

[Русский](#) | [Polski](#) | [中文](#) | [Castellano](#) | [日本語](#) | [Português do Brasil](#) | [Magyar](#) | [Srpski](#)

This is a linux command line reference for common operations.

Examples 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](#).

All these commands have been tested both on Fedora and Ubuntu.

Command	Description
• <code>apropos whatis</code>	Show commands pertinent to string. See also threadsafe
• <code>man -t man ps2pdf - > man.pdf</code>	make a pdf of a manual page
<code>which command</code>	Show full path name of command
<code>time command</code>	See how long a command takes
• <code>time cat</code>	Start stopwatch. Ctrl-d to stop. See also sw
• <code>nice info</code>	Run a low priority command (The "info" reader in this case)
• <code>renice 19 -p \$\$</code>	Make shell (script) low priority. Use for non interactive tasks
dir navigation	
• <code>cd -</code>	Go to previous directory
• <code>cd</code>	Go to \$HOME directory
<code>(cd dir && command)</code>	Go to dir, execute command and return to current dir
• <code>pushd .</code>	Put current dir on stack so you can popd back to it
file searching	
• <code>alias l='ls -l --color=auto'</code>	quick dir listing
• <code>ls -lrt</code>	List files by date. See also newest and find_mm_yyyy
• <code>ls /usr/bin pr -T9 -W\$COLUMNS</code>	Print in 9 columns to width of terminal
<code>find -name '*.ch' xargs grep -E 'expr'</code>	Search 'expr' in this dir and below. See also findrepo
<code>find -type f -print0 xargs -r0 grep -F 'example'</code>	Search all regular files for 'example' in this dir and below
<code>find -maxdepth 1 -type f xargs grep -F 'example'</code>	Search all regular files for 'example' in this dir
<code>find -maxdepth 1 -type d while read dir; do echo \$dir; echo cmd2; done</code>	Process each item with multiple commands (in while loop)
• <code>find -type f ! -perm -444</code>	Find files not readable by all (useful for web site)
• <code>find -type d ! -perm -111</code>	Find dirs not accessible by all (useful for web site)
• <code>locate -r 'file[^\]*\.txt'</code>	Search cached index for names. This re is like <code>glob *file*.txt</code>
• <code>look reference</code>	Quickly search (sorted) dictionary for prefix
• <code>grep --color reference /usr/share/dict/words</code>	Highlight occurrences of regular expression in dictionary
archives and compression	
<code>gpg -c file</code>	Encrypt file
<code>gpg file.gpg</code>	Decrypt file
<code>tar -c dir/ bzip2 > dir.tar.bz2</code>	Make compressed archive of dir/
<code>bzip2 -dc dir.tar.bz2 tar -x</code>	Extract archive (use <code>gzip</code> instead of <code>bzip2</code> for <code>tar.gz</code> files)
<code>tar -c dir/ gzip gpg -c ssh user@remote 'dd of=dir.tar.gz.gpg'</code>	Make encrypted archive of dir/ on remote machine
<code>find dir/ -name '*.txt' tar -c --files-from=- bzip2 > dir_txt.tar.bz2</code>	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/sda gzip ssh user@remote 'dd of=sda.gz'	Backup harddisk to remote machine
rsync (Network efficient file copier: 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
ssh (Secure SHell)	
ssh \$USER@\$HOST command	Run command on \$HOST as \$USER (default command=shell)
• ssh -f -Y \$USER@\$HOSTNAME xeyes	Run GUI command on \$HOSTNAME as \$USER
scp -p -r \$USER@\$HOST: file dir/	Copy with permissions to \$USER's home directory on \$HOST
ssh -g -L 8080:localhost:80 root@\$HOST	Forward connections to \$HOSTNAME:8080 out to \$HOST:80
ssh -R 1434:imap:143 root@\$HOST	Forward connections from \$HOST:1434 in to imap:143
wget (multi purpose download tool)	
• (cd cli && 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/dir/	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)
networking (Note ifconfig, route, mii-tool, nslookup commands are obsolete)	
ethtool eth0	Show status of ethernet interface eth0
ethtool --change eth0 autoneg off speed 100 duplex full	Manually set ethernet interface speed
iwconfig eth1	Show status of wireless interface eth1
iwconfig eth1 rate 1Mb/s fixed	Manually set wireless interface speed
• iwlist scan	List wireless networks in range
• ip link show	List network interfaces
ip link set dev eth0 name wan	Rename interface eth0 to wan
ip link set dev eth0 up	Bring interface eth0 up (or down)
• ip addr show	List addresses for interfaces
ip addr add 1.2.3.4/24 brd + dev eth0	Add (or del) ip and mask (255.255.255.0)
• ip route show	List routing table
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 DNS ip address for name or vice versa

