
<td>CTRL + ALT + R</td>
<td>Switch to previous workspace</td>
</tr>
<tr>
<td>CTRL + ALT + BACKSPACE</td>
<td></td>
<td>Reboot the X Window server</td>
</tr>
<tr>
<td>ALT + TAB</td>
<td></td>
<td>Switch between windows in the current workspace</td>
</tr>
<tr>
<td>SUPER + L</td>
<td></td>
<td>Show activities overview</td>
</tr>
<tr>
<td>SUPER + M</td>
<td></td>
<td>Lock screen</td>
</tr>
<tr>
<td>SUPER + ↓</td>
<td></td>
<td>Show tray messages</td>
</tr>
<tr>
<td>SUPER + ↓</td>
<td></td>
<td>Maximize current window</td>
</tr>
<tr>
<td>SUPER + ↑</td>
<td></td>
<td>Restore normal size of current window</td>
</tr>
<tr>
<td>SUPER + ←</td>
<td></td>
<td>Maximize current window to left half screen</td>
</tr>
<tr>
<td>SUPER + →</td>
<td></td>
<td>Maximize current window to right half screen</td>
</tr>
<tr>
<td>ALT + F2</td>
<td></td>
<td>Run command</td>
</tr>
<tr>
<td>CTRL + ↑</td>
<td></td>
<td>Increase terminal font size</td>
</tr>
<tr>
<td>CTRL + ↓</td>
<td></td>
<td>Decrease terminal font size</td>
</tr>
</tbody>
</table>

**GNOME**

- SUPER + ↓
- SUPER + ↑
The Hardware Abstraction Layer (HAL) manages device files and provides plug-and-play facilities. The HAL daemon `hald` maintains a persistent database of devices. udev is the device manager for the Linux kernel. It dynamically generates the device nodes in `/dev/` for devices present on the system; it also provides persistent naming for storage devices in `/dev/disk`. When a device is added, removed, or changes state, the kernel sends an uevent received by the `udevd` daemon which will pass the uevent through a set of rules stored in `/etc/udev/rules.d/*.rules` and `/lib/udev/rules.d/*.rules`.

<table>
<thead>
<tr>
<th><code>udevadm monitor</code></th>
<th>Show all kernel uevents and udev messages</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>udevadm info --attribute-walk --name=/dev/sda</code></td>
<td>Print all attributes of device <code>/dev/sda</code> in udev rules key format</td>
</tr>
<tr>
<td><code>cat /sys/block/sda/size</code></td>
<td>Print the size attribute of disk <code>sda</code> in 512-byte blocks. This information is retrieved from sysfs</td>
</tr>
<tr>
<td><code>udevadm test /dev/sdb</code></td>
<td>Simulate an udev event run for the device and print debug output</td>
</tr>
<tr>
<td><code>gnome-device-manager</code></td>
<td>Browser for the HAL device manager</td>
</tr>
</tbody>
</table>

```
/etc/udev/rules.d/*.rules and /lib/udev/rules.d/*.rules  udev rules

KERNEL=="hda", NAME="mydisk"  
Match a device which was named by the kernel as `hda`; name the device node as "mydisk". The device node will be therefore `/dev/mydisk`

KERNEL=="hdb", DRIVER=="ide-disk", SYMLINK=="mydisk myhd"  
Match a device with kernel name and driver as specified; name the device node with the default name and create two symbolic links `/dev/mydisk` and `/dev/myhd` pointing to `/dev/hdb`

KERNEL=="fd[0-9]*", NAME="floppy/%n", SYMLINK=="%k"  
Match all floppy disk drives (i.e. fdn); place device node in `/dev/floppy/n` and create a symlink `/dev/fd n` to it

SUBSYSTEM=="block", ATTR{size}=="41943040", SYMLINK=="mydisk"  
Match a block device with a size attribute of 41943040; create a symlink `/dev/mydisk`

KERNEL=="fd[0-9]*", OWNER="jdoe"  
Match all floppy disk drives; give ownership of the device file to user "jdoe"

KERNEL=="sda", PROGRAM="/bin/mydevicenamer %k", SYMLINK=="%c"  
Match a device named by the kernel as `sda`; to name the device, use the defined program which takes on stdin the kernel name and output on stdout e.g. `name1` `name2`. Create symlinks `/dev/name1` and `/dev/name2` pointing to `/dev/sda`

KERNEL=="sda", ACTION=="add", RUN=="/bin/myprogram"  
Match a device named by the kernel as `sda`; run the defined program when the device is connected

KERNEL=="sda", ACTION=="remove", RUN=="/bin/myprogram"  
Match a device named by the kernel as `sda`; run the defined program when the device is disconnected

%n = kernel number (e.g. = 3 for fd3)  
%k = kernel name (e.g. = fd3 for fd3)  
%c = device name as output from program
```
A kernel version number has the form `major.minor.patchlevel`.

Kernel images are usually gzip-compressed and can be of two types: zImage (max 520 Kb) and bzImage (no size limit). Kernel modules can be loaded dynamically into the kernel to provide additional functionalities on demand, instead of being included when the kernel is compiled; this reduces memory footprint. `kerneld` (daemon) and `kmod` (kernel thread) facilitate the dynamic loading of kernel modules.

```
/lib/modules/X.Y.Z/*.ko
/lib/modules/X.Y.Z/modules.dep
/etc/modules.conf
/etc/conf.modules (deprecated)
/usr/src/linux/
/usr/src/linux/.config
```

- Kernel modules for kernel version X.Y.Z
- Modules dependencies.
  - This file needs to be recreated (via the command `depmod -a`) after a reboot or a change in module dependencies
- Modules configuration file
- Directory containing the kernel source code to be compiled
- Kernel configuration file

```
freeramdisk
mknitrd initrd_image kernel_version (Red Hat)
mknitramfs (Debian)
dracut
dbus-monitor
dbus-monitor --session
dbus-monitor --system
```

- Free the memory used for the initrd image. This command must be run directly after unmounting /initrd
- Create an initrd image file
- Create an initrd image file according to the configuration file `/etc/initramfs-tools/initramfs.conf`
- Create initial ramdisk images for preloading modules
- Monitor messages going through a D-Bus message bus
  - Monitor session messages (default)
  - Monitor system messages

The runtime loader `ld.so` loads the required shared libraries of the program into RAM, searching in this order:

1. `LD_LIBRARY_PATH` Environment variable specifying the list of dirs where libraries should be searched for first
2. `/etc/ld.so.cache` Cache file
3. `/lib` and `/usr/lib` Default locations for shared libraries

Shared library locations (other than the default ones `/lib` and `/usr/lib`) can be specified in the file `/etc/ld.so.conf`.

```
ldconfig
```

- Create a cache file `/etc/ld.so.cache` of all available dynamically linked libraries.
  - To be run when the system complains about missing libraries

```
ldd program_or_lib
```

- Print library dependencies
Kernel management

\texttt{lspci} \hspace{1cm} \text{List PCI devices}
\texttt{lspci -d 8086:} \hspace{1cm} \text{List all Intel hardware present. PCI IDs are stored in:} \\
/\texttt{usr/share/misc/pci.ids} \hspace{1cm} \text{(Debian)} \\
/\texttt{usr/share/hwdata/pci.ids} \hspace{1cm} \text{(Red Hat)}

\texttt{lsusb} \hspace{1cm} \text{List USB devices}
\texttt{lsusb -d 8086:} \hspace{1cm} \text{List all Intel USB devices present. USB IDs are stored in:} \\
/\texttt{var/lib/usbutils/usb.ids} \hspace{1cm} \text{(Debian)} \\
/\texttt{usr/share/hwdata/usb.ids} \hspace{1cm} \text{(Red Hat)}

\texttt{lsdev} \hspace{1cm} \text{List information about the system hardware}

\texttt{lshw} \hspace{1cm} \text{List system hardware}

\texttt{lscpu} \hspace{1cm} \text{List information about the CPU architecture}

\texttt{uname -s} \hspace{1cm} \text{Print the kernel name}
\texttt{uname -n} \hspace{1cm} \text{Print the network node hostname}
\texttt{uname -r} \hspace{1cm} \text{Print the kernel release number } X.Y.Z \hspace{1cm} \text{USB IDs are stored in:} \\
/\texttt{var/lib/usbutils/usb.ids} \hspace{1cm} \text{(Debian)} \\
/\texttt{usr/share/hwdata/usb.ids} \hspace{1cm} \text{(Red Hat)}
\texttt{uname -v} \hspace{1cm} \text{Print the kernel version number}
\texttt{uname -m} \hspace{1cm} \text{Print the machine hardware name}
\texttt{uname -p} \hspace{1cm} \text{Print the processor type}
\texttt{uname -i} \hspace{1cm} \text{Print the hardware platform}
\texttt{uname -o} \hspace{1cm} \text{Print the operating system}
\texttt{uname -a} \hspace{1cm} \text{Print all the above information, in that order}

\texttt{evtest} \hspace{1cm} \text{Monitor and query input device events in } /\texttt{dev/input/eventn}

\texttt{dmesg} \hspace{1cm} \text{Print the messages of the kernel ring buffer}
\texttt{dmesg -n 1} \hspace{1cm} \text{Set the logging level to 1 (= only panic messages)}

\texttt{journalctl} \hspace{1cm} \text{Display the Systemd Journal, which contains the kernel logs}
\texttt{journalctl --since "1 hour ago"} \hspace{1cm} \text{Display the most recent } n \text{ log lines (default is } 10\text{)}
\texttt{journalctl -x} \hspace{1cm} \text{Display events happened in the last hour}
\texttt{journalctl -f} \hspace{1cm} \text{Display events, adding explanations from the message catalog}
\texttt{journalctl _SYSTEMD_UNIT=crond.service} \hspace{1cm} \text{Display the journal in real-time}
\texttt{journalctl -u crond.service} \hspace{1cm} \text{Display the log entries created by the cron service}
\texttt{mkdir -p /var/log/journal/ &\& \} \hspace{1cm} \text{Enable persistent storage of logs in } /\texttt{var/log/journal/}} \\
\texttt{systemctl restart systemd-journald} \hspace{1cm} \text{(by default, journalctl stores the logfiles in RAM only)}
## Kernel compile and patching

### Download

Download the kernel source code `linux-X.Y.Z.tar.bz2` from [http://www.kernel.org](http://www.kernel.org) to the base of the kernel source tree `/usr/src/linux`

### Clean

- `make clean` Delete most generated files
- `make mrproper` Delete all generated files and kernel configuration
- `make distclean` Delete temporary files, patch leftovers, and similar files

### Configure

- `make config` Terminal-based (options must be set in sequence)
- `make menuconfig` ncurses UI
- `make xconfig` GUI
- `make gconfig` GU
- `make oldconfig` Create a new configuration file, based on the options in the old configuration file and in the source code

Components (e.g. device drivers) can be either:
- not compiled
- compiled into the kernel binary, for support of devices always used on the system or necessary for the system to boot
- compiled as a kernel module, for optional devices

The configuration command creates a configuration file `/usr/src/linux/.config` containing instructions for the kernel compilation

### Build

- `make bzImage` Compile the kernel
- `make modules` Compile the kernel modules
- `make all` Compile kernel and kernel modules
- `make -j2 all` will speed up compilation by allocating 2 simultaneous compile jobs

### Modules install

- `make modules_install` Install the previously built modules present in `/lib/modules/X.Y.Z`

### Kernel install

1. Copy the new compiled kernel and other files into the boot partition:
   - `cp /usr/src/linux/arch/boot/bzImage /boot/vmlinuz-X.Y.Z` (kernel)
   - `cp /usr/src/linux/arch/boot/System.map-X.Y.Z /boot` (config options used for this compile)
   - `cp /usr/src/linux/arch/boot/config-X.Y.Z /boot` (config options used for this compile)

2. Create an entry in GRUB to boot on the new kernel

### Package

Optionally, the kernel can be packaged for install on other machines

- `make rpm-pkg` Build source and binary RPM packages
- `make binrpm-pkg` Build binary RPM package
- `make deb-pkg` Builds binary DEB package

### Kernel patching

Download and decompress the patch to `/usr/src`

#### Patch

- `patch -p1 < file.patch` Apply the patch
- `patch -Rp1 < file.patch` Remove (reverse) a patch. Alternatively, applying the patch again reverses it

#### Build

Build the patched kernel as explained above

#### Install

Install the patched kernel as explained above
Kernel modules allow the kernel to access functions (symbols) for kernel services e.g. hardware drivers, network stack, or filesystem abstraction.

**lsmod**
List the modules that are currently loaded into the kernel

**insmod module**
Insert a module into the kernel. If the module requires another module or if it does not detect compatible hardware, insertion will fail

**rmmod module**
Remove a module from the kernel. If the module is in use by another module, it is necessary to remove the latter first

**modinfo module**
Display the list of parameters accepted by the module

**depmod -a**
Probe all modules in the kernel modules directory and generate the file that lists their dependencies

It is recommended to use **modprobe** instead of **insmod** and **rmmod**, because it automatically handles prerequisites when inserting modules, is more specific about errors, and accepts just the module name instead of requiring the full path name.

**modprobe module option=value**
Insert a module into the running kernel, with the specified parameters. Prerequisite modules will be inserted automatically

**modprobe -a**
Insert all modules

**modprobe -t directory**
Attempt to load all modules contained in the directory until a module succeeds. This action probes the hardware by successive module-insertion attempts for a single type of hardware, e.g. a network adapter

**modprobe -r module**
Remove a module

**modprobe -c module**
Display module configuration

**modprobe -l**
List loaded modules

---

**Configuration of device drivers**

Device drivers support the kernel with instructions on how to use that device.

