(Ubuntu) Linux Cheat Sheet

Keyboard Shortcuts

Desktop

Switch workspace	`[Ctrl] + [Alt] + [←|→]`
Move window to workspace	`[Shift] + [Ctrl] + [Alt] + [←|→]`
Switch window	`[Shift] + [Alt] + [Tab]`
Show desktop	`[Ctrl] + [Alt] + [D]`
Open terminal	`[Ctrl] + [Alt] + [T]`
Lock screen	`[Ctrl] + [Alt] + [L]`
Show hidden files	`[Ctrl] + [H]`
Show file properties	`[Alt] + [Enter]`
Restart session	`[Ctrl] + [Alt] + [Backspace]`
Open launcher	`[Super]`
Configure Unity	`[Alt] + [F2]`

Nano

Save file	`[Ctrl] + [O]`
Load file (at cursor position)	`[Ctrl] + [R]`
Exit Nano	`[Ctrl] + [X]`
Cut line at cursor position	`[Ctrl] + [K]`
Paste line at cursor position	`[Ctrl] + [U]`
Find word or phrase	`[Ctrl] + [W]`
Check spelling	`[Ctrl] + [T]`

File Management

Filestructure

`/`	Root element
`/dev`	Interfaces for hardware control
`/mnt`	Mountpoint for primary partitions
`/bin`	Binaries: commands, executable by anyone
`/sbin`	System binaries: commands, executable by root
`/lib`	Required libraries
`/usr`	* Unix system resources: Programm-files
`/opt`	Non-standard-software
`/etc`	* Configuration files
`/var`	* Variable Data: Databases, repositiories
`/sys`	Plug-and-play user interface
`/root`	Home directory of user root
`/home`	* Directory for registered users
`/media`	Mountpoint for external devices
`/tmp`	* Temporary files

* Marked directories should be placed on seperate partitions.

Finding files

`updatedb`	Update file location database
`locate` *file*	Find all locations of *file*
`find` *directory* `-name "*`*file*`*"`	Find location of **file** in *directory*
`whereis` *command*	Find command related paths

Working with the filestructure

`ls`	List files
`ls -a`	List files including hidden files
`ls -l`	List files in long format
`cd`	Change to home directory
`cd` *directory*	Change to directory *directory*
`mkdir` *directory*	Make a Directory with name *directory*
`mv` *source target*	Move *source* to location *target*
`cp` *source target*	Copy *source* to location *target*
`rm` *file*	Remove file with name *file*
`rm -r` *directory*	Remove directory with name *directory*
`rm -f` *file*	Force remove of file *file*
`ln -s` *target link*	Create a symbolic link named *link* to *target*
`touch` *file*	Change timestamps of file *file* to now

`ls -l` shows the filetype and the permissions (see File permissions).
Filetypes are: `d` = directory, `.` = file, `l` = link.

Inspecting files

`more` *file*	Show contents of file *file* pagewise
`less` *file*	Show contents of file *file* without loading all of it
`head -`*n file*	Show top *n* lines of file *file*
`tail -`*n file*	Show last *n* lines of file *file*
`tail -f` *file*	Show last 20 lines of file *file* appending new lines
`grep` *pattern file*	Show all lines containing *pattern* in file *file*
`grep -r` *pattern directory*	Show all lines containing *pattern* in directory *directory*

File permissions

`chmod` *octal file*	Changes permission of file *file* to *octal*
`chown` *owner file*	Change ownership of file *file* to *owner*
`chgrp` *group file*	Change groupmembership of file *file* to *group*

File permission octal

order: owner/group/world

4	read	r
2	write	w
1	execute	x

e.g.:
`chmod 777` - rwx for everyone
`chmod 755` - rwx for owner, rx for everyone else

Compression

Works on files as well as on directories.