• hostname -i	Lookup local ip address (equivalent to host `hostname`)
• whois pixelbeat.org	Lookup whois info for hostname or ip address
• netstat -tupl	List internet services on a system
• netstat -tup	List active connections to/from system
windows networking (Note samba is the package that provides all this windows specific networking support)	
• smbtree	Find windows machines. See also find smb
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)
text manipulation (Note sed uses stdin and stdout. Newer versions support inplace editing with the -i option)	
sed 's/string1/string2/g'	Replace string1 with string2
sed 's/^(.*)1/12/g'	Modify anystring1 to anystring2
sed '/ *#/d; /^ */d'	Remove comments and blank lines
sed ':a; \\\\$/N; s/\\n//; ta'	Concatenate lines with trailing \
sed 's/[\t]*\$//'	Remove trailing spaces from lines
sed 's/([\\`"\"\$\\])/\ /g'	Escape shell metacharacters active within double quotes
• seq 10 sed "s/^ / /; s/*(\\{7,\\})/1/"	Right align numbers
sed -n '1000p;1000q'	Print 1000th line
sed -n '10,20p;20q'	Print lines 10 to 20
sed -n 's/.*<title>(.*?)</title>.*\ /ip;T;q'	Extract title from HTML web page
sed -i 42d ~/.ssh/known_hosts	Delete a particular line
sort -t. -k1,1n -k2,2n -k3,3n -k4,4n	Sort IPV4 ip addresses
• echo 'Test' tr '[:lower:]' '[:upper:]'	Case conversion
• tr -dc '[:print:]' < /dev/urandom	Filter non printable characters
• history wc -l	Count lines
set operations (Note you can export LANG=C for speed. Also these assume no duplicate lines within a file)	
sort file1 file2 uniq	Union of unsorted files
sort file1 file2 uniq -d	Intersection of unsorted files
sort file1 file1 file2 uniq -u	Difference of unsorted files
sort file1 file2 uniq -u	Symmetric Difference of unsorted files
join -a1 -a2 file1 file2	Union of sorted files
join file1 file2	Intersection of sorted files
join -v2 file1 file2	Difference of sorted files
join -v1 -v2 file1 file2	Symmetric Difference of sorted files
math	
• echo '(1 + sqrt(5))/2' bc -l	Quick math (Calculate ϕ). See also bc
• 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
• echo 'obase=16; ibase=10; 64206' bc	Base conversion (decimal to hexadecimal)
• echo '\$((0x2dec))'	Base conversion (hex to dec) ((shell arithmetic expansion))
• units -t '100m/9.69s' 'miles/hour'	Unit conversion (metric to imperial)
• units -t '500GB' 'GiB'	Unit conversion (SI to IEC prefixes)
• units -t '1 googol'	Definition lookup
• seq 100 (tr '\n' +; echo 0) bc	Add a column of numbers. See also add and funcpy
calendar	
• cal -3	Display a calendar
• cal 9 1752	Display a calendar for a particular month year