<table>
<thead>
<tr>
<th>Device driver compiled into the kernel</th>
<th>Configure the device driver by passing a kernel parameter in the GRUB menu:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>kernel /vmlinuz ro root=/dev/vg0/root vga=0x33c</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device driver provided as a kernel module</th>
<th>Edit module configuration in /etc/modprobe.conf or /etc/modprobe.d/ (Red Hat):</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>alias eth0 3c59x</td>
</tr>
<tr>
<td></td>
<td>options 3c509 irq=10,11</td>
</tr>
<tr>
<td></td>
<td>Specify that eth0 uses the 3c59x.ko driver module</td>
</tr>
<tr>
<td></td>
<td>Assign IRQ 10 and 11 to 3c509 devices</td>
</tr>
</tbody>
</table>
/proc is a pseudo filesystem that gives access to process data held in the kernel.

<table>
<thead>
<tr>
<th>File</th>
<th>Information stored (can be viewed via cat)</th>
<th>Equivalent command</th>
</tr>
</thead>
<tbody>
<tr>
<td>/proc/buses</td>
<td>Buses (e.g. PCI, USB, PC Card)</td>
<td></td>
</tr>
<tr>
<td>/proc/cpuinfo</td>
<td>CPUs information</td>
<td></td>
</tr>
<tr>
<td>/proc/devices</td>
<td>Drivers currently loaded</td>
<td></td>
</tr>
<tr>
<td>/proc/dma</td>
<td>DMA channels in use</td>
<td></td>
</tr>
<tr>
<td>/proc/filesystems</td>
<td>Filesystems supported by the system</td>
<td></td>
</tr>
<tr>
<td>/proc/interrupts</td>
<td>Current IRQs (Interrupt Requests)</td>
<td>/procinfo</td>
</tr>
<tr>
<td>/proc/ioports</td>
<td>I/O addresses in use</td>
<td></td>
</tr>
<tr>
<td>/proc/loadavg</td>
<td>System load averages</td>
<td>uptime</td>
</tr>
<tr>
<td>/proc/mdstat</td>
<td>Information about RAID arrays and devices</td>
<td></td>
</tr>
<tr>
<td>/proc/meminfo</td>
<td>Total and free memory</td>
<td>free</td>
</tr>
<tr>
<td>/proc/modules</td>
<td>Kernel modules currently loaded</td>
<td>lsmod</td>
</tr>
<tr>
<td>/proc/mounts</td>
<td>Mounted partitions</td>
<td>mount</td>
</tr>
<tr>
<td>/proc/net/dev</td>
<td>Network interface statistics</td>
<td></td>
</tr>
<tr>
<td>/proc/partitions</td>
<td>Drive partition information</td>
<td>fdisk -1</td>
</tr>
<tr>
<td>/proc/swaps</td>
<td>Size of total and used swap areas</td>
<td>swapon -s</td>
</tr>
<tr>
<td>/proc/sys/</td>
<td>sysfs: exposes tunable kernel parameters</td>
<td></td>
</tr>
<tr>
<td>/proc/sys/kernel/</td>
<td>Kernel information and parameters</td>
<td></td>
</tr>
<tr>
<td>/proc/sys/net/</td>
<td>Network information and parameters</td>
<td></td>
</tr>
<tr>
<td>/proc/uptime</td>
<td>Time elapsed since boot</td>
<td>uptime</td>
</tr>
<tr>
<td>/proc/version</td>
<td>Linux version</td>
<td>uname -a</td>
</tr>
<tr>
<td>/proc/n/</td>
<td>Information about process with PID n</td>
<td>ps n</td>
</tr>
<tr>
<td>/proc/n/cmdline</td>
<td>Command by which the process was launched</td>
<td></td>
</tr>
<tr>
<td>/proc/n/cwd</td>
<td>Symlink to process' working directory</td>
<td></td>
</tr>
<tr>
<td>/proc/n/environ</td>
<td>Values of environment variables of process</td>
<td></td>
</tr>
<tr>
<td>/proc/n/exe</td>
<td>Symlink to process' executable</td>
<td></td>
</tr>
<tr>
<td>/proc/n/fd</td>
<td>Files currently opened by the process</td>
<td>lsof -p n</td>
</tr>
<tr>
<td>/proc/n/root</td>
<td>Symlink to process' filesystem root</td>
<td></td>
</tr>
<tr>
<td>/proc/n/status</td>
<td>Status of process</td>
<td></td>
</tr>
</tbody>
</table>

/sysctl is the only writable branch of /proc and can be used to tune kernel parameters on-the-fly. All changes are lost after system shutdown, unless applied via sysctl -p.

```
sysctl fs.file-max
Get the maximum allowed number of open files

cat /proc/sys/fs/file-max

sysctl -w "fs.file-max=100000"
Set the maximum allowed number of open files to 100000

echo "100000" > /proc/sys/fs/file-max

sysctl -a
List all available kernel tuning options

sysctl -p
Apply all tuning settings listed in /etc/sysctl.conf. This command is usually run at boot by the system initialization script, to make permanent changes to kernel parameters
```
If the kernel has been booted in emergency mode and `init` has not been run, some initial configuration is necessary e.g.

```
mount /proc
mount -o remount,rw /
mount -a
```

If mounting the filesystems fails:

```
mknod /dev/sda
mknod /dev/sdal
fdisk -l /dev/sda
fsck -y /dev/sdal
mount -t ext3 /dev/sdal /mnt/sysimage
chroot /mnt/sysimage
```

To install a package using an alternative root directory (useful if the system has been booted from a removable media):

```
rpm -U --root /mnt/sysimage package.rpm
```

To install GRUB on the specified directory (which must contain `/boot/grub/`):

```
grub-install --root-directory=/mnt/sysimage /dev/sda
```

Alternative method:

```
chroot /mnt/sysimage
grub-install /dev/sda
```

Run `sync` and unmount all filesystems before exiting the shell, to ensure that all changes have been written on disk.

**How to reset the root password (RHEL 7)**

1. Power up the system and, on the GRUB 2 boot screen, press `E` to edit the current entry.
2. Edit the kernel line that mentions `linux16`, removing the `rhgb` and `quiet` parameters and adding `rd.break` at the end.
3. Press `CTRL+X`; the system will boot on the initramfs `switch_root` prompt.
4. Remount the filesystem as writable:
   ```
   mount -o remount,rw /sysroot
   chroot /sysroot
   ```
5. Change the filesystem root:
   ```
   passwd root
   ```
6. Modify the root password:
7. Force SELinux to relabel context on next boot:
8. Remount the filesystem as readonly (not strictly necessary):
9. Exit the chroot environment:
10. Resume system boot:
### DNS implementations

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIND</td>
<td>Berkeley Internet Name Domain system, is the standard DNS server for UNIX</td>
</tr>
<tr>
<td>Unbound</td>
<td>Standard DNS server in RHEL 7</td>
</tr>
<tr>
<td>dnsmasq</td>
<td>Lightweight DNS, DHCP and TFTP server for a small network</td>
</tr>
<tr>
<td>djbdns</td>
<td>Security-hardened DNS server that also includes DNS debugging tools</td>
</tr>
<tr>
<td>PowerDNS</td>
<td>Alternative open-source DNS server</td>
</tr>
</tbody>
</table>

**named**  
BIND Name Daemon

**rndc**  
Name Daemon Controller for BIND 8

**rndc**  
Remote Name Daemon Controller for BIND 9, uses a shared key to communicate securely with named

**dnswalk example.org.**  
DNS debugger

**rndc reconfig**  
Reload BIND configuration and new zones

**rndc reload example.org**  
Reload the zone example.org

**rndc freeze example.org**  
Suspend updates for the zone example.org

**rndc thaw example.org**  
Resume updates for the zone example.org

**rndc tsig-list**  
List all currently active TSIG keys

**DNSSEC** was designed to secure the DNS tree and hence prevent cache poisoning. The TSIG (Transaction SIGnature) standard, that authenticates communications between two trusted systems, is used to sign zone transfers and DDNS (Dynamic DNS) updates.

**dnssec-keygen -a dsa -b 1024 -n HOST dns1.example.org**  
Generate a TSIG key with DNSSEC algorithm *nnn* and key fingerprint *ffff*.  
This will create two key files  
*Kdns1.example.org.+nnn+fffff.key*  
*Kdns1.example.org.+nnn+fffff.private*  
which contain a key number that must be inserted both in `/etc/named.conf` and `/etc/rndc.conf`

**rndc-confgen -a**  
Generate a `/etc/rndc.key` key file:

```plaintext
key "rndc-key" {
    algorithm hmac-md5;
    secret "vy2gL3tPHsqnA57e4LTeE=";
};
options {
    default-key "rndc-key";
    default-server 127.0.0.1;
    default-port 953;
};
```

This file is automatically read both by *named* and *rndc*

**dnssec-signzone example.org**  
Sign the zone example.org

**named -u named -g named**  
Run BIND as user/group "named" (must be created if needed) instead of root

**named -t /var/cache/bind**  
Run BIND in a chroot jail `/var/cache/bind`  
(actually it is the *chroot* command that starts the named server)
## DNS server configuration file: /etc/named.conf

```conf

controls {
    inet 127.0.0.1 allow {localhost;}; keys {rndckey;};
};

key "rndc-key" {                                 // TSIG key
    algorithm dsa;
    secret "HYZur46fftdUQ43BJKI093t4t781kp";
};

acl "mynetwork" {10.7.0.0/24;};                // Alias definition
    // Built-in ACLs: any, none, localhost, localnets

options {
    directory "/var/named";                     // Working directory
    version "0.0";                              // Hide version number by replacing it with 0.0
    listen-on port 53 {10.7.0.1; 127.0.0.1;};   // Port and own IP addresses to listen on
    blackhole {172.17.17.0/24;};               // IPs whose packets are to be ignored
    allow-query {mynetwork;};                   // IPs allowed to do iterative queries
    allow-query-on {any;};                      // Local IPs that can accept iterative queries
    allow-query-cache {any;};                   // IPs that can get an answer from cache
    allow-recursion {mynetwork;};       // IPs to accept recursive queries from (typically
        // resolution process on behalf of these client IPs,
        // and returns a referral for the other IPs
    allow-recursion-on {mynetwork;};          // Local IPs that can accept recursive queries
    allow-transfer {10.7.0.254;};             // Zone transfer is restricted to these IPs (slaves);
        // on slave servers, this option should be disabled
    allow-update {any;};                      // IPs to accept DDNS updates from
    recursive-clients 1000;                   // Max number of simultaneous recursive lookups
    dnssec-enable yes;                        // Enable DNSSEC
    dialup no;                                // Not a dialup connection: external zone maintenance
        // (e.g. sending heartbeat packets, external zone transfers)
        // is then permitted
    forward first;                             // Site-wide cache: bypass the normal resolution
    forwarders {10.7.0.252; 10.7.0.253;};      // method by querying first these central DNS
        // servers if they are available
};

// Define the root name servers
zone "." {                     // Working directory
    type hint;
    file "root.cache";
}                                // Hide version number by replacing it with 0.0

// Configure system to act as a master server for the example.org domain
zone "example.org" IN {       // Zone file for the example.org domain
    type master;
    file "master/example.org.zone";
};
zone "240.123.224.in-addr.arpa" IN {   // Configure reverse lookup zone (for 224.123.240.0/24)
    type master;
    file "slave/example.org.revzone";
};

// Configure system to act as a slave server for the example2.org domain
zone "example2.org" IN {       // Slave: do not edit this zone file!
    type slave;
    file "slave/example2.org.zone";
    masters {10.7.0.254;};
};
zone "0.7.10.in-addr.arpa" IN {  // Configure reverse lookup zone (for 10.7.0.0/24)
    type slave;
    file "slave/10.7.0.revzone";
    masters {10.7.0.254;};
};
```
DNS zone file

```
/var/named/master/example.org.zone  DNS zone file for the example.org zone

$TTL 86400  ; TTL (1 day)
$ORIGIN example.org.
example.org IN SOA dns1.example.org. help.example.org. (   ; Master DNS server is dns1.example.org
 2014052300   ; serial                                   ; If problems, contact help@example.org
 28800        ; refresh (8 hours)
 7200         ; retry (2 hours)
 604800       ; expire (1 week)
 600 )        ; negative TTL (10 mins)
  IN NS     dns1.example.org.
  IN NS     dns2.example.org.
  IN MX     10 mail1.example.org.
  IN MX     20 mail2.example.org.

dns1 IN A      224.123.240.3
dns2 IN A      224.123.240.4
mail1 IN A     224.123.240.73
mail2 IN A     224.123.240.77
foo IN A       224.123.240.10
bar IN A       224.123.240.13
www IN A       224.123.240.19
baz IN CNAME   bar
subdomain IN NS ns1.subdomain.example.org.  ; Glue records
  IN NS   ns2.subdomain.example.org.
ns1.subdomain.example.org.   IN A   224.123.240.201
ns2.subdomain.example.org.   IN A   224.123.240.202
```

/var/named/master/example.org.revzone  DNS reverse zone file for the example.org zone

```
$TTL 86400  ; TTL (1 day)
exmaple.org IN SOA dns1.example.org. help.example.org. (   ; Master DNS server is dns1.example.org
 2014052300   ; serial                                   ; If problems, contact help@example.org
 28800        ; refresh (8 hours)
 7200         ; retry (2 hours)
 604800       ; expire (1 week)
 600 )        ; negative TTL (10 mins)
 12.240.123.224.in-addr.arpa   IN PTR   foo
 13.240.123.224.in-addr.arpa   IN PTR   bar
 19.240.123.224.in-addr.arpa   IN PTR   www
```

