Linux Command Line Cheat-Sheet

This is a list of Linux commands for common operations.
Note items marked with • are valid/safe to paste without modification into a terminal, so you may want to keep a terminal window open while reading this so you can cut & paste.

Command		Description
•	apropos word	Show commands pertinent to word. See also threadsafe
	which command	Show full path name of command
	time command	See how long a command takes
•	time cat	Start stopwatch. Ctrl-d to stop. See also sw
•	nice info	Run a low priority command (info in this case)
•	renice 19 -p $$	Make shell (script) low priority. Use for non interactive tasks
•	look prefix	Quickly search (sorted) dictionary
•	grep --color expr…ion /usr/share/dict/words	Highlight occurances of regular expression in dictionary
	gpg -c file	Encrypt file
	gpg file.gpg	Decrypt file
•	alias hd='od -Ax -tx1z -v'	Handy hexdump. (usage e.g.: • hd /proc/self/cmdline | less)
•	alias realpath='readlink -f'	Canonicalize path. (usage e.g.: • realpath ~/../$USER)
•	set | grep $USER	Search current environment
•	ls /usr/bin | pr -T9 -W$COLUMNS	Print in 9 columns to width of terminal
	touch -c -t 0304050607 file	Set file timestamp (YYMMDDhhmm)
dir navigation		
•	cd -	Go to previous directory
•	cd	Go to home directory
	(cd dir && command)	Go to dir, execute command and return to current dir
•	pushd **.**	Put current dir on stack so you can **popd** back to it
CDs		
	gzip < /dev/cdrom > cdrom.iso.gz	Save copy of data cdrom
	mkisofs -V NAME -r dir | gzip > cdrom.iso.gz	Create cdrom image from contents of dir
	mount -o loop cdrom.iso /mnt/dir	Mount the cdrom image at /mnt/dir (read only)
	cdrecord -v dev=/dev/cdrom blank=fast	Clear a CDRW
	gzip -dc cdrom.iso.gz | cdrecord -v dev=/dev/cdrom -	Burn cdrom image (use dev=ATAPI -scanbus to confirm dev)
	cdparanoia -B	Rip audio tracks from CD to wav files in current dir
	cdrecord -v dev=/dev/cdrom -audio *.wav	Make audio CD from all wavs in current dir (see also cdrdao)

	oggenc --tracknum='track' track.cdda.wav -o 'track.ogg'	Make ogg file from wav file
archives		
	tar c dir/ | bzip2 > dir.tar.bz2	Make archive of dir/
	bzip2 -dc dir.tar.bz2 | tar x	Extract archive (use gzip instead of bzip2 for tar.gz files)
	tar c dir/ | gzip | gpg -c | ssh user@remote 'dd of=dir.tar.gz.gpg'	Make encrypted archive of dir/ on remote machine
	find dir/ -name '*.txt' | tar c --files-from=- | bzip2 > dir_txt.tar.bz2	Make archive of subset of dir/ and below
	find dir/ -name '*.txt' | xargs cp -a --target-directory=dir_txt/ --parents	Make copy of subset of dir/ and below
	(tar c /dir/to/copy) | (cd /where/to/ && tar x -p)	Copy (with permissions) copy/ dir to /where/to/ dir
	(cd /dir/to/copy && tar c **.**) | (cd /where/to/ && tar x -p)	Copy (with permissions) contents of copy/ dir to /where/to/
	(tar c /dir/to/copy) | ssh -C user@remote 'cd /where/to/ && tar x -p'	Copy (with permissions) copy/ dir to remote:/where/to/ dir
	dd bs=1M if=/dev/hda | gzip | ssh user@remote 'dd of=hda.gz'	Backup harddisk to remote machine
rsync (Use the --dry-run option for testing)		
	rsync -P rsync://rsync.server.com/path/to/file file	Only get diffs. Do multiple times for troublesome downloads
	rsync --bwlimit=1000 fromfile tofile	Locally copy with rate limit. It's like nice for I/O
	rsync -az -e ssh --delete ~/public_html/ remote.com:'~/public_html'	Mirror web site (using compression and encryption)
	rsync -auz -e ssh remote:/dir/ **.** && rsync -auz -e ssh **.** remote:/dir/	Synchronize current directory with remote one
file searching		
•	alias l='ls -l --color=auto'	quick dir listing
•	ls -lrt	List files by date. See also newest
	find -name '*.[ch]' | xargs grep -E 'expr'	Search 'expr' in this dir and below. See also findrepo
	find -type f -print0 | xargs -r0 grep -F 'string'	Search all regular files for 'string' in this dir and below
	find -maxdepth 1 -type f | xargs grep -F 'string'	Search all regular files for 'string' in this dir
	find -maxdepth 1 -type d | while read dir; do echo $dir; echo cmd2; done	Process each item with multiple commands (in while loop)
•	find -type f ! -perm -444	Find files not readable by all (useful for web site)
•	find -type d ! -perm -111	Find dirs not accessible by all (useful for web site)
•	locate -r 'file[^/]*\.txt'	Search cached index for names. This re is like glob *file*.txt
networking (Note ifconfig, route, mii-tool, nslookup commands are obsolete)		
•	ip link show	List interfaces
	ethtool interface	List interface status
	ip link set dev eth0 name wan	Rename eth0 to wan
	ip addr add 1.2.3.4/24 brd + dev eth0	Add ip and mask(255.255.255.0)
	ip link set dev interface up	Bring interface up (or down)
	ip route add default via 1.2.3.254	Set default gateway to 1.2.3.254