`tar -cwf` *archive.tar directory*	Compress *directory* to *archive.tar*
`tar -xwf` *archive.tar*	Extract archive *archive.tar* into current directory
`tar -tf` *archive.tar*	Show contents of *archive.tar*
`gzip` *file*	Compress *file* and rename it to *file*`.gz`
`gzip -d` *file.gz*	Extract file *file.gz*

All tar flags

`c`	Compress	`j`	bzip compression
`x`	Extract	`v`	Verbose output
`t`	Table of contents	`w`	Ask for confirmation
`f`	Filename	`k`	Don't overwrite
`z`	Also use gzip	`T`	Files from file

Network Management

`ping` *host*	Ping host *host*
`whois` *domain*	Get who is information about *domain*
`dig` *domain*	Get DNS for *domain*
`dig -x` *host*	Reverse lookup *host*
`ifconfig`	Show network address(es)
`netstat`	Show usage of ports and protocols
`nmap`	Scan ports (see `man nmap`)
`/etc/hosts/`	Stores local dns lookup table

Firewall

`ufw enable`	Enable Linux firewall (now and forever)
`ufw disable`	Disable Linux firewall (now and forever)
`ufw status`	Show status of Linux firewall

iptables

`iptables` contains the rules for the unix firewall.
It is advisable to create a script setting up the rules, which should be loaded at startup. Create two executable scripts:
`/etc/network/if-pre-up.d/iptablesload` and
`/etc/network/if-post-down.d/iptablessave` (see scripts)
Flags:

`-A`	Append rule
`-i`	Apply rule to interace: `[eth0|lo]`
`-p`	Connection protocol: `[tcp|udp|udplite|icmp|esp|ah|sctp|all]`
`-d`	Destination address[/mask] e.g. `192.168.1.0/16`
`-s`	Source address[/mask] e.g. `192.168.1.0/16`
`--dport`	Destination ports: *start*`[:`*end*`]`
`--sport`	Source ports: *start*`[:`*end*`]`
`-m conntrack --cstate`	Base rule on connection state: `[NEW|RELATED|ESTABLISHED|INVALID]`
`-j`	What to do: `[ACCEPT|REJECT|DROP|LOG]`

wget

General usage to download a file: `wget` *url*

`-i` *file*	Get urls from file
`-nv`	Non verbose output
`-q`	Don't print anything on console
`-t`*n*	Try maximal *n* times
`-N`	Download only if newer than stored file
`-O` *file*	Store downloaded file as *file*
`-P` *directory*	Store files to directory *directory*
`-nd`	Don't make directories as in source
`-x`	Make same structure as in source
`-r`	Recursive download
`-l`*n*	Go *n* levels deep
`-A` *list*	Comma seperated list of file extensions to download
`-R` *list*	Comma seperated list of file extensions to not download
`-D` *list*	Comma seperated list of domains to search
`-np`	Ignore parent directories
`-p`	Pretend download of whole page

e.g.
`wget -r -l1 -A mp3,aac` *url* - Download all direct linked mp3 or aac files from *url*

System Management

System info

`date`	Show current date/time
`cal`	Show this months calendar
`uptime`	Show uptime
`w`	Show users online
`whoami`	Show who you are logged in as
`df`	Show disk usage
`du`	Show directory space usage
`du -sh`	Show directory space usage in GB
`free`	Show Memory and swap space
`man` *command*	Show manual for command *command*
`/etc/cron.*`	Stores cronjobs executed every month, week, day or hour

Process Management

`ps`	Show currently active processes
`ps aux`	Show currently active processes in detail
`kill` *pid*	End process with pid *pid*
`killall` *process*	End all processes named *process*
`apachectl [start|stop|restart]`	Start, stop or restart Apache
`service` *service*`[start|stop]`	Start, stop service *service*

Device Management

Mounting drives

Mount device *device* with filesystem *filesystem* to directory *directory*:
`mount -t` *filesystem* `/dev/`*device directory*
Mount a CDRom: `mount -t iso9660 /dev/cdrom /media/cdrom`
Unmount device mounted at directory *directory*: `umount` *directory*
`/etc/fstab` stores commands for mounting devices at startup

Devices

Types:

`sd`	SATA, SCSI and USB devices
`hd`	IDE devices
`cdrom`	CDRom or DVDRom
`cdrw`	Writeable CDRom or DVDRom

Devices are found in `\dev`. Devices of the same type get letters to identify them. The letter is followed by a number indicating the partition. Partitions $1-4$ are primary partitions, $5-15$ are logical partitions. So devices are indicated as `/dev/`*type*`[a-z][a-z][1-15]`.
e.g.
`/dev/sdc5` indicates the first logical partition of the third SATA/SCSI or USB device.