### Resource Records

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TTL</td>
<td>How long to cache a positive response</td>
</tr>
<tr>
<td>$ORIGIN</td>
<td>Suffix appended to all names not ending with a dot. Useful when defining multiple subdomains inside the same zone</td>
</tr>
<tr>
<td>SOA</td>
<td>Start Of Authority for the example.org zone</td>
</tr>
<tr>
<td>serial</td>
<td>Serial number. Must be increased after each edit of the zone file</td>
</tr>
<tr>
<td>refresh</td>
<td>How frequently a slave server refreshes its copy of zone data from the master</td>
</tr>
<tr>
<td>retry</td>
<td>How frequently a slave server retries connecting to the master</td>
</tr>
<tr>
<td>expire</td>
<td>How long a slave server relies on its copy of zone data. After this time period expires, the slave server is not authoritative anymore for the zone unless it can contact a master</td>
</tr>
<tr>
<td>negative TTL</td>
<td>How long to cache a non-existent answer</td>
</tr>
<tr>
<td>A</td>
<td>Address: maps names to IP addresses. Used for DNS lookups.</td>
</tr>
<tr>
<td>PTR</td>
<td>Pointer: maps IP addresses to names. Used for reverse DNS lookups. Each A record must have a matching PTR record</td>
</tr>
<tr>
<td>CNAME</td>
<td>Canonical Name: specifies an alias for a host with an A record (even in a different zone). Discouraged as it causes multiple lookups; it is better to use multiple A records instead</td>
</tr>
<tr>
<td>NS</td>
<td>Name Service: specifies the authoritative name servers for the zone</td>
</tr>
<tr>
<td>MX</td>
<td>Mailserver: specifies address and priority of the servers able to handle mail for the zone</td>
</tr>
<tr>
<td>Glue Records</td>
<td>are not really part of the zone; they delegate authority for other zones, usually subdomains</td>
</tr>
</tbody>
</table>
### HTTP response codes

<table>
<thead>
<tr>
<th>1XX Informational</th>
<th>2XX Success</th>
<th>3XX Redirection</th>
<th>4XX Client Error</th>
<th>5XX Server Error</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>100 Continue</strong></td>
<td><strong>200 OK</strong></td>
<td><strong>301 Moved Permanently</strong></td>
<td><strong>400 Bad Request</strong></td>
<td><strong>500 Internal Server Error</strong></td>
</tr>
<tr>
<td>The server received the request headers, so the client should continue by sending the remainder of the request</td>
<td>The request was successful</td>
<td>The requested resource was permanently moved to a new URI</td>
<td>The server was unable to understand the request due to bad syntax</td>
<td>The server encountered a generic error while trying to fulfill the request</td>
</tr>
<tr>
<td><strong>101 Switching Protocols</strong></td>
<td><strong>201 Created</strong></td>
<td><strong>302 Found</strong></td>
<td><strong>401 Unauthorized</strong></td>
<td><strong>501 Not Implemented</strong></td>
</tr>
<tr>
<td>The server agreed to switch protocol upon client's demand</td>
<td>The request was successful, and resulted in a resource being created</td>
<td>The requested resource was temporarily moved to a new URI</td>
<td>The request requires user authentication</td>
<td>The server was unable to recognize the request method</td>
</tr>
<tr>
<td><strong>204 No Content</strong></td>
<td><strong>203 See Other</strong></td>
<td><strong>303 Not Modified</strong></td>
<td><strong>403 Forbidden</strong></td>
<td><strong>502 Bad Gateway</strong></td>
</tr>
<tr>
<td>The request was successful, and the server does not need to return any content</td>
<td>The requested resource can be found on another URI, and should be retrieved from there via a GET</td>
<td>The client sent a conditional GET request, and the resource has not been modified since last time it was requested</td>
<td>The client did not have the necessary permissions to access the requested resource</td>
<td>The server is acting as a gateway or proxy, and received an invalid response from the upstream server</td>
</tr>
<tr>
<td><strong>206 Partial Content</strong></td>
<td><strong>204 Not Found</strong></td>
<td><strong>304 Temporary Redirect</strong></td>
<td><strong>404 Not Found</strong></td>
<td><strong>503 Service Unavailable</strong></td>
</tr>
<tr>
<td>The request was successful, and the server is returning only partial content because the client sent a Range header field</td>
<td>The requested resource was not found on the server</td>
<td>The requested resource was temporarily moved to a new URI, but future requests should use the original URI</td>
<td>The requested resource was not found on the server</td>
<td>The server is temporarily unavailable due to overload or maintenance</td>
</tr>
<tr>
<td><strong>304 Not Modified</strong></td>
<td><strong>408 Request Timeout</strong></td>
<td><strong>410 Gone</strong></td>
<td><strong>504 Gateway Timeout</strong></td>
<td><strong>505 HTTP Version Not Supported</strong></td>
</tr>
<tr>
<td>The server timed out while waiting for the request</td>
<td>The request could not be processed because of a conflict in the resource state</td>
<td>The requested resource is no longer available on the server and will not be available again</td>
<td>The server is acting as a gateway or proxy, and a request to the upstream server timed out</td>
<td>The server does not support the HTTP protocol version used in the request</td>
</tr>
</tbody>
</table>
Apache is an open source and widespread HTTP server, originally based on the NCSA HTTPd server.

**apachectl** (Red Hat) **httpd** (Red Hat) **apache2ctl** (Debian)

Manage the Apache webserver

Start the Apache webserver daemon

Display a brief status report

Display a detailed status report

Gracefully restart Apache; currently open connections are not aborted

Gracefully stop Apache; currently open connections are not aborted

Test the configuration file, reporting any syntax error

List all loaded and shared modules

/var/www/html Default document root directory

$HOME/public_html Default document root directory for users’ websites

Web content must be readable by the user/group the Apache process runs as. For security reasons, it should be owned and writable by the superuser or the webmaster user/group (usually *www-data*), not the Apache user/group.

/etc/httpd/conf/httpd.conf
/etc/httpd/conf.d/*.conf (Red Hat) Apache configuration files

/etc/apache2/httpd.conf (Debian and SUSE)

The Apache webserver contains a number of MPMs (Multi-Processing Modules) which can operate following two methods:

**prefork MPM**
A number of child processes is spawned in advance, with each child serving one connection. Highly reliable due to Linux memory protection that isolates each child process.

**worker MPM**
Multiple child processes spawn multiple threads, with each thread serving one connection. More scalable but prone to deadlocks if third-party non-threadsafe modules are loaded.

**HTTPS**

HTTPS (i.e. HTTP over SSL/TLS) allows securing communications between the webserver and the client by encrypting all communications end-to-end between the two. A webserver using HTTPS hands over its public key to the client when the client connects to the server via port 443. The server’s public key is signed by a CA (Certification Authority), whose validity is ensured by the root certificates stored into the client’s browser.

The `openssl` command and its user-friendly `CA.pl` script are the tools of the OpenSSL crypto library that can be used to accomplish all public key crypto operations e.g. generate key pairs, Certificate Signing Requests, and self-signed certificates. Another user-friendly tool is `genkey`.

Virtual hosting with HTTPS requires assigning a unique IP address for each virtual host; this because the SSL handshake (during which the server sends its certificate to the client’s browser) takes place before the client sends the `Host:` header (which tells to which virtual host the client wants to talk).

A workaround for this is SNI (Server Name Indication) which makes the browser send the hostname in the first message of the SSL handshake. Another workaround is to have all multiple name-based virtual hosts use the same SSL certificate with a wildcard domain e.g. *.*example.org.
### Apache server configuration

**Apache configuration file**

<table>
<thead>
<tr>
<th>Server configuration directives</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ServerName <a href="http://www.mysite.org:80">www.mysite.org:80</a></strong></td>
<td>Name and port (if omitted, uses default HTTP port 80) of server</td>
</tr>
<tr>
<td><strong>ServerRoot /etc/httpd</strong></td>
<td>Root directory for configuration and log files</td>
</tr>
<tr>
<td><strong>ServerAdmin <a href="mailto:webmaster@mysite.org">webmaster@mysite.org</a></strong></td>
<td>Contact address that the server includes in any HTTP error messages to the client. Can be an email address or a URL</td>
</tr>
<tr>
<td><strong>StartServers 5</strong></td>
<td>Number of servers to start initially</td>
</tr>
<tr>
<td><strong>MinSpareServers 5</strong></td>
<td>Minimum and maximum number of idle child server processes</td>
</tr>
<tr>
<td><strong>MaxSpareServers 10</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ServerLimit 256</strong></td>
<td>Prefork MPM: max configured value for <code>MaxRequestWorkers</code>. Worker MPM: max total number of threads available to serve requests</td>
</tr>
<tr>
<td><strong>ThreadsPerChild 25</strong></td>
<td>Worker MPM: number of threads created by each child process</td>
</tr>
<tr>
<td><strong>ThreadLimit 64</strong></td>
<td>Worker MPM: max configured value for <code>ThreadsPerChild</code></td>
</tr>
<tr>
<td><strong>MaxRequestsPerChild 16</strong></td>
<td>Worker MPM: max configured value for <code>MaxRequestsPerChild</code></td>
</tr>
<tr>
<td><strong>MaxConnectionsPerChild 16</strong></td>
<td>Worker MPM: max configured value for <code>MaxConnectionsPerChild</code></td>
</tr>
<tr>
<td><strong>LoadModule mime_module modules/mod_mime.so</strong></td>
<td>Load the module <code>mime_module</code> by linking in the object file or library <code>modules/mod_mime.so</code></td>
</tr>
<tr>
<td><strong>Listen 10.17.1.1:80</strong></td>
<td>Make the server accept connections on the specified IP addresses (optional) and ports</td>
</tr>
<tr>
<td><strong>Listen 10.17.1.5:8080</strong></td>
<td></td>
</tr>
<tr>
<td><strong>User nobody</strong></td>
<td>User and group the Apache process runs as. For security reasons, this should not be root</td>
</tr>
<tr>
<td><strong>Group nobody</strong></td>
<td></td>
</tr>
</tbody>
</table>

| MaxClient 256 | Max number of simultaneous requests that will be served; clients above this limit will get a HTTP error 503 - Service Unavailable. |
| MaxRequestWorkers 256 | Prefork MPM: max number of child processes launched to serve requests. Worker MPM: max total number of threads available to serve requests |
| (before v2.3.13) | |
| (v2.3.13 and later) | |
| MaxClients 256 | Max number of simultaneous requests that will be served; clients above this limit will get a HTTP error 503 - Service Unavailable. |
| MaxRequestWorkers 256 | Prefork MPM: max configured value for `MaxRequestWorkers`. Worker MPM: max total number of threads available to serve requests |
| (before v2.3.13) | |
| (v2.3.13 and later) | |
## Apache main configuration

### Apache configuration file

<table>
<thead>
<tr>
<th><strong>Main configuration directives</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DocumentRoot</strong> /var/www/html</td>
</tr>
<tr>
<td><strong>Alias</strong> /image /mydir/pub/image</td>
</tr>
<tr>
<td><strong>TypesConfig</strong> conf/mime.types</td>
</tr>
<tr>
<td><strong>AddType</strong> image/jpeg jpeg jpg jpe</td>
</tr>
<tr>
<td><strong>Redirect</strong> permanent /foo /bar</td>
</tr>
<tr>
<td><strong>Redirect</strong> /foo <a href="http://www.example.com/foo">http://www.example.com/foo</a></td>
</tr>
<tr>
<td><strong>AccessFileName</strong> .htaccess</td>
</tr>
</tbody>
</table>


Specify which global directives an .htaccess file can override:

- **AuthConfig** Authorization directives for directory protection
- **FileInfo** Document type and metadata
- **Indexes** Directory indexing
- **Limit** Host access control
- **Options** Specific directory features
- **All** All directives
- **None** No directive


Limit the scope of the specified directives to the directory /var/www/html/foobar and its subdirectories

```<Location /foobar> [list of directives] </Location>```

Limit the scope of the specified directive to the URL http://www.mysite.org/foobar/ and its subdirectories

### Logging directives

| **LogFormat** "%h %l %u %t "%r" %>s %b" | Specify the format of a log |
| **LogFormat** "%h %l %u %t \"%r\" %s %b" common | Specify a nickname for a log format. In this case, specifies "common" for the CLF (Common Log Format) which is defined as such: %h IP address of the client host %l Identity of client as determined by identd %u User ID of client making the request %t Timestamp the server completed the request %r Request as done by the user %s Status code sent by the server to the client %b Size of the object returned, in bytes |
| **CustomLog** /var/log/httpd/access_log common | Set up a log filename, with the format or (as in this case) the nickname specified |
| **TransferLog** /var/log/httpd/access_log | Set up a log filename, with format determined by the most recent LogFormat directive which did not define a nickname |
| **TransferLog** "|rotatelogs access_log 86400" | Set log rotation every 24 hours |
| **HostnameLookups** Off | Disable DNS hostname lookup to save network traffic. Hostnames can be resolved later by processing the log file: logresolve <access_log >accessdns_log |
### Apache virtual hosts

#### NameVirtualHost *

Specify which IP address will serve virtual hosting. The argument can be an IP address, an address:port pair, or * for all IP addresses of the server. The same argument need to be inserted in the relevant `<VirtualHost>` directive.

```
<VirtualHost *:80>
  ServerName www.mysite.org
  ServerAlias mysite.org *.mysite.org
  DocumentRoot /var/www/vhosts/mysite
</VirtualHost>
```

The first listed virtual host is also the default virtual host. It inherits those main settings that does not override. This virtual host answers to http://www.mysite.org, and also redirects there all HTTP requests on the domain mysite.org.

```
<VirtualHost *:80>
  ServerAdmin webmaster@www.mysite2.org
  ServerName www.mysite2.org
  DocumentRoot /var/www/vhosts/mysite2
  ErrorLog /var/www/logs/mysite2
</VirtualHost>
```

Name-based virtual host http://www.mysite2.org. Multiple name-based virtual hosts can share the same IP address; DNS must be configured accordingly to map each name to the correct IP address. Cannot be used with HTTPS.

```
<VirtualHost *:8080>
  ServerName www.mysite3.org
  DocumentRoot /var/www/vhosts/mysite3
</VirtualHost>
```

Port-based virtual host answering to connections on port 8080. A `Listen 8080` directive must also be present.