• <code>date -d fri</code>	What date is it this friday. See also day
• <code>[\$(date -d "tomorrow" +%d) = "01"] exit</code>	exit a script unless it's the last day of the month
• <code>date --date='25 Dec' +%A</code>	What day does xmas fall on, this year
• <code>date --date='@2147483647'</code>	Convert seconds since the epoch (1970-01-01 UTC) to date
• <code>TZ=':America/Los_Angeles' date</code>	What time is it on West coast of US (use tzselect to find TZ)
<code>echo "mail -s 'get the train' P@draigBrady.com < /dev/null" at 17:45</code>	Email reminder
• <code>echo "DISPLAY=\$DISPLAY xmessage cooker" at "NOW + 30 minutes"</code>	Popup reminder
locales	
• <code>printf "%'d\n" 1234</code>	Print number with thousands grouping appropriate to locale
• <code>BLOCK_SIZE='1 ls -l</code>	get ls to do thousands grouping appropriate to locale
• <code>echo "I live in `locale territory`"</code>	Extract info from locale database
• <code>LANG=en_IE.utf8 locale int_prefix</code>	Lookup locale info for specific country. See also ccodes
• <code>locale cut -d= -f1 xargs locale -kc less</code>	List fields available in locale database
recode (Obsoletes iconv, dos2unix, unix2dos)	
• <code>recode -l less</code>	Show available conversions (aliases on each line)
<code>recode windows-1252.. file_to_change.txt</code>	Windows "ansi" to local charset (auto does CRLF conversion)
<code>recode utf-8/CRLF.. file_to_change.txt</code>	Windows utf8 to local charset
<code>recode iso-8859-15..utf8 file_to_change.txt</code>	Latin9 (western europe) to utf8
<code>recode ../b64 < file.txt > file.b64</code>	Base64 encode
<code>recode /qp.. < file.txt > file.qp</code>	Quoted printable decode
<code>recode ../HTML < file.txt > file.html</code>	Text to HTML
• <code>recode -lf windows-1252 grep euro</code>	Lookup table of characters
• <code>echo -n 0x80 recode latin-9/x1..dump</code>	Show what a code represents in latin-9 charmap
• <code>echo -n 0x20AC recode ucs-2/x2..latin-9/x</code>	Show latin-9 encoding
• <code>echo -n 0x20AC recode ucs-2/x2..utf-8/x</code>	Show utf-8 encoding
CDs	
<code>gzip < /dev/cdrom > cdrom.iso.gz</code>	Save copy of data cdrom
<code>mkisofs -V LABEL -r dir gzip > cdrom.iso.gz</code>	Create cdrom image from contents of dir
<code>mount -o loop cdrom.iso /mnt/dir</code>	Mount the cdrom image at /mnt/dir (read only)
<code>cdrecord -v dev=/dev/cdrom blank=fast</code>	Clear a CDRW
<code>gzip -dc cdrom.iso.gz cdrecord -v dev=/dev/cdrom -</code>	Burn cdrom image (use dev=ATAPI -scanbus to confirm dev)
<code>cdparanoia -B</code>	Rip audio tracks from CD to wav files in current dir
<code>cdrecord -v dev=/dev/cdrom -audio *.wav</code>	Make audio CD from all wavs in current dir (see also cdrdao)
<code>oggenc --tracknum='track' track.cdda.wav -o 'track.ogg'</code>	Make ogg file from wav file
disk space (See also FSLint)	
• <code>ls -lSr</code>	Show files by size, biggest last
• <code>du -s * sort -k1,1rn head</code>	Show top disk users in current dir. See also dutor
• <code>df -h</code>	Show free space on mounted filesystems
• <code>df -i</code>	Show free inodes on mounted filesystems
• <code>fdisk -l</code>	Show disks partitions sizes and types (run as root)
• <code>rpm -q -a --qf '%10{SIZE}\t%{NAME}\n' sort -k1,1n</code>	List all packages by installed size (Bytes) on rpm distros
• <code>dpkg-query -W -f='\${Installed-Size;10}\t\${Package}\n' sort -k1,1n</code>	List all packages by installed size (KBytes) on deb distros