•	tc qdisc add dev lo root handle 1:0 netem delay 20msec	Add 20ms latency to loopback device (for testing)
•	tc qdisc del dev lo root	Remove latency added above
•	host pixelbeat.org	Lookup ip address for name or vice versa
•	hostname -i	Lookup local ip address (equivalent to host \`hostname\`)
•	netstat -tupl	List internet services on a system
•	netstat -tup	List active connections to/from system
wget (multi purpose download tool)		
•	(cd cmdline && wget -nd -pHEKk http://www.pixelbeat.org/cmdline.html)	Store local browsable version of a page to the current dir
	wget -c http://www.example.com/large.file	Continue downloading a partially downloaded file
	wget -r -nd -np -l1 -A '*.jpg' http://www.example.com/	Download a set of files to the current directory
	wget ftp://remote/file[1-9].iso/	FTP supports globbing directly
•	wget -q -O- http://www.pixelbeat.org/timeline.html | grep 'a href' | head	Process output directly
	echo 'wget url' | at 01:00	Download url at 1AM to current dir
	wget --limit-rate=20k url	Do a low priority download (limit to 20KB/s in this case)
	wget -nv --spider --force-html -i bookmarks.html	Check links in a file
	wget --mirror http://www.example.com/	Efficiently update a local copy of a site (handy from cron)
windows (note samba is the package that provides all this windows specific networking support)		
•	smbtree	Find windows machines. See also findsmb
	nmblookup -A 1.2.3.4	Find the windows (netbios) name associated with ip address
	smbclient -L windows_box	List shares on windows machine or samba server
	mount -t smbfs -o fmask=666,guest //windows_box/share /mnt/share	Mount a windows share
	echo 'message' | smbclient -M windows_box	Send popup to windows machine (off by default in XP sp2)
math		
•	echo '(1 + sqrt(5))/2' | bc -l	Quick math
•	echo 'obase=16; ibase=10; 123' | bc	Base conversion (decimal to hexadecimal)
•	echo $((0x2dec))	Base conversion (hex to dec) ((shell arithmetic expansion))
•	echo 'pad=20; min=64; (100*10^6)/((pad+min)*8)' | bc	More complex (int) e.g. This shows max FastE packet rate
•	echo 'pad=20; min=64; print (100E6)/((pad+min)*8)' | python	Python handles scientific notation
•	echo 'pad=20; plot [64:1518] (100*10**6)/((pad+x)*8)' | gnuplot -persist	Plot FastE packet rate vs packet size
text manipulation (note sed uses stdin and stdout, so if you want to edit files, append <*oldfile* >*newfile*)		
	sed 's/string1/string2/g'	Replace string1 with string2