```
<VirtualHost 10.17.1.5:80>
  ServerName www.mysite4.org
  DocumentRoot /var/www/vhosts/mysite4
</VirtualHost>
```

IP-based virtual host answering to http://10.17.1.5.
### Apache configuration file

<table>
<thead>
<tr>
<th>Authorization directives</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthName &quot;Protected zone&quot;</td>
<td>Name of the realm. The client will be shown the realm name and prompted to enter a user and password</td>
</tr>
<tr>
<td>AuthType Basic</td>
<td>Type of user authentication: Basic, Digest, Form, or None</td>
</tr>
<tr>
<td>AuthUserFile &quot;/var/www/.htpasswd&quot;</td>
<td>User database file. Each line has the format user:encryptedpassword To add a user to the database file, use the command: htpasswd /var/www/.htpasswd user (will prompt for password)</td>
</tr>
<tr>
<td>AuthGroupFile &quot;/var/www/.htgroup&quot;</td>
<td>Group database file. Each line specifies a group followed by the usernames of all its members: group: user1 user2 user3</td>
</tr>
<tr>
<td>Require valid-user</td>
<td>Control who can access the protected resource. valid-user any user in the user database file user user only the specified user group group only the members of the specified group</td>
</tr>
<tr>
<td>Satisfy Any</td>
<td>Set the access policy concerning user and host control. All both Require and Allow criteria must be satisfied Any any of Require or Allow criteria must be satisfied</td>
</tr>
<tr>
<td>Allow from 10.13.13.0/24 Deny from 10.13.14.0/24</td>
<td>Control which host can or cannot access the protected resource</td>
</tr>
<tr>
<td>Order Allow,Deny (v2.2)</td>
<td>Control the evaluation order of Allow and Deny directives. Allow,Deny First, all Allow directives are evaluated; at least one must match, or the request is rejected. Next, all Deny directives are evaluated; if any matches, the request is rejected. Last, any requests which do not match an Allow or a Deny directive are denied Deny,Allow First, all Deny directives are evaluated; if any match, the request is denied unless it also matches an Allow directive. Any requests which do not match any Allow or Deny directives are permitted</td>
</tr>
<tr>
<td><strong>Apache configuration file</strong></td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>SSL/TLS directives (mod_ssl module)</strong></td>
<td></td>
</tr>
<tr>
<td>SSLCertificateFile \ /etc/httpd/conf/ssl.crt/server.crt</td>
<td>SSL server certificate</td>
</tr>
<tr>
<td>SSLCertificateKeyFile \ /etc/httpd/conf/ssl.key/server.key</td>
<td>SSL server private key (for security reasons, this file must be mode 600 and owned by root)</td>
</tr>
<tr>
<td>SSLCACertificatePath \ /usr/local/apache2/conf/ssl.crt/</td>
<td>Directory containing the certificates of CAs. Files in this directory are PEM-encoded and accessed via symlinks to hash filenames</td>
</tr>
<tr>
<td>SSLCACertificateFile \ /usr/local/apache2/conf/ssl.crt/ca-bundle.crt</td>
<td>Certificates of CAs. Certificates are PEM-encoded and concatenated in a single bundle file in order of preference</td>
</tr>
<tr>
<td>SSLCertificateChainFile \ /usr/local/apache2/conf/ssl.crt/ca.crt</td>
<td>Certificate chain of the CAs. Certificates are PEM-encoded and concatenated from the issuing CA certificate of the server certificate to the root CA certificate. Optional</td>
</tr>
<tr>
<td>SSLEngine on</td>
<td>Enable the SSL/TLS Protocol Engine</td>
</tr>
<tr>
<td>SSLProtocol +SSLv3 +TLSv1.2</td>
<td>SSL protocol flavors that the client can use to connect to server. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>SSLv2 (deprecated)</td>
</tr>
<tr>
<td></td>
<td>SSLv3</td>
</tr>
<tr>
<td></td>
<td>TLSv1</td>
</tr>
<tr>
<td></td>
<td>TLSv1.1</td>
</tr>
<tr>
<td></td>
<td>TLSv1.2</td>
</tr>
<tr>
<td></td>
<td>All (all the above protocols)</td>
</tr>
<tr>
<td>SSLCipherSuite \ ALL:!aDH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP</td>
<td>Cipher suite available for the SSL handshake (key exchange algorithms, authentication algorithms, cipher/encryption algorithms, MAC digest algorithms)</td>
</tr>
<tr>
<td>ServerTokens Full</td>
<td>Server response header field to send back to client. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>Prod send Server: Apache</td>
</tr>
<tr>
<td></td>
<td>Major send Server: Apache/2</td>
</tr>
<tr>
<td></td>
<td>Minor send Server: Apache/2.4</td>
</tr>
<tr>
<td></td>
<td>Minimal send Server: Apache/2.4.2</td>
</tr>
<tr>
<td></td>
<td>OS send Server: Apache/2.4.2 (Unix)</td>
</tr>
<tr>
<td></td>
<td>Full send Server: Apache/2.4.2 (Unix) PHP/4.2.2 MyMod/1.2 (default)</td>
</tr>
<tr>
<td>ServerSignature Off</td>
<td>Trailing footer line on server-generated documents. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>Off no footer line (default)</td>
</tr>
<tr>
<td></td>
<td>On server version number and ServerName</td>
</tr>
<tr>
<td></td>
<td>EMail as above, plus a mailto link to ServerAdmin</td>
</tr>
<tr>
<td>SSLVerifyClient none</td>
<td>Certificate verification level for client authentication. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>none no client certificate is required</td>
</tr>
<tr>
<td></td>
<td>require the client needs to present a valid certificate</td>
</tr>
<tr>
<td></td>
<td>optional the client may present a valid certificate (this option is unused as it doesn't work on all browsers)</td>
</tr>
<tr>
<td></td>
<td>optional_no_ca the client may present a valid certificate but it doesn't need to be successfully verifiable (this option is used in practice only for SSL testing)</td>
</tr>
<tr>
<td>TraceEnable on</td>
<td>Enable TRACE requests</td>
</tr>
</tbody>
</table>
Apache proxy

A **forward proxy** provides proxy services, typically web content caching and/or filtering, for clients located in a LAN. All outgoing requests from the clients, and the responses from the Internet, pass through the proxy. The clients must be manually configured (e.g. in the browser’s connection settings) to use the proxy.

<table>
<thead>
<tr>
<th>Apache configuration file</th>
<th>Forward proxy</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProxyRequests On</td>
<td>Enable forward proxy requests</td>
</tr>
<tr>
<td>ProxyVia On</td>
<td>Add a Via: HTTP header line to every request and reply</td>
</tr>
<tr>
<td>&lt;Proxy “*”&gt;</td>
<td>Serve only proxy requests coming from 10.1.1.0/24</td>
</tr>
<tr>
<td>Require ip 10.1.1</td>
<td></td>
</tr>
<tr>
<td>&lt;/Proxy&gt;</td>
<td></td>
</tr>
</tbody>
</table>

A **reverse proxy** aka gateway allows to expose a single entry point for one or more webservers in a LAN. This improves security and simplifies management, as features (e.g. load balancing, firewalling, automatic redirection from HTTP to HTTPS, redirection on default ports) can be configured centrally. It is necessary to create a DNS A record that maps site.example.com to the public IP address of the proxy.

<table>
<thead>
<tr>
<th>Apache configuration file</th>
<th>Reverse proxy</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;VirtualHost *:80&gt;</td>
<td>Virtual host for HTTP</td>
</tr>
<tr>
<td>ServerName site.example.com</td>
<td>Define website name</td>
</tr>
<tr>
<td>RewriteEngine On</td>
<td>Redirect all HTTP requests to HTTPS</td>
</tr>
<tr>
<td>RewriteCond %{HTTPS} off</td>
<td>Alternatively:</td>
</tr>
<tr>
<td>RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}</td>
<td>Redirect “/” “<a href="https://10.2.2.73:443/%E2%80%9D">https://10.2.2.73:443/”</a></td>
</tr>
<tr>
<td>&lt;/VirtualHost&gt;</td>
<td></td>
</tr>
<tr>
<td>&lt;VirtualHost *:443&gt;</td>
<td>Virtual host for HTTPS</td>
</tr>
<tr>
<td>ServerName site.example.com</td>
<td>Define website name</td>
</tr>
<tr>
<td>ServerSignature On</td>
<td>Set a footer line under server-generated pages</td>
</tr>
<tr>
<td>&lt;Proxy *&gt;</td>
<td>Serve all proxy requests</td>
</tr>
<tr>
<td>Require all granted</td>
<td></td>
</tr>
<tr>
<td>&lt;/Proxy&gt;</td>
<td></td>
</tr>
<tr>
<td>SSLEngine on</td>
<td>Enable and configure SSL</td>
</tr>
<tr>
<td>SSLProtocol ALL -SSLv2 -SSLv3</td>
<td></td>
</tr>
<tr>
<td>SSLHonorCipherOrder on</td>
<td></td>
</tr>
<tr>
<td>SSLCipherSuite DEFAULT</td>
<td></td>
</tr>
<tr>
<td>SSLCertificateFile /etc/httpd/ssl/site.crt</td>
<td></td>
</tr>
<tr>
<td>SSLCertificateKeyFile /etc/httpd/ssl/site.key</td>
<td></td>
</tr>
<tr>
<td>SSLLCACertificateFile /etc/httpd/ssl/site.ca.crt</td>
<td></td>
</tr>
<tr>
<td>ProxyPass ”/” “<a href="http://10.2.2.73:8080/%E2%80%9D">http://10.2.2.73:8080/”</a></td>
<td>Enable reverse proxying for server 10.2.2.73</td>
</tr>
<tr>
<td>ProxyPassReverse ”/” “<a href="http://10.2.2.73:8080/%E2%80%9D">http://10.2.2.73:8080/”</a></td>
<td></td>
</tr>
<tr>
<td>&lt;/VirtualHost&gt;</td>
<td></td>
</tr>
</tbody>
</table>
Tomcat is an open source Java Servlet Container implementing several Java EE specifications, and was originally part of the Jakarta Project. It is composed of:
- Catalina, the core component and servlet container implementation;
- Coyote, an HTTP connector component, providing a pure Java webservice environment to run Java code;
- Jasper, a JSP (Java Server Pages) engine, which parses JSP files and compiles them into Java servlets.

Tomcat may also be configured for multiple instances by defining the variable `$CATALINA_BASE` for each instance. If a single instance of Tomcat is running, `$CATALINA_BASE` is the same as `$CATALINA_HOME`.

<table>
<thead>
<tr>
<th>Tomcat global files</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>$CATALINA_BASE/conf/server.xml</code></td>
<td>Tomcat main configuration file</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/web.xml</code></td>
<td>Options and values applied to all web applications running on a specific Tomcat instance. These can be overridden by the application-specific servlet configuration defined in <code>$CATALINA_BASE/webapps/appname/WEB-INF/web.xml</code></td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/context.xml</code></td>
<td>Context applied to all web applications running on a specific Tomcat instance</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/tomcat-users.xml</code></td>
<td>Users, passwords, and roles applied to a specific Tomcat instance</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/catalina.policy</code></td>
<td>Tomcat’s core security policy for the Catalina class</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/logging.properties</code></td>
<td>Java properties file for the Catalina class</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/catalina.properties</code></td>
<td>Java properties file for Catalina’s built-in logging functions</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/conf/catalina.policy</code></td>
<td>Java properties file for Catalina’s built-in logging functions</td>
</tr>
<tr>
<td><code>$JAVA_HOME/jre/lib/security/keystore.jks</code></td>
<td>Java keystore</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tomcat application-specific files</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>$CATALINA_BASE/webapps/appname/WEB-INF/</code></td>
<td>HTML, JSP, and other files to serve to the client browser</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/webapps/appname/WEB-INF/web.xml</code></td>
<td>Description of servlets and other components of the application, and initialization parameters</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/webapps/appname/WEB-INF/classes/</code></td>
<td>Java class files that aren't in JAR format. The directory hierarchy from here reflects the class hierarchy</td>
</tr>
<tr>
<td><code>$CATALINA_BASE/webapps/appname/WEB-INF/lib/</code></td>
<td>Other JAR files (e.g. third-party libraries, JDBC drivers) required by the application</td>
</tr>
</tbody>
</table>

```
java -X                Display all available -X options (nonstandard HotSpot JVM options)
java -XshowSettings:properties -version Print Java runtime settings
```
Samba is a free-software, cross-platform implementation of SMB/CIFS. SMB (Server Message Block) is Microsoft’s proprietary protocol for file and printer sharing, while CIFS (Common Internet File System) is the public version of SMB.

<table>
<thead>
<tr>
<th>Commonly used ports in Samba</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP/UDP 137  netbios-ns</td>
</tr>
<tr>
<td>TCP/UDP 138  netbios-dgm</td>
</tr>
<tr>
<td>TCP/UDP 139  netbios-ssn</td>
</tr>
<tr>
<td>TCP 445     microsoft-ds</td>
</tr>
<tr>
<td>TCP 389     LDAP</td>
</tr>
<tr>
<td>TCP 901     SWAT service</td>
</tr>
</tbody>
</table>

The full list of used ports can be found via the command `grep -i netbios /etc/services`

**Samba server**

**smbd**  
Server Message Block daemon. Provides SMB file and printer sharing, browser services, user authentication, and resource lock. An extra copy of this daemon runs for each client connected to the server.

**nmbd**  
NetBIOS Name Service daemon. Handles NetBIOS name lookups, WINS requests, list browsing and elections. An extra copy of this daemon runs if Samba functions as a WINS server; another extra copy of this daemon runs if DNS is used to translate NetBIOS names.

WINS (Windows Internet Name Service) is a name service used to translate NetBIOS names to IP addresses.

**/etc/smb/ (RHEL 7)**  
**/etc/samba/**  
**/etc/samba/1mhosts**  
**/etc/samba/netlogon**  

- **smbd -V**  
  Show the version of the Samba server

- **smbclient -V**  
  
- **testparm**  
  Check the Samba configuration file and report any error

- **smbpasswd user**  
  Change the Samba password of *user*

- **smbpasswd -a user**  
  Create a new Samba *user* and set his password

- **nmblookup smbserver**  
  Look up the NetBIOS name of a server and map it to an IP address

- **nmblookup -U winsserver -R WORKGROUP#1B**  
  Query recursively a WINS server for the Domain Master Browser for the specified workgroup

- **nmblookup -U winsserver -R WORKGROUP#1D**  
  Query recursively a WINS server for the Domain Controller for the specified workgroup

**net**  
**net rpc shutdown -r -S smbserver -U root%password**  
Reboot a CIFS server

**net rpc service list -S smbserver**  
List available services on a CIFS server

**net status sessions**  
Show active Samba sessions

**net status shares**  
Show Samba shares

**net rpc info**  
Show information about the domain

**net groupmap list**  
Show group mappings between Samba and Windows
Samba client