Filesystems

Types:

`ext`	Linux: extended filesystem
`ext[2-4]`	Linux: successors of `ext`
`ntfs`	Windows: journaling filesystem
`fat16`	Windows: names must be 8.3, max. file size = 2 GiB
`fat32`	Windows: max. file size = 4 GiB
`iso9660`	Format for DVDs and CDs

User Management

`adduser` *user*	Create new user *user*
`adduser -r` *user*	Create new system user *user*
`passwd` *user*	Give user *user* a new password
`groupadd` *group*	Create new group *group*
`usermod -G -a` *group user*	Add user *user* to group *group*
`/etc/passwd`	Stores users
`/etc/shadow`	Stores encrypted passwords
`/etc/group`	Stores groups
`/etc/aliases`	Stores email nicknames
`/etc/sudoers`	Stores users with superuser privileges

Subversion

Install subversion	`sudo apt-get install subversion`
Create repository	`svnadmin create /`*path/to/repo/*
Make a project folder	`svn mkdir file:///`*path/to/repo/project*
Import project	Inside the project directory: `svn import file:///`*path/to/repo/project*
Checkout	`svn checkout file:///`*path/to/repo/project*
Commit changes	Inside the project directory: `svn commit -m "`*commit message*`"`
Get latest revision	Inside the project directory: `svn update`
Revert	Inside the project directory: `svn revert <filename>`
Revert to version	Inside the project directory: `svn update -r` *number* `<filename>`
Add files	Inside the project directory: `svn add (`*file*`|`*directory*`)`
Remove files	Inside the project directory: `svn delete (`*file*`|`*directory*`)`
List files in project	`svn list --verbose file:///`*path/to/repo/project*
Creating releases	`svn copy file:///`*path/to/repo/project*`/trunk file:///`*path/to/repo/project*`/tags/`*version* `-m "Tagging the` *version* `release"`
Listing releases	`svn list file:///`*path/to/repo/project*`/tags`
Checkout release	`svn checkout file:///`*path/to/repo/project*`/tags/`*version*

`<filename>` is an optional name of a file.
Before you can use subversion checkout a working-copy of your projects first. After importing a project you can remove it and than check it out from the projects parent directory. Sometimes subversions opens your default text editor. You should enter comments there.
Don't forget to make the repository accessible (`chmod 666`)

Miscellaneous

Setting JAVA_HOME

1. Find the location of your jvm. ex: `/usr/lib/jvm/java-6-sun/` 2. Edit your environmental variables: `nano /etc/environment` 3. Add a line: `JAVA_HOME="/usr/lib/jvm/java-6-sun"` 4. Log out 5. Log back in 6. Open terminal and run `echo $JAVA_HOME`

Create custom launcher

Create a new launcher:
`gnome-desktop-item-edit ~/.local/share/applications/ --create-new`
Drag the launcher to the unity launcher bar

Installing Google Chrome

Add the google signing key:
`wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | sudo apt-key add -`
Add google to your repositories:
`sh -c 'echo "deb http://dl.google.com/linux/chrome/deb/ stable main" >> /etc/apt/sources.list.d/google.list'`
Update your repositories: `apt-get update`
Install Goolge Chrome:
`apt-get install [google-chrome-stable|google-chrome-beta]`

Installing fonts

Place your `*.ttf` Fonts inside `~/.fonts`

Personalizing the terminal

Paste your welcome message to the end of `~/.bashrc`
e.g.
`figlet "`*big message*`"`
`echo -n "Hello "; whoami`

Grab youTube-Videos

`sudo apt-get install youtube-dl`
`youtube-dl -o` *target*`.flv "http://www.youtube.com/watch?v=`*youtubeId*`"`

Convert video formats

`ffmpeg -i` *source.format target.format*

Scripts

iptablesload

iptablesload:

```
#!/bin/sh
iptables-restore < /etc/iptables.rules
exit 0
```

iptablessave

```
#!/bin/sh
iptables-save -c > /etc/iptables.rules
if [ -f /etc/iptables.downrules ]; then
   iptables-restore < /etc/iptables.downrules
fi
exit 0
```


Revision: 1.0, Datum: 2011-11-01