• <code>dd bs=1 seek=2TB if=/dev/null of=ext3.test</code>	Create a large test file (taking no space). See also truncate
• <code>> file</code>	truncate data of file or create an empty file
monitoring/debugging	
• <code>tail -f /var/log/messages</code>	Monitor messages in a log file
• <code>strace -c ls >/dev/null</code>	Summarise/profile system calls made by command
• <code>strace -f -e open ls >/dev/null</code>	List system calls made by command
• <code>ltrace -f -e getenv ls >/dev/null</code>	List library calls made by command
• <code>lsop -p \$\$</code>	List paths that process id has open
• <code>lsop ~</code>	List processes that have specified path open
• <code>tcpdump not port 22</code>	Show network traffic except ssh. See also tcpdump_not_me
• <code>ps -e -o pid,args --forest</code>	List processes in a hierarchy
• <code>ps -e -o pcpu,cpu,nice,state,cputime,args --sort pcpu sed '/^ 0.0 /d'</code>	List processes by % cpu usage
• <code>ps -e -orss=,args= sort -b -k1,1n pr -TW\$COLUMNS</code>	List processes by mem usage. See also ps_mem.py
• <code>ps -C firefox-bin -L -o pid,tid,pcpu,state</code>	List all threads for a particular process
• <code>ps -p 1,2</code>	List info for particular process IDs
• <code>last reboot</code>	Show system reboot history
• <code>free -m</code>	Show amount of (remaining) RAM (-m displays in MB)
• <code>watch -n.1 'cat /proc/interrupts'</code>	Watch changeable data continuously
system information (see also sysinfo) ('#' means root access is required)	
• <code>uname -a</code>	Show kernel version and system architecture
• <code>head -n1 /etc/issue</code>	Show name and version of distribution
• <code>cat /proc/partitions</code>	Show all partitions registered on the system
• <code>grep MemTotal /proc/meminfo</code>	Show RAM total seen by the system
• <code>grep "model name" /proc/cpuinfo</code>	Show CPU(s) info
• <code>lspci -tv</code>	Show PCI info
• <code>lsusb -tv</code>	Show USB info
• <code>mount column -t</code>	List mounted filesystems on the system (and align output)
• <code>grep -F capacity: /proc/acpi/battery/BAT0/info</code>	Show state of cells in laptop battery
# <code>dmidecode -q less</code>	Display SMBIOS/DMI information
# <code>smartctl -A /dev/sda grep Power_On_Hours</code>	How long has this disk (system) been powered on in total
# <code>hdparm -i /dev/sda</code>	Show info about disk sda
# <code>hdparm -tT /dev/sda</code>	Do a read speed test on disk sda
# <code>badblocks -s /dev/sda</code>	Test for unreadable blocks on disk sda
interactive (see also linux keyboard shortcuts)	
• readline	Line editor used by bash, python, bc, gnuplot, ...
• screen	Virtual terminals with detach capability, ...
• mc	Powerful file manager that can browse rpm, tar, ftp, ssh, ...
• gnuplot	Interactive/scriptable graphing
• links	Web browser
• <code>xdg-open http://www.pixelbeat.org/</code>	open a file or url with the registered desktop application
miscellaneous	
• <code>alias hd='od -Ax -tx1z -v'</code>	Handy hexdump. (usage e.g.: • <code>hd /proc/self /cmdline less</code>)
• <code>alias realpath='readlink -f'</code>	Canonicalize path. (usage e.g.: • <code>realpath ~/./\$USER</code>)
• <code>set grep \$USER</code>	Search current environment
• <code>touch -c -t 0304050607 file</code>	Set file timestamp (YYMMDDhhmm)

<ul style="list-style-type: none">• <code>python -m SimpleHTTPServer</code>	Serve current directory tree at <code>http://\$HOSTNAME:8000/</code>
-------------------------------------------------------------------------------------------	-------------------------------------------------------------------------

© Jan 7 2008 [[Comments](#)] | [[Add link to ...](#)]

[**Ads by Google**](#)

[Perl Scripting](#)

[IP Address Finder](#)

[IP Tracker](#)

[Reverse IP Looku](#)