```
mount.cifs
smbmount

mount //smbserver/share1 /mnt/share1 -t cifs -o username=user

Mount a Samba share as user

smbstatus

Display current information about shares, clients connections, and locked files

smbclient //smbserver/share1

Access a Samba share on a server (with an FTP-like interface)

smbclient -L //smbserver -W WORKGROUP -U user

List the Samba resources available on a server, belonging to the specified workgroup and accessible to the specified user

cat msg.txt | smbclient -M client -U user

Show a message popup on the client machine, using the WinPopup protocol

---

### Samba mount options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username=user</td>
<td>Mount the share as user</td>
</tr>
<tr>
<td>password=password</td>
<td>Specify the mount user's password</td>
</tr>
</tbody>
</table>
| credentials=credfile | Mount the share as the user defined in the credentials file credfile which must have this format:  
                       | username=user password=password                                              |
| multiuser         | Mount the share in multiuser mode                                           |
| sec=ntlmssp       | Set the security level to NTLMSSP. This is required in RHEL 7 to enable multiuser mode |
```
Samba global configuration

/etc/samba/smb.conf  Samba configuration

[global]

workgroup = MYWORKGROUP
server string = Linux Samba Server %L
hosts allow = 10.9.0/255.255.255.0
security = user
encrypt passwords = yes
smb passwd file = /etc/samba/smbpasswd

unix password sync = yes
username map = /etc/samba/smbusers

netbios name = Mysambabox	netbios aliases = Mysambabox1
wins support = yes
logon server = yes
log file = /var/log/samba/log.%m
max log size = 1000
syslog only = no
syslog = 0

panic action = \
/usr/share/samba/panic-action %d

[netlogon]

comment = Netlogon for Windows clients
path = /home/netlogon
logon script = %U.bat

browserable = no
writeable = no
guest ok = no

[Canon LaserJet 3]

printer name = lp
comment = Canon LaserJet 3 main printer
path = /var/spool/lpd/samba
printable = yes
writeable = no

Global server settings: defines parameters applicable for the whole Samba server and sets the defaults that will be used for the parameters not mentioned in other sections

Make Samba join the specified workgroup

Describe server to the clients

Set up user-level authentication

Use encrypted passwords

Refer to the specified password file for user authentication.
A new user’s password will need to be set both in Linux and Samba by using these commands from shell prompt:

    passwd newuser
    smbpasswd newuser

When the password of a client user (e.g. under Windows) is changed, change the Linux and Samba passwords accordingly

Map each Samba server user name to client user name(s).
The file /etc/samba/smbusers has the following format:

    root = Administrator Admin
    jdoe = "John Doe"
    kgreen = "Kim Green"

Set NetBIOS name and alias

Make Samba play the role of a WINS server.
Note: There should be only one WINS server on a network

Enable logon support.
Logon script parameters will be defined in a [netlogon] section

Use a separate logfile for each machine that connects

Maximum size of each logfile, in Kb

Do not use only syslog to log

Log everything to the logfiles /var/log/samba/log.smbd and /var/log/samba/log.nmbd, and log a minimum amount of information to syslog. This parameter can be set to a higher value to have syslog log more information

Mail a backtrace to the sysadmin in case Samba crashes

Section defining a logon script

Specifies a per-user script e.g. /home/netlogon/jdoo.bat will be called when user jdoe logs in.
It is also possible to specify a per-clientname script %m.bat, which will be called when a specific machine logs in.

Guest access to the service (i.e. access without entering a password) is disabled

Section defining a printer accessible via the network
### Samba share configuration

<table>
<thead>
<tr>
<th>Section</th>
<th>Configuration Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>[public]</td>
<td>Section defining a public share accessible on read/write by anyone.</td>
</tr>
<tr>
<td>comment = Public Storage on %L</td>
<td>Describe the public share to users.</td>
</tr>
<tr>
<td>path = /home/samba</td>
<td>Path of the public share on the server.</td>
</tr>
<tr>
<td>browsable = yes</td>
<td>Show the public share when browsing.</td>
</tr>
<tr>
<td>writeable = yes</td>
<td>Allow all users to write in this directory.</td>
</tr>
<tr>
<td>[homes]</td>
<td>Section enabling users that have an account and a home directory on the Samba server to access it and modify its contents from a Samba client.</td>
</tr>
<tr>
<td>comment = %U's home directory on %L from %m</td>
<td>The <code>path</code> variable is not set, by default is <code>path=/home/%U</code>.</td>
</tr>
<tr>
<td>browsable = no</td>
<td>Describe the share to the user.</td>
</tr>
<tr>
<td>writeable = yes</td>
<td>Do not show the homes share when browsing.</td>
</tr>
<tr>
<td>writeable = yes</td>
<td>Allow the user to write in his home directory.</td>
</tr>
<tr>
<td>[foobar]</td>
<td>Section defining a specific share.</td>
</tr>
<tr>
<td>path = /foobar</td>
<td>Path of the share on the server.</td>
</tr>
<tr>
<td>comment = Share Foobar on %L from %m</td>
<td>Describe the share to users.</td>
</tr>
<tr>
<td>browsable = yes</td>
<td>Show the share when browsing.</td>
</tr>
<tr>
<td>writeable = yes</td>
<td>Allow the users to write in this share.</td>
</tr>
<tr>
<td>valid users = jdoe, kgreen, +geeks</td>
<td>Allow access only to users &quot;jdoe&quot; and &quot;kgreen&quot;, and to local group &quot;geeks&quot;.</td>
</tr>
<tr>
<td>invalid users = csmith</td>
<td>Deny access to user &quot;csmith&quot;.</td>
</tr>
<tr>
<td>read list = bcameron</td>
<td>Allow read-only access to user &quot;bcameron&quot;.</td>
</tr>
<tr>
<td>write list = fcastle</td>
<td>Allow read-write access to user &quot;fcastle&quot;.</td>
</tr>
</tbody>
</table>
### Samba access configuration

#### /etc/samba/smb.conf

**User-level authentication**

```
[global]
  security = user
  guest account = nobody
  map to guest = Never
```

Set up user-level authentication

Map the guest account to the system user nobody (default)

Specify how incoming requests are mapped to the guest account:
- **Bad User**: redirect from an invalid user to guest account on server
- **Bad Password**: redirect from an invalid password to guest account on server
- **Never**: reject unauthenticated users

#### Server-level authentication

```
[global]
  security = server
  password server = srv1 srv2
```

Set up server-level authentication

Authenticate to server `srv1`, or to server `srv2` if the first one is unavailable

#### Domain-level authentication

```
[global]
  security = ADS
  realm = KR liberal_REALM
```

Set up domain-level authentication as an Active Directory member server

Join the specified realm.

Kerberos must be installed and an administrator account must be created:
```
net ads join -U Administrator% password
```

#### Share-level authentication

```
[global]
  security = share

[foobar]
  path = /foobar
  username = user
  only user = yes
```

Set up share-level authentication

Define a "foobar" share accessible to any user which can supply user's password.

The user must be created on the system:
```
useradd -c "Foobar account" -d /tmp -m -s /sbin/nologin user
```

and added to the Samba password file:
```
smbpasswd -a user
```

### Samba macros

| `%S` | Username |
| `%U` | Session username (the username that the client requested, not necessarily the same as the one he got) |
| `%G` | Primary group of session username |
| `%h` | Samba server hostname |
| `%M` | Client hostname |
| `%L` | NetBIOS name of the server |
| `%m` | NetBIOS name of the client |
| `%d` | Process ID of the current server process |
| `%a` | Architecture of remote machine |
| `%I` | IP address of client machine |
| `%i` | Local IP address to which a client connected |
| `%t` | Current date and time |
| `%D` | Domain or workgroup of the current user |
| `%w` | Winbind separator |
| `%{var}` | Value of the environment variable var |

The substitutes below apply only to the configuration options that are used when a connection has been established:

| `%S` | Name of the current service, if any |
| `%P` | Root directory of the current service, if any |
| `%u` | Username of the current service, if any |
| `%g` | Primary group name of username |
| `%h` | Home directory of username |
| `%N` | Name of the NIS home directory server as obtained from the NIS auto.map entry. Same as `%L` if Samba was not compiled with the --with-automount option |
| `%p` | Path of service's home directory as obtained from the NIS auto.map entry. The NIS auto.map entry is split up as `%N`: `%p` |
## Samba setup

This procedure allows sharing on read-write the local directory `/smbshare` on server 10.1.1.1 to client 10.2.2.2.

### Server setup:

1. Create the group for write access to the share
   
   `groupadd -r geeks`

2. Create the user and assign it to the group
   
   `useradd -G geeks jdoe`

3. Add the user to Samba.
   You will be prompted to enter a password
   
   `smbpasswd -a jdoe`

4. Assign correct ownership to the share
   
   `chgrp geeks /smbshare`

5. Set the SGID bit to the share
   
   `chmod 2775 /smbshare`

6. Set the correct SELinux label to the share
   
   `semanage fcontext -a -t samba_share_t '/smbshare'

   `restorecon -FR /smbshare`

7. Enable the SELinux boolean for write access to the share
   
   `setsebool -P samba_export_all_rw=on`

8. Add a section for the share on `/etc/samba/smb.conf`:

   ```
   [smbshare]
   path = /smbshare
   hosts allow = 10.2.2.2
   write list = @geeks
   ```

9. Ensure that the `smb` and `nmb` services are running

### Client setup:

1. Add an entry to `/etc/fstab` to mount the Samba share device automatically:

   ```
   //10.1.1.1/smbshare /mountpoint cifs username=jdoe,password=s3cr3t 0 0
   ```

### Client multiuser setup:

1. Add an entry to `/etc/fstab` to mount the Samba share device automatically in multiuser mode:

   ```
   //10.1.1.1/smbshare /mountpoint cifs username=jdoe,password=s3cr3t,multiuser,sec=ntlmssp 0 0
   ```

2. Login as another user (there must be a matching Samba user on the Samba server 10.1.1.1)
   
   `su - ksmith`

3. Store the Samba username and password in the kernel keyring for the current session
   
   `cifscreds add 10.1.1.1`
A Network File System (NFS) server makes filesystems available to remote clients for mounting.

NFS requires the portmapper to map incoming TCP/IP connections to the appropriate NFS RPC calls. Some Linux distributions use rpcbind instead of the portmapper. For security reasons, the TCP Wrapper should be configured to limit access to the portmapper to NFS clients only:

- File `/etc/hosts.deny` should contain `portmap: ALL`
- File `/etc/hosts.allow` should contain `portmap: IP_addresses_of_clients`

NFS handles user permissions across systems by considering users with same UID and username as the same user. Group permission is evaluated similarly, by GID and groupname.

**rpc.nfsd**
**rpc.mountd**
**rpc.lockd**
**rpc.statd**

**/etc/exports**
**/var/lib/nfs/xtab**
**/proc/fs/nfs/exports**

**exportfs -ra**
Export or reexport all directories.
When exporting, fills the kernel export table `/proc/fs/nfs/exports`
When reexporting, removes the entries in `/var/lib/nfs/xtab` that are deleted from `/etc/exports` (therefore synchronizing the two files), and removes the entries from `/proc/fs/nfs/exports` that are no longer valid.

**exportfs -ua**
Unexport all directories.
Removes from `/proc/fs/nfs/exports` the entries that are listed in `/var/lib/nfs/xtab`, and clears the latter file.

**showmount**
**showmount --directories**
Show the remote client hosts currently having active mounts
**showmount --exports**
Show the directories currently mounted by a remote client host
**showmount --all**
Show both remote client hosts and directories
**showmount -e nfsserver**
Show the shares a NFS server has available for mounting

**rpcinfo -p nfsserver**
Probe the portmapper on a NFS server and display the list of all registered RPC services there
**rpcinfo -t nfsserver nfs**
Test a NFS connection by sending a null pseudo request (using TCP)
**rpcinfo -u nfsserver nfs**
Test a NFS connection by sending a null pseudo request (using UDP)

**nfsstat**
Display NFS/RPC client/server statistics.

<table>
<thead>
<tr>
<th>Options</th>
<th>NFS</th>
<th>RPC</th>
<th>both</th>
</tr>
</thead>
<tbody>
<tr>
<td>server</td>
<td>-sn</td>
<td>-sr</td>
<td>-s</td>
</tr>
<tr>
<td>client</td>
<td>-cn</td>
<td>-cr</td>
<td>-c</td>
</tr>
<tr>
<td>both</td>
<td>-n</td>
<td>-r</td>
<td>-nr</td>
</tr>
</tbody>
</table>

**mount -t nfs nfsserver:/share /usr**
Command to be run on a client to mount locally a remote NFS share.
NFS shares accessed frequently should be added to `/etc/fstab` e.g. `nfsserver:/share /usr nfs intr 0 0`
## /etc/exports

**Filesystem on the NFS server to be exported to clients**

<table>
<thead>
<tr>
<th>Client Identity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filesystem</td>
<td>Client systems permitted to access the exported directory. Can be specified by hostname, IP address, wildcard, subnet, or @NIS workgroup. Multiple client systems can be listed, and each one can have different options</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Client Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ro</td>
<td>Read-only access (default)</td>
</tr>
<tr>
<td>rw</td>
<td>Read and write access. The client may choose to mount read-only anyway</td>
</tr>
<tr>
<td>sync</td>
<td>Reply to requests only after the changes made by these requests have been committed to stable storage</td>
</tr>
<tr>
<td>async</td>
<td>Reply to requests without waiting that changes are committed to stable storage. Improves performances but might cause loss or corruption of data if server crashes</td>
</tr>
<tr>
<td>root_squash</td>
<td>Requests by user root on client will be done as user nobody on server (default)</td>
</tr>
<tr>
<td>no_root_squash</td>
<td>Requests by user root on client will be done as same user root on server</td>
</tr>
<tr>
<td>all_squash</td>
<td>Requests by a non-root user on client will be done as user nobody on server</td>
</tr>
<tr>
<td>no_all_squash</td>
<td>Requests by a non-root user on client will be attempted as same user on server (default)</td>
</tr>
</tbody>
</table>

### NFS mount options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rsize=nnn</td>
<td>Size for read transfers (from server to client)</td>
</tr>
<tr>
<td>wsize=nnn</td>
<td>Size for write transfers (from client to server)</td>
</tr>
<tr>
<td>nfsvers=n</td>
<td>Use NFS version n for transport</td>
</tr>
<tr>
<td>retry=n</td>
<td>Keep retrying a mount attempt for n minutes before giving up</td>
</tr>
<tr>
<td>timeo=n</td>
<td>A mount attempt times out after n tenths of a second</td>
</tr>
<tr>
<td>intr</td>
<td>User can interrupt a mount attempt</td>
</tr>
<tr>
<td>nointr</td>
<td>User cannot interrupt a mount attempt (default)</td>
</tr>
<tr>
<td>hard</td>
<td>The system will try a mount indefinitely (default)</td>
</tr>
<tr>
<td>soft</td>
<td>The system will try a mount until an RPC timeout occurs</td>
</tr>
<tr>
<td>bg</td>
<td>Try a mount in the foreground; all retries occur in the background</td>
</tr>
<tr>
<td>fg</td>
<td>All mount attempts occur in the background (default)</td>
</tr>
<tr>
<td>tcp</td>
<td>Connect using TCP</td>
</tr>
<tr>
<td>udp</td>
<td>Connect using UDP</td>
</tr>
<tr>
<td>sec=krb5p</td>
<td>Use Kerberos to encrypt all requests between client and server</td>
</tr>
<tr>
<td>v4.2</td>
<td>Enable NFS v4.2, which allows the server to export the SELinux context</td>
</tr>
</tbody>
</table>
## NFS setup

This procedure allows sharing on read-write the local directory `/nfsshare` on server 10.1.1.1 to client 10.2.2.2.

**Server setup:**
1. Ensure that the `nfs-server` service is running
2. Change ownership of the share
   ```bash
   chown nfsnobody /nfsshare
   ```
3. Add an entry for the share on `/etc/exports`:
   ```bash
   /nfsshare  10.2.2.2(rw)
   ```
4. Reload the exports file
   ```bash
   exportfs -r
   ```

**Client setup:**
1. Add an entry to `/etc/fstab` to mount the NFS share device automatically:
   ```bash
   10.1.1.1:/nfsshare  /mountpoint  nfs  defaults  0 0
   ```

## Secure NFS setup

This procedure allows sharing on read-write the local directory `/nfsshare` on server 10.1.1.1 to client 10.2.2.2, securely with Kerberos enabled.

**Server setup:**
1. Install the appropriate server keytab on `/etc/krb5.keytab`
2. Ensure that the `nfs-secure-server` service is running
3. Change ownership of the share
   ```bash
   chown nfsnobody /nfsshare
   ```
4. Add an entry for the share on `/etc/exports`:
   ```bash
   /nfsshare  10.2.2.2(sec=krb5p,rw)
   ```
5. Reload the exports file
   ```bash
   exportfs -r
   ```

**Client setup:**
1. Install the appropriate client keytab on `/etc/krb5.keytab`
2. Ensure that the `nfs-secure` service is running
3. Add an entry to `/etc/fstab` to mount the NFS share device automatically:
   ```bash
   10.1.1.1:/nfsshare  /mountpoint  nfs defaults,sec=krb5p  0 0
   ```
iSCSI (Internet Small Computer System Interface) is a network protocol that allows emulating an SCSI local storage device over a TCP/IP network. By default it uses TCP port 3260. An iSCSI server can use a local block device (physical or virtual disk, disk partition, or Logical Volume), a file, a physical SCSI device, or a ramdisk as the underlying storage resource (backstore) and make it available by assigning it a LUN (Logical Unit Number). An iSCSI server provides one or more targets, each of which presents one or more LUNs and is able to accept connections from an iSCSI client (initiator). Targets and initiators are called nodes and are identified by a unique IQN (iSCSI Qualified Name) e.g. iqn.2017-11.org.example.subdomain:foo:bar. The IP address and port of a node is called a portal. A target accepts connections from an initiator via a TPG (Target Portal Group) i.e. its IP address and port. A TPG may have in place an ACL so to accept connections only from a specific initiator's IQN.