<table>
<tr><td></td><td>sed 's/\(.*\)1/\12/g'</td><td>Modify anystring1 to anystring2</td></tr>
<tr><td></td><td>sed '/ *#/d; /^ *$/d'</td><td>Remove comments and blank lines</td></tr>
<tr><td></td><td>sed ':a; /\\$/N; s/\\\n//; ta'</td><td>Concatenate lines with trailing \</td></tr>
<tr><td></td><td>sed 's/[\t]*$//'</td><td>Remove trailing spaces from lines</td></tr>
<tr><td></td><td>sed 's/\([\\`\\"$\\\\]\)/\\\1/g'</td><td>Escape shell metacharacters active within double quotes</td></tr>
<tr><td></td><td>sed -n '1000p;1000q'</td><td>Print 1000th line</td></tr>
<tr><td></td><td>sed -n '10,20p;20q'</td><td>Print lines 10 to 20</td></tr>
<tr><td></td><td>sed -n 's/.*<[tT][iI][tT][lL][eE]>\(.*\)<\/[tT][iI][tT][lL][eE]>.*/\1/p;T;q'</td><td>Extract title from HTML web page</td></tr>
<tr><td></td><td>sort -t. -k1,1n -k2,2n -k3,3n -k4,4n</td><td>Sort IPV4 ip addresses</td></tr>
<tr><td>•</td><td>echo 'Test' | tr '[:lower:]' '[:upper:]'</td><td>Case conversion</td></tr>
<tr><td>•</td><td>tr -dc '[:print:]' < /dev/urandom</td><td>Filter non printable characters</td></tr>
<tr><td>•</td><td>grep 'processor' /proc/cpuinfo | wc -l</td><td>Count lines</td></tr>
<tr><td colspan="3"><b>set operations</b> (Note LANG=C is for speed)</td></tr>
<tr><td></td><td>LANG=C sort file1 file2 | uniq</td><td>Union of unsorted files</td></tr>
<tr><td></td><td>LANG=C sort file1 file2 | uniq -d</td><td>Intersection of unsorted files</td></tr>
<tr><td></td><td>LANG=C sort file1 file1 file2 | uniq -u</td><td>Difference of unsorted files</td></tr>
<tr><td></td><td>LANG=C sort file1 file2 | uniq -u</td><td>Symmetric Difference of unsorted files</td></tr>
<tr><td></td><td>LANG=C comm file1 file2 | sed 's/^\t*//'</td><td>Union of sorted files</td></tr>
<tr><td></td><td>LANG=C comm -12 file1 file2</td><td>Intersection of sorted files</td></tr>
<tr><td></td><td>LANG=C comm -13 file1 file2</td><td>Difference of sorted files</td></tr>
<tr><td></td><td>LANG=C comm -3 file1 file2 | sed 's/^\t*//'</td><td>Symmetric Difference of sorted files</td></tr>
<tr><td colspan="3"><b>calendar</b></td></tr>
<tr><td>•</td><td>cal -3</td><td>Display a calendar</td></tr>
<tr><td>•</td><td>cal 9 1752</td><td>Display a calendar for a particular month year</td></tr>
<tr><td>•</td><td>date -d fri</td><td>What date is it this friday. See also day</td></tr>
<tr><td>•</td><td>date --date='25 Dec' +%A</td><td>What day does xmas fall on, this year</td></tr>
<tr><td>•</td><td>date --date '1970-01-01 UTC 1234567890 seconds'</td><td>Convert number of seconds since the epoch to a date</td></tr>
<tr><td>•</td><td>TZ=':America/Los_Angeles' date</td><td>What time is it on West coast of US (use tzselect to find TZ)</td></tr>
<tr><td></td><td>echo "mail -s 'get the train' P@draigBrady.com < /dev/null" | at 17:45</td><td>Email reminder</td></tr>
<tr><td>•</td><td>echo "DISPLAY=$DISPLAY xmessage cooker" | at "NOW + 30 minutes"</td><td>Popup reminder</td></tr>
<tr><td colspan="3"><b>locales</b></td></tr>
<tr><td>•</td><td>printf "%'d\n" 1234</td><td>Print number with thousands grouping appropriate to locale</td></tr>
<tr><td>•</td><td>echo "I live in `locale territory`"</td><td>Extract info from locale database</td></tr>
<tr><td>•</td><td>locale | cut -d= -f1 | xargs locale -kc | less</td><td>List fields in locale database</td></tr>
<tr><td>•</td><td>LANG=en_IE.utf8 locale int_prefix</td><td>Lookup locale info for specific country. See also ccodes</td></tr>
<tr><td colspan="3"><b>disk space</b> (See also FSlint)</td></tr>
<tr><td>•</td><td>ls -lSr</td><td>Show files, biggest last</td></tr>
<tr><td>•</td><td>du -s * | sort -k1,1rn | head</td><td>Show top disk users in current dir. See also dutop</td></tr>
</table>