```
targetcli  Target configurator (server side). Can be used as a command line tool or as an interactive shell. Configuration is saved to /etc/target/saveconfig.json

iscsiadm   Administration tool for iSCSI devices (client side)
```
This procedure makes available the local disk /dev/sbd on server 10.1.1.1 to the client having IQN iqn.2017-11.org.example:client.

Server (target) setup:
1. Ensure that the targetcli service is running

2. Enter the targetcli shell

3. Create a backstore

4. Create a IQN for the target. This automatically creates a TPG for the IQN

5. On the TPG, create an ACL to allow connections from the initiator with a specific IQN

6. On the TPG, create a LUN for the backstore

7. On the TPG, create a portal listening from the server’s IP address

8. Verify the configuration

   ls /
   o- / .......................................................... [..]
   o- block .................................................. [Storage Objects: 1]
   o- mydisk ........................................ [/dev/sdb (100.0MiB) write-thru activated]
   o- alua ............................................. [ALUA Groups: 1]
   o- default_tg_pt_gp .............................. [ALUA state: Active/optimized]
   o- fileio ............................................ [Storage Objects: 0]
   o- pscsi ........................................... [Storage Objects: 0]
   o- ramdisk ......................................... [Storage Objects: 0]
   o- iscsi ........................................... [Targets: 1]
   o- tpgl ........................................... [no-gen-acls, no-auth]
   o- acls ............................................. [ACLs: 1]
   o- mapped_lun0 ................................... [Mapped LUNs: 1]
   o- lun0 ........................................... [lun0 block/mydisk (rw)]
   o- portals ......................................... [Portals: 1]
   o- 10.1.1.1:3260 ................................... [OK]
   o- loopback ........................................ [Targets: 0]

9. Exit the targetcli shell.
   Configuration is automatically saved

Client (initiator) setup:
1. Set the correct initiator IQN in the file /etc/iscsi/initiatorname.iscsi:

   InitiatorName=iqn.2017-11.org.example:client

2. Ensure that the iscsi service is running

3. Discover the iSCSI target(s) provided by the portal. This echoes the target(s) IQN found

4. Login to the target IQN found

   iscsiadm -m node -t iqn.2017-11.org.example:target -p 10.1.1.1 -l

   The iSCSI device is now locally available and can be formatted and mounted. Node records remain after logout or reboot; the system will login again to the target IQN automatically

5. Add an entry to /etc/Fstab to mount the iSCSI device automatically:

   UUID=nnnnnnn-nnnn-nnnn-nnnn-nnnnnnnnnnnn /mountpoint fsype _netdev 0 0
DHCP (Dynamic Host Configuration Protocol) is a protocol for network management that automatically provides a requesting host with an IP address and other network configuration parameters. It largely supersedes BOOTP (Bootstrap Protocol). A DHCP server listens for requests on UDP port 67 and answers to UDP port 68. The assignment of an IP address to a host is done through a sequence of DHCP messages initiated by the client host: DHCP Discover, DHCP Offer, DHCP Request, and finally DHCP Acknowledgment.

Because DHCP Discover messages are broadcast and therefore not routed outside a LAN, a DHCP relay agent is necessary for those clients situated outside the DHCP server’s LAN. The DHCP relay agent listens to DHCP Discover messages and relays them in unicast to the DHCP server.

/etc/dhcpd.conf
/etc/sysconfig/dhcrelay (SUSE)
/var/lib/dhcpd/dhcpd.leases

<table>
<thead>
<tr>
<th>/etc/dhcpd.conf</th>
<th>DHCP server configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>option domain-name-servers 10.2.2.2;</td>
<td>Global parameters for DNS, mail, NTP, and news servers specification</td>
</tr>
<tr>
<td>option smtp-servers 10.3.3.3;</td>
<td></td>
</tr>
<tr>
<td>option pop-servers 10.4.4.4;</td>
<td></td>
</tr>
<tr>
<td>option time-servers 10.5.5.5;</td>
<td></td>
</tr>
<tr>
<td>option nntp-servers 10.6.6.6;</td>
<td></td>
</tr>
<tr>
<td>shared-network geek-net {</td>
<td>Definition of a network</td>
</tr>
<tr>
<td>default-lease-time 86400;</td>
<td>Time, in seconds, that will be assigned to a lease if a client does not ask for a specific expiration time</td>
</tr>
<tr>
<td>max-lease-time 172800;</td>
<td>Maximum time, in seconds, that can be assigned to a lease if a client asks for a specific expiration time</td>
</tr>
<tr>
<td>option routers 10.0.3.252;</td>
<td></td>
</tr>
<tr>
<td>option broadcast-address 10.0.3.255;</td>
<td></td>
</tr>
<tr>
<td>subnet 10.0.3.0 netmask 255.255.255.128 {</td>
<td>Definition of different subnets in the network, with specification of different ranges of IP addresses that will be leased to clients depending on the client's subnet</td>
</tr>
<tr>
<td>range 10.0.3.1 10.0.3.101;</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>subnet 10.0.3.128 netmask 255.255.255.128 {</td>
<td></td>
</tr>
<tr>
<td>range 10.0.3.129 10.0.3.229;</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>group {</td>
<td>Definition of a group</td>
</tr>
<tr>
<td>option routers 10.0.17.252;</td>
<td></td>
</tr>
<tr>
<td>option broadcast-address 10.0.17.255;</td>
<td></td>
</tr>
<tr>
<td>netmask 255.255.255.0;</td>
<td></td>
</tr>
<tr>
<td>host linuxbox1 {</td>
<td>Definition of different hosts to whom static IP addresses will be assigned to, depending on their MAC address</td>
</tr>
<tr>
<td>hardware ethernet AA:BB:CC:DD:EE:FF;</td>
<td></td>
</tr>
<tr>
<td>fixed-address 10.0.17.42;</td>
<td></td>
</tr>
<tr>
<td>option host-name &quot;linuxbox1&quot;;</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>host linuxbox2 {</td>
<td></td>
</tr>
<tr>
<td>fixed-address 10.0.17.66;</td>
<td></td>
</tr>
<tr>
<td>option host-name &quot;linuxbox2&quot;;</td>
<td></td>
</tr>
</tbody>
</table>
PAM (Pluggable Authentication Modules) is an abstraction layer that allows applications to use authentication methods while being implementation-agnostic.

/etc/pam.d/service  PAM configuration for service
/etc/pam.conf  (obsolete)  PAM configuration for all services

`ldd /usr/sbin/service | grep libpam`  Check if service is enabled to use PAM

<table>
<thead>
<tr>
<th>/etc/pam.d/service</th>
</tr>
</thead>
<tbody>
<tr>
<td>auth requisite pam_securetty.so</td>
</tr>
<tr>
<td>auth required pam_nologin.so</td>
</tr>
<tr>
<td>auth required pam_unix.so nullok</td>
</tr>
<tr>
<td>account required pam_unix.so</td>
</tr>
<tr>
<td>session required pam_unix.so</td>
</tr>
<tr>
<td>session optional pam_lastlog.so</td>
</tr>
<tr>
<td>password required pam_unix.so nullok obscure min=4 max=8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>auth</td>
<td>Authentication module to verify user identity and group membership</td>
</tr>
<tr>
<td>account</td>
<td>Authorization module to determine user's right to access a resource (other than his identity)</td>
</tr>
<tr>
<td>password</td>
<td>Module to update a user's authentication credentials</td>
</tr>
<tr>
<td>session</td>
<td>Module (run at end and beginning of a user session) to set up the user environment</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>optional</td>
<td>Module is not critical to the success or failure of service</td>
</tr>
<tr>
<td>sufficient</td>
<td>If this module successes, and no previous module has failed, module stack processing ends successfully. If this module fails, it is non-fatal and processing of the stack continues</td>
</tr>
<tr>
<td>required</td>
<td>If this module fails, processing of the stack continues until the end, and service fails</td>
</tr>
<tr>
<td>requisite</td>
<td>If this module fails, service fails and control returns to the application that invoked service</td>
</tr>
<tr>
<td>include</td>
<td>Include modules from another PAM service file</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pam_unix.so</td>
<td>Standard UNIX authentication module via /etc/passwd and /etc/shadow</td>
</tr>
<tr>
<td>pam_nis.so</td>
<td>Module for authentication via NIS</td>
</tr>
<tr>
<td>pam_ldap.so</td>
<td>Module for authentication via LDAP</td>
</tr>
<tr>
<td>pam_fshadow.so</td>
<td>Module for authentication against an alternative shadow passwords file</td>
</tr>
<tr>
<td>pam_cracklib.so</td>
<td>Module for password strength policies (e.g. length, case, max number of retries)</td>
</tr>
<tr>
<td>pam_limits.so</td>
<td>Module for system policies and system resource usage limits</td>
</tr>
<tr>
<td>pam_listfile.so</td>
<td>Module to deny or allow the service based on an arbitrary text file</td>
</tr>
</tbody>
</table>
LDAP (Lightweight Directory Access Protocol) is a simplified version of the X.500 standard and uses TCP port 389. LDAP permits organizing hierarchically a database of entries, each one of which is identified by a unique DN (Distinguished Name). Each DN has a set of attributes, each one of which has a value. An attribute may appear multiple times.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Example</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>cn</td>
<td>cn: John Doe</td>
<td>Common Name</td>
</tr>
<tr>
<td>dc</td>
<td>dc=example,dc=org</td>
<td>Domain Component</td>
</tr>
<tr>
<td>givenName</td>
<td>givenName: John</td>
<td>First name</td>
</tr>
<tr>
<td>sn</td>
<td>sn: Doe</td>
<td>Surname</td>
</tr>
<tr>
<td>mail</td>
<td>mail: <a href="mailto:jdoe@example.org">jdoe@example.org</a></td>
<td>Email address</td>
</tr>
<tr>
<td>telephoneNumber</td>
<td>telephoneNumber: +1 555 1234 567</td>
<td>Telephone number</td>
</tr>
<tr>
<td>uid</td>
<td>uid: jdoe</td>
<td>User ID</td>
</tr>
<tr>
<td>c</td>
<td>c: US</td>
<td>Country code</td>
</tr>
<tr>
<td>l</td>
<td>l: San Francisco</td>
<td>Locality</td>
</tr>
<tr>
<td>st</td>
<td>st: California</td>
<td>State or province</td>
</tr>
<tr>
<td>street</td>
<td>street: 42, Penguin Road</td>
<td>Street</td>
</tr>
<tr>
<td>o</td>
<td>o: The Example Foundation</td>
<td>Organization</td>
</tr>
<tr>
<td>ou</td>
<td>ou: IT Dept</td>
<td>Organizational Unit</td>
</tr>
<tr>
<td>manager</td>
<td>manager: cn=Kim Green,dc=example,dc=org</td>
<td>Manager</td>
</tr>
</tbody>
</table>

Example commands:

```
ldapsearch -H ldap://ldapserver.example.org -s base -b "ou=people,dc=example,dc=com" "(sn=Doe)" cn sn telephoneNumber
```

Query the specified LDAP server for entries in the OU "people" whose surname is "Doe", and print common name, surname, and telephone number of the entries found. Output is shown in LDIF.

```
ldappasswd -x -D "cn=Admin,dc=example,dc=org" -W -S "uid=jdoe,ou=IT Dept,dc=example,dc=org"
```

Authenticating as "Admin" on example.org, change the password of user "jdoe" in the OU "IT Dept".

```
ldapmodify -b -r -f file.ldif
```

Modify an entry according to the LDIF file specified.

```
ldapadd -h ldapserver.example.org -D "cn=Admin" -W -f file.ldif
```

Authenticating as "Admin", add an entry by adding the content of the specified LDIF file to the directory. This command actually invokes `ldapmodify -a`.

```
ldapdelete -v "uid=jdoe,dc=example,dc=org" -D "cn=Admin,dc=example,dc=org" -W
```

Authenticating as "Admin", delete the user "jdoe".

LDIF (LDAP Data Interchange Format):

```
dn: cn=John Doe, dc=example, dc=org
changetype: modify
replace: mail
mail: johndoe@otherexample.com
-add: jpegPhoto
jpegPhoto:< file://tmp/jdoe.jpg
-delete: description
```

This LDIF file will change the email address of user "jdoe", add a picture, and delete the description attribute for the entry.
slapd is the Standalone OpenLDAP daemon. It was initially developed together with the LDAP protocol. To provide access to OpenLDAP as an authentication and identity provider, sssd (the System Security Services Daemon) must be running.

/var/lib/ldap/

/etc/openldap/slapd.conf
/usr/local/etc/openldap/slapd.conf

Files constituting the OpenLDAP database

OpenLDAP configuration file

slapcat -l file.ldif

Dump the contents of an OpenLDAP database to an LDIF file

slapadd -l file.ldif

Import an OpenLDAP database from an LDIF file

slapindex

Regenerate OpenLDAP's database indexes

yum install openldap openldap-clients \ authconfig sssd nss-pam-ldapd authconfig-gtk

Install the OpenLDAP client (on RHEL 7)

authconfig --enableldap --enableldapauth \ --ldapserver=ldap://ldapserver \ --ldapbasedn="dc=example,dc=org" \ --enablesssd --update

Set up the LDAP client to connect to a ldapserver. This will update the configuration files /etc/sssd/sssd.conf and /etc/openldap/ldap.conf

getent group groupname

Get entries about groupname from NSS libraries

authconfig-gtk
system-config-authentication

OpenLDAP configuration GUI
Security-Enhanced Linux (SELinux) is a Linux kernel security module that provides a mechanism for supporting access control security policies.

SELinux implements a Mandatory Access Control framework that allows the definition of fine-grained permissions for how subjects (i.e. processes) interact with objects (i.e. other processes, files, devices, ports, sockets); this improves security with respect to the standard Discretionary Access Control, which defines accesses based on users and groups. The security context of a file is stored in its extended attributes.

The decisions SELinux takes about allowing or disallowing access are stored in the AVC (Access Vector Cache).

```
setenforce 0
```

Enter permissive mode

```
setenforce 1
echo 1 > /selinux/enforce
```

Enter enforcing mode

```
cat /selinux/enforce
getenforce
```

Display current mode

SELinux mode can be configured permanently in `/etc/selinux/config` (symlinked in `/etc/sysconfig/selinux`):

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
# enforcing - SELinux security policy is enforced.
# permissive - SELinux prints warnings instead of enforcing.
# disabled - No SELinux policy is loaded.

SELINUX=enforcing

SELINUXTYPE= can take one of these two values:
# targeted - Only targeted network daemons are protected.
# strict - Full SELinux protection.

SELINUXTYPE=targeted
```

```
chcon context file
chcon --reference=file0 file
restorecon -f file
ls -Z
ps -eZ
```

Change the security context of `file` to the specified `context`

Change the security context of `file` to be the same as `file0`

Restore the security context of `file` to the system default

```
ls -Z
ps -eZ
```

List files and their security context

List processes and their security context

```
semanage fcontext -1
semanage fcontext -a -t label file
semanage port -1
semanage port -a -t portlabel -p tcp n
semanage port -a -t http_port_t -p tcp 8888
semanage port -d -t http_port_t -p tcp 8888
semanage port -m -t http_cache_port_t -p tcp 8888
```

Manage SELinux policies

List files and their assigned SELinux labels

Assign the SELinux `label` to `file`.

You then need to apply the label via `restorecon -f file`

List port numbers and their assigned SELinux type definitions

Assign the SELinux `portlabel` to TCP port `n`

Allow a local webserver to serve content on port 8888

Remove the binding of `http_port_t` port label to TCP 8888

Modify the port label bound to TCP 8888

```
getsebool boolean
setsebool boolean=value
```

Get the value of a SELinux boolean

Set the value of a SELinux boolean

```
tar --selinux [other args]
tar -xattr -H=ext2star [other args]
```

Create or extract archives that retain the security context of the original files
/selinux/

Pseudo filesystem created by SELinux, containing commands used by the kernel for its operations

/var/log/audit/audit.log
/var/log/messages

Logfile containing AVC denials, if `auditd` is running
Logfile containing AVC denials, if `rsyslogd` is running

```
sealert -a logfile
grep nnnnn.mmm:pp logfile | audit2why
```

Analyze a SELinux logfile and display SELinux policy violations
Diagnostic a specific AVC event entry from a SELinux logfile.
The event appears in the logfile as
```plaintext
type=AVC msg=audit(nnnnn.mmm:pp): avc: denied (...)
```
KVM (Kernel-based Virtual Machine) is a virtualization infrastructure for the Linux kernel that allows it to function as a hypervisor.

/etc/libvirt/qemu/  
Directory containing the XML files that define VMs properties.
libvirtd must be restarted after modifying an XML file

/var/lib/libvirt/  
Directory containing files related to the VMs

virt-manager

virt-install --prompt
virt-install -n vmname -r 2048 \
--disk path=/var/lib/libvirt/images/vmname.img \ 
-l /root/vmstuff/inst/ \ 
-x "ks=/root/vmstuff/kickstart.cfg"

virt-clone --prompt
virt-clone -o vmname -n vmclonename

virsh
virsh list --all
virsh start vmname
virsh destroy vmname
virsh shutdown vmname
virsh autostart vmname

virsh autostart --disable vmname
virsh edit vmname

KVM GUI

virt-what

Kickstart

Kickstart is a method to perform automatic installation and configuration of RHEL machines. This can be done by specifying \inst.ks=bd:dev/sda:root/path/ksfile\ either as a boot option, or an option to the kernel command in GRUB 2.

system-config-kickstart  
GUI tool to create a Kickstart file

ksvalidator ksfile  
Check the validity of a Kickstart file

/anaconda-ks.cfg  
Kickstart file describing the current system, automatically generated during the installation

ksverdiff -f RHEL6 -t RHEL7  
Show the differences in the Kickstart syntax between RHEL 6 and RHEL 7
Git is an open source version control system with a small footprint and very high performances. A Git directory is a complete repository with full history and version tracking abilities, independent of any remote repository.

- **git init**
  Initialize the current directory as a repository

- **git clone repoaddress**
  Clone a remote repository. *repoaddress* can be a URL (SSH, HTTP, HTTPS, FTP, FTPS, Git) or a local path e.g. `ssh://user@example.com:8888/path/to/repo.git` or `git://example.com:9999/path/to/repo.git` or `/path/to/repo.git`

- **git checkout branch**
  Start working into an already existing *branch*

- **git checkout -B branch**
  Create *branch* and start working into it

- **git pull**
  Pull the changes from the remote repository branch to the local branch

- **git add file**
  Add *file* to the content staged for the next commit (hence starting to track it)

- **git rm file**
  Remove *file* from the content staged for the next commit

- **git status**
  See the status (e.g. files changed but not yet staged) of the current branch

- **git commit -m "Message"**
  Commit all staged files in the current branch

- **git commit -am "Message"**
  Add all changed files to the staging area in the current branch, and commit them

- **git push**
  Push the local commits from the current branch to the remote repository

- **git push origin branch**
  Push the local commits from *branch* to the remote repository

- **git merge branch**
  Merge changes made on *branch* to the master branch

- **git diff checksum1 checksum2**
  Compare two commits

- **git log -Gword**
  Show the commits whose added or deleted lines contain *word*

- **git branch**
  Show local branches

- **git branch -r**
  Show remote branches

- **git branch -a**
  Show remote and local branches

- **git config user.name name**
  Set your username in the Git configuration

- **git config user.email email**
  Set your email address in the Git configuration

- **git config option**
  Get the value of *option* in the Git configuration

- **git config --list**
  Get all currently set options and their values in the Git configuration
Vagrant is an open source software that allows building and maintaining lightweight and portable virtual environments for software development. It relies on an underlying virtualization solution e.g. VirtualBox.

```
vagrant -h
vagrant command -h
vagrant init hashicorp/precise64
vagrant up vmname
vagrant provision vmname
vagrant ssh vmname
vagrant halt vmname
vagrant destroy vmname
vagrant status
vagrant global-status
vagrant global-status --prune
```

Print the list of commands recognized by Vagrant

Print help about the Vagrant command

Initialize the current directory as a specific Vagrant environment (in this case, Ubuntu 12.04 64-bit) by creating a Vagrantfile on it

Start a guest virtual machine and do a first provisioning according to the Vagrantfile

Provision a virtual machine

Connect via SSH to a virtual machine

Shut down the virtual machine

Delete the virtual machine and free any resource allocated to it

Print the status of the virtual machines currently managed by Vagrant

Print the status of all Vagrant environments on the system, by reading cached data. Completes quickly but results may be outdated

Print the status of all Vagrant environments on the system, after rebuilding the environment information cache. Results are always correct but completion takes longer

The directory containing the Vagrantfile on the host can be accessed on the guest via /vagrant.
Puppet is a software configuration management tool. It is based on a client-server architecture, where a Puppet agent (client, running as root on each managed node) periodically gathers information (facts) about the local node state via the Facter tool, then communicates this information to the Puppet master (server, running as the puppet user and listening on TCP port 8140). The Puppet master then sends back to the Puppet agent a catalog containing the desired configuration for that node. The Puppet agent applies the needed changes so that the node's configuration converges with the desired configuration, and sends back a report to the Puppet master. Puppet changes are idempotent.

Puppet configurations are based on resources (e.g. "package", "service", "file", "user"). For each resource, a list of attributes is specified, with the desired value for each attribute. Each resource type is implemented through providers (e.g. yum, rpm, apt, opkg ... for the resource "package"). Resources managed together as a single unit can be grouped into classes; classes are contained in manifests which are files with the .pp extension. Modules are directories containing self-contained pieces of configuration and classes for a specific complex setting, e.g. an Apache webserver or a MySQL server.

```bash
/etc/puppet/puppet.conf
/etc/puppetlabs/puppet/puppet.conf
```

Configuration file (Puppet free)
Configuration file (Puppet Enterprise)

```bash
facter
```

Gather the facts about the managed node, and return a list of key-value pairs

```bash
puppet agent
```

Main Puppet client. Retrieves the node's desired configuration from the Puppet master and applies it

```bash
puppet agent --disable
puppet agent --enable
puppet agent --noop
```

Disable or enable the Puppet agent on the node
Perform a dry run, displaying the changes that Puppet would have applied without actually applying them

```bash
puppet resource user username
puppet resource service httpd enable=false
```

Inspect the state of the resource "user" with respect to username Modify directly the state of the resource "service" (in this case, disable the HTTP server)

```bash
puppet describe user
puppet describe --list
puppet describe user --providers
puppet apply modulename/init.pp
puppet cert operation
```

Show information about the resource "user"
List all resource types
Return the list of providers for the resource "user"
Apply a manifest one time only
Manage the SSL certificates used for communications between master and agents
Ansible is an open source tool for configuration management and software provisioning. It is agentless and connects to the managed machines via SSH pubkey authentication. It only requires OpenSSH and Python to be installed on the managed nodes.

The configuration for managed nodes is specified in one or more playbook, written in YAML and containing a number of tasks. When a playbook is run, first it collects system and environment information (facts) which is then stored in multiple variables named ansible_varname.