•	df -h	Show free disk space
•	df -i	Show free inodes
•	fdisk -l	Show disks partitions sizes and types (run as root)
•	rpm -q -a --qf '%10{SIZE}\t%{NAME}\n' | sort -k1,1n	List all packages by installed size (Bytes) on rpm distros
•	dpkg-query -W -f='${Installed-Size;10}\t${Package}\n' | sort -k1,1n	List all packages by installed size (KBytes) on deb distros
•	dd bs=1 seek=2TB if=/dev/null of=ext3.test	Create a large test file (taking no space). See also truncate
monitoring/debugging		
•	strace -c ls >/dev/null	Summarise/profile system calls made by command
•	strace -f -e open ls >/dev/null	List system calls made by command
•	ltrace -f -e getenv ls >/dev/null	List library calls made by command
•	lsof -p $$	List paths that process id has open
•	lsof ~	List processes that have specified path open
•	tcpdump not port 22	Show network traffic except ssh. See also tcpdump_not_me
•	ps -e -o pid,args --forest	List processes in a hierarchy
•	ps -e -o pcpu,cpu,nice,state,cputime,args --sort pcpu | sed '/^ 0.0 /d'	List processes by % cpu usage
•	ps -e -orss=,args= | sort -b -k1,1n | pr -TW$COLUMNS	List processes by mem usage. See also ps_mem.py
•	ps -C firefox-bin -L -o pid,tid,pcpu,state	List all threads for a particular process
•	ps -p 1,2	List info for particular process IDs
•	last reboot	Show system reboot history.
•	free -m	Show amount of (remaining) RAM (-m displays in MB)
•	watch -n1 'cat /proc/interrupts'	Watch changeable data continuously
System information (see also sysinfo)		
	hdparm -i /dev/hda	Show info about disk hda
	hdparm -tT /dev/hda	Do a read speed test on disk hda
	badblocks -s /dev/hda	Test for unreadable blocks on disk hda
•	mount | column -t	Show mounted filesystems on the system (and align output)
•	cat /proc/partitions	Show all partitions registered on the system
•	grep MemTotal /proc/meminfo	Show RAM total seen by the system
•	grep "model name" /proc/cpuinfo	Show CPU(s) info
•	lspci -tv	Show PCI info
•	lsusb -tv	Show USB info
recode (Obsoletes iconv, dos2unix, unix2dos)		
•	recode -l | less	Show available conversions (aliases on each line)
	recode windows-1252.. file_to_change.txt	Windows "ansi" to local charset (auto does CRLF conversion)
	recode utf-8/CRLF.. file_to_change.txt	Windows utf8 to local charset

	recode iso-8859-15..utf8 file_to_change.txt	Latin9 (western europe) to utf8
	recode ../b64 < file.txt > file.b64	Base64 encode
	recode /qp.. < file.txt > file.qp	Quoted printable decode
	recode ..HTML < file.txt > file.html	Text to HTML
•	recode -lf windows-1252 | grep euro	Lookup table of characters
•	echo -n 0x80 | recode latin-9/x1..dump	Show what a code represents in latin-9 charmap
•	echo -n 0x20AC | recode ucs-2/x2..latin-9/x	Show latin-9 encoding
•	echo -n 0x20AC | recode ucs-2/x2..utf-8/x	Show utf-8 encoding
interactive		
•	mc	Powerful filemanager that can browse rpm, tar, ftp, ssh, ...
•	screen	Virtual terminals with detach capability, ...
•	links	Web browser
•	gnuplot	Interactive/scriptable graphing
•	octave	Matlab like environment