/etc/ansible/hosts

Inventory file, containing the list of hosts managed by Ansible. Can be in INI or YAML format

ansible hosts -m module options

Apply the options concerning module to the specified hosts

ansible-playbook options playbook.yml

Apply the specified playbook
<table>
<thead>
<tr>
<th>Tag</th>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;h1&gt;</code> ... <code>&lt;h6&gt;</code></td>
<td>Heading</td>
<td>Heading alignment †</td>
</tr>
<tr>
<td><code>&lt;br&gt;</code></td>
<td>Line break and carriage return</td>
<td>Line alignment †</td>
</tr>
<tr>
<td><code>&lt;hr&gt;</code></td>
<td>align=left</td>
<td>center</td>
</tr>
<tr>
<td><code>&lt;p&gt;</code></td>
<td>Line break and carriage return</td>
<td>Solid rendering instead of 3D †</td>
</tr>
<tr>
<td><code>&lt;div&gt;</code></td>
<td>Paragraph or section alignment †</td>
<td>Paragraph or section alignment †</td>
</tr>
<tr>
<td><code>&lt;span&gt;</code></td>
<td>Group</td>
<td>Paragraph or section alignment †</td>
</tr>
<tr>
<td><code>&lt;a&gt;</code></td>
<td>Anchor</td>
<td>Hyperlink</td>
</tr>
<tr>
<td><code>&lt;dt&gt;</code></td>
<td>Definition term</td>
<td>Description of a definition term</td>
</tr>
<tr>
<td><code>&lt;dd&gt;</code></td>
<td>Definition description</td>
<td>Description of a definition term</td>
</tr>
<tr>
<td><code>&lt;ol&gt;</code></td>
<td>Order list</td>
<td>List must be more compact †</td>
</tr>
<tr>
<td><code>&lt;ul&gt;</code></td>
<td>Unordered list</td>
<td>List type</td>
</tr>
<tr>
<td><code>&lt;li&gt;</code></td>
<td>List item</td>
<td>List item type</td>
</tr>
</tbody>
</table>

† = deprecated
<table>
<thead>
<tr>
<th>Tag</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;i&gt;</code></td>
<td>Italic</td>
</tr>
<tr>
<td><code>&lt;b&gt;</code></td>
<td>Bold</td>
</tr>
<tr>
<td><code>&lt;strike&gt;</code></td>
<td>Strike-through text †</td>
</tr>
<tr>
<td><code>&lt;u&gt;</code></td>
<td>Underlined</td>
</tr>
<tr>
<td><code>&lt;big&gt;</code></td>
<td>Bigger</td>
</tr>
<tr>
<td><code>&lt;small&gt;</code></td>
<td>Smaller</td>
</tr>
<tr>
<td><code>&lt;sub&gt;</code></td>
<td>Subscript</td>
</tr>
<tr>
<td><code>&lt;sup&gt;</code></td>
<td>Superscript</td>
</tr>
<tr>
<td><code>&lt;tt&gt;</code></td>
<td>Teletype</td>
</tr>
<tr>
<td><code>&lt;em&gt;</code></td>
<td>Emphasized</td>
</tr>
<tr>
<td><code>&lt;strong&gt;</code></td>
<td>Strong</td>
</tr>
<tr>
<td><code>&lt;del&gt;</code></td>
<td>Deleted</td>
</tr>
<tr>
<td><code>&lt;ins&gt;</code></td>
<td>Inserted</td>
</tr>
<tr>
<td><code>&lt;pre&gt;</code></td>
<td>Preformatted</td>
</tr>
<tr>
<td><code>&lt;code&gt;</code></td>
<td>Code</td>
</tr>
<tr>
<td><code>&lt;samp&gt;</code></td>
<td>Sample</td>
</tr>
<tr>
<td><code>&lt;kbd&gt;</code></td>
<td>Keyboard</td>
</tr>
<tr>
<td><code>&lt;var&gt;</code></td>
<td>Variable</td>
</tr>
<tr>
<td><code>&lt;cite&gt;</code></td>
<td>Citation</td>
</tr>
<tr>
<td><code>&lt;blockquote&gt;</code></td>
<td>Quotation</td>
</tr>
<tr>
<td><code>&lt;q&gt;</code></td>
<td>Short quotation</td>
</tr>
<tr>
<td><code>&lt;address&gt;</code></td>
<td>Address</td>
</tr>
<tr>
<td><code>&lt;abbr&gt;</code></td>
<td>Abbreviation</td>
</tr>
<tr>
<td><code>&lt;acronym&gt;</code></td>
<td>Acronym</td>
</tr>
<tr>
<td><code>&lt;dfn&gt;</code></td>
<td>Definition</td>
</tr>
<tr>
<td><code>&lt;font&gt;</code></td>
<td>Font</td>
</tr>
<tr>
<td><code>&lt;bdo&gt;</code></td>
<td>Bidirectional override</td>
</tr>
<tr>
<td><code>&lt;xmp&gt;</code></td>
<td>XMP</td>
</tr>
<tr>
<td>other tags</td>
<td>Attributes common to almost all other tags</td>
</tr>
</tbody>
</table>

† = deprecated
### HTML 4.01 images

<table>
<thead>
<tr>
<th>Tag</th>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;img&gt;</code></td>
<td>align=top</td>
<td>bottom</td>
</tr>
<tr>
<td></td>
<td>alt=alternatetext</td>
<td>Description of the image for text-only browsers</td>
</tr>
<tr>
<td></td>
<td>border=npixels</td>
<td>Border width around the image †</td>
</tr>
<tr>
<td></td>
<td>height=npixels</td>
<td>percent</td>
</tr>
<tr>
<td></td>
<td>hspace=mpixels</td>
<td>Blank space on the left and right side of image †</td>
</tr>
<tr>
<td></td>
<td>ismap=url</td>
<td>URL for server-side image map</td>
</tr>
<tr>
<td></td>
<td>longdesc=url</td>
<td>URL containing a long description of the image</td>
</tr>
<tr>
<td></td>
<td>src=url</td>
<td>URL of the image</td>
</tr>
<tr>
<td></td>
<td>usemap=url</td>
<td>URL for client-side image map</td>
</tr>
<tr>
<td></td>
<td>vspace=mpixels</td>
<td>Blank space on top and bottom of image †</td>
</tr>
<tr>
<td></td>
<td>width=mpixels</td>
<td>percent</td>
</tr>
<tr>
<td><code>&lt;map&gt;</code></td>
<td>id=id</td>
<td>Unique ID for the map tag</td>
</tr>
<tr>
<td></td>
<td>name=name</td>
<td>Unique name for the map tag</td>
</tr>
<tr>
<td><code>&lt;area&gt;</code></td>
<td>alt=alternatetext</td>
<td>Description of area for text-only browsers</td>
</tr>
<tr>
<td></td>
<td>coords=left,top,right,bottom</td>
<td>cx,cy,radius</td>
</tr>
<tr>
<td></td>
<td>href=url</td>
<td>Target URL of area</td>
</tr>
<tr>
<td></td>
<td>nohref=true</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td>shape=rectangle</td>
<td>circle</td>
</tr>
<tr>
<td></td>
<td>target=_blank</td>
<td>_parent</td>
</tr>
</tbody>
</table>

† = deprecated
### HTML 4.01 tables

<table>
<thead>
<tr>
<th>Tag</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>&lt;table&gt;</em></td>
<td><strong>Table</strong> Table alignment † Table background color † Border width</td>
</tr>
<tr>
<td></td>
<td>Space around the content of each cell Space between cells Visibility of sides of the table border Horizontal or vertical divider lines Summary of the table for text-only browsers Table width</td>
</tr>
<tr>
<td><em>&lt;tr&gt;</em></td>
<td><strong>Table row</strong> Horizontal text alignment Row background color † Character to align text on, if align=char Alignment offset to first character, if align=char Vertical text alignment</td>
</tr>
<tr>
<td><em>&lt;td&gt;</em></td>
<td><strong>Table cell</strong> Abbreviated content in a cell Horizontal text alignment Cell name Cell background color † Character to align text on, if align=char Alignment offset to first character, if align=char Number of columns this cell spans on Cell header information for text-only browsers Cell height † Text in cell stays on a single line † Number of rows this cell spans on Target for cell header information Vertical text alignment Cell width †</td>
</tr>
<tr>
<td><em>&lt;th&gt;</em></td>
<td><strong>Table header</strong></td>
</tr>
</tbody>
</table>

† = deprecated
### 7-bit ASCII table

<table>
<thead>
<tr>
<th>Dec</th>
<th>Hex</th>
<th>Char</th>
<th>Dec</th>
<th>Hex</th>
<th>Char</th>
<th>Dec</th>
<th>Hex</th>
<th>Char</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>NUL</td>
<td>32</td>
<td>20</td>
<td>space</td>
<td>64</td>
<td>40</td>
<td>@</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>SOH</td>
<td>33</td>
<td>21</td>
<td>!</td>
<td>65</td>
<td>41</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>STX</td>
<td>34</td>
<td>22</td>
<td>&quot;</td>
<td>66</td>
<td>42</td>
<td>B</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>ETX</td>
<td>35</td>
<td>23</td>
<td>#</td>
<td>67</td>
<td>43</td>
<td>C</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>EOT</td>
<td>36</td>
<td>24</td>
<td>$</td>
<td>68</td>
<td>44</td>
<td>D</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>ENQ</td>
<td>37</td>
<td>25</td>
<td>%</td>
<td>69</td>
<td>45</td>
<td>E</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>ACK</td>
<td>38</td>
<td>26</td>
<td>&amp;</td>
<td>70</td>
<td>46</td>
<td>F</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>BEL</td>
<td>39</td>
<td>27</td>
<td>'</td>
<td>71</td>
<td>47</td>
<td>G</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>BS</td>
<td>40</td>
<td>28</td>
<td>(</td>
<td>72</td>
<td>48</td>
<td>H</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>TAB</td>
<td>41</td>
<td>29</td>
<td>)</td>
<td>73</td>
<td>49</td>
<td>I</td>
</tr>
<tr>
<td>10</td>
<td>A</td>
<td>LF</td>
<td>42</td>
<td>2A</td>
<td>*</td>
<td>74</td>
<td>4A</td>
<td>J</td>
</tr>
<tr>
<td>11</td>
<td>B</td>
<td>VT</td>
<td>43</td>
<td>2B</td>
<td>+</td>
<td>75</td>
<td>4B</td>
<td>K</td>
</tr>
<tr>
<td>12</td>
<td>C</td>
<td>FF</td>
<td>44</td>
<td>2C</td>
<td>,</td>
<td>76</td>
<td>4C</td>
<td>L</td>
</tr>
<tr>
<td>13</td>
<td>D</td>
<td>CR</td>
<td>45</td>
<td>2D</td>
<td>-</td>
<td>77</td>
<td>4D</td>
<td>M</td>
</tr>
<tr>
<td>14</td>
<td>E</td>
<td>SO</td>
<td>46</td>
<td>2E</td>
<td>.</td>
<td>78</td>
<td>4E</td>
<td>N</td>
</tr>
<tr>
<td>15</td>
<td>F</td>
<td>SI</td>
<td>47</td>
<td>2F</td>
<td>/</td>
<td>79</td>
<td>4F</td>
<td>O</td>
</tr>
<tr>
<td>16</td>
<td>G</td>
<td>DLE</td>
<td>48</td>
<td>30</td>
<td>0</td>
<td>80</td>
<td>50</td>
<td>P</td>
</tr>
<tr>
<td>17</td>
<td>H</td>
<td>DC1</td>
<td>49</td>
<td>31</td>
<td>1</td>
<td>81</td>
<td>51</td>
<td>Q</td>
</tr>
<tr>
<td>18</td>
<td>I</td>
<td>DC2</td>
<td>50</td>
<td>32</td>
<td>2</td>
<td>82</td>
<td>52</td>
<td>R</td>
</tr>
<tr>
<td>19</td>
<td>J</td>
<td>DC3</td>
<td>51</td>
<td>33</td>
<td>3</td>
<td>83</td>
<td>53</td>
<td>S</td>
</tr>
<tr>
<td>20</td>
<td>K</td>
<td>DC4</td>
<td>52</td>
<td>34</td>
<td>4</td>
<td>84</td>
<td>54</td>
<td>T</td>
</tr>
<tr>
<td>21</td>
<td>L</td>
<td>NAK</td>
<td>53</td>
<td>35</td>
<td>5</td>
<td>85</td>
<td>55</td>
<td>U</td>
</tr>
<tr>
<td>22</td>
<td>M</td>
<td>SYN</td>
<td>54</td>
<td>36</td>
<td>6</td>
<td>86</td>
<td>56</td>
<td>V</td>
</tr>
<tr>
<td>23</td>
<td>N</td>
<td>ETB</td>
<td>55</td>
<td>37</td>
<td>7</td>
<td>87</td>
<td>57</td>
<td>W</td>
</tr>
<tr>
<td>24</td>
<td>O</td>
<td>CAN</td>
<td>56</td>
<td>38</td>
<td>8</td>
<td>88</td>
<td>58</td>
<td>X</td>
</tr>
<tr>
<td>25</td>
<td>P</td>
<td>EM</td>
<td>57</td>
<td>39</td>
<td>9</td>
<td>89</td>
<td>59</td>
<td>Y</td>
</tr>
<tr>
<td>26</td>
<td>Q</td>
<td>SUB</td>
<td>58</td>
<td>3A</td>
<td>:</td>
<td>90</td>
<td>5A</td>
<td>Z</td>
</tr>
<tr>
<td>27</td>
<td>R</td>
<td>ESC</td>
<td>59</td>
<td>3B</td>
<td>;</td>
<td>91</td>
<td>5B</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>S</td>
<td>FS</td>
<td>60</td>
<td>3C</td>
<td>&lt;</td>
<td>92</td>
<td>5C</td>
<td>\</td>
</tr>
<tr>
<td>29</td>
<td>T</td>
<td>GS</td>
<td>61</td>
<td>3D</td>
<td>=</td>
<td>93</td>
<td>5D</td>
<td>{</td>
</tr>
<tr>
<td>30</td>
<td>U</td>
<td>RS</td>
<td>62</td>
<td>3E</td>
<td>&gt;</td>
<td>94</td>
<td>5E</td>
<td>^</td>
</tr>
<tr>
<td>31</td>
<td>V</td>
<td>US</td>
<td>63</td>
<td>3F</td>
<td>?</td>
<td>95</td>
<td>5F</td>
<td>_</td>
</tr>
</tbody>
</table>

Characters 0-31 and 127 are non-printable.