

Command	Description
• apropos whatis	Show commands pertinent to string. See also threadsafe
• <code>man -t man ps2pdf - > man.pdf</code>	make a pdf of a manual page
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
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
file searching	
• <code>alias l='ls -l --color=auto'</code>	quick dir listing
• ls -lrt	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)

• locate -r 'file[^\.]*\.txt'	Search cached index for names. This re is like glob *file*.txt
• look reference	Quickly search (sorted) dictionary for prefix
• grep --color reference /usr/share/dict/words	Highlight occurrences of regular expression in dictionary
archives and compression	
gpg -c file	Encrypt file
gpg file.gpg	Decrypt file
tar -c dir/ bzip2 > dir.tar.bz2	Make compressed 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/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
ssh-copy-id \$USER@\$HOST	Install \$USER's public key on \$HOST for password-less log in
wget (multi purpose download tool)	
• (cd dir/ && 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
• 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 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)
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

<code>sed '/ *#/d; /^ *\$/d'</code>	Remove comments and blank lines
<code>sed ':a; \\\\$/N; s\\\n//; ta'</code>	Concatenate lines with trailing \
<code>sed 's/[\t]*\$/'</code>	Remove trailing spaces from lines
<code>sed 's/([`"\$\])\\\1/g'</code>	Escape shell metacharacters active within double quotes
<code>seq 10 sed "s/^ / /; s/ *\(\{7,\}\)\^1/"</code>	Right align numbers
<code>sed -n '1000{p;q}'</code>	Print 1000th line
<code>sed -n '10,20p;20q'</code>	Print lines 10 to 20
<code>sed -n 's/.*<title>\(.*\)</title>.*\^1/ip;T;q'</code>	Extract title from HTML web page
<code>sed -i 42d ~/.ssh/known_hosts</code>	Delete a particular line
<code>sort -t. -k1,1n -k2,2n -k3,3n -k4,4n</code>	Sort IPV4 ip addresses
<code>echo 'Test' tr '[:lower:]' '[:upper:]'</code>	Case conversion
<code>tr -dc '[:print:]' < /dev/urandom</code>	Filter non printable characters
<code>tr -s '[:blank:]' '\t' </proc/diskstats cut -f4</code>	cut fields separated by blanks
<code>history wc -l</code>	Count lines
set operations (Note you can export LANG=C for speed. Also these assume no duplicate lines within a file)	
<code>sort file1 file2 uniq</code>	Union of unsorted files
<code>sort file1 file2 uniq -d</code>	Intersection of unsorted files
<code>sort file1 file1 file2 uniq -u</code>	Difference of unsorted files
<code>sort file1 file2 uniq -u</code>	Symmetric Difference of unsorted files
<code>join -t'\0' -a1 -a2 file1 file2</code>	Union of sorted files
<code>join -t'\0' file1 file2</code>	Intersection of sorted files
<code>join -t'\0' -v2 file1 file2</code>	Difference of sorted files
<code>join -t'\0' -v1 -v2 file1 file2</code>	Symmetric Difference of sorted files
math	
<code>echo '(1 + sqrt(5))/2' bc -l</code>	Quick math (Calculate ϕ). See also bc
<code>echo 'pad=20; min=64; (100*10^6)/((pad+min)*8)' bc</code>	More complex (int) e.g. This shows max FastE

	packet rate
<ul style="list-style-type: none"> • <code>echo 'pad=20; min=64; print (100E6)/((pad+min)*8)' python</code> 	Python handles scientific notation
<ul style="list-style-type: none"> • <code>echo 'pad=20; plot [64:1518] (100*10**6)/((pad+x)*8)' gnuplot -persist</code> 	Plot FastE packet rate vs packet size
<ul style="list-style-type: none"> • <code>echo 'obase=16; ibase=10; 64206' bc</code> 	Base conversion (decimal to hexadecimal)
<ul style="list-style-type: none"> • <code>echo \$((0x2dec))</code> 	Base conversion (hex to dec) ((shell arithmetic expansion))
<ul style="list-style-type: none"> • <code>units -t '100m/9.58s' 'miles/hour'</code> 	Unit conversion (metric to imperial)
<ul style="list-style-type: none"> • <code>units -t '500GB' 'GiB'</code> 	Unit conversion (SI to IEC prefixes)
<ul style="list-style-type: none"> • <code>units -t '1 googol'</code> 	Definition lookup
<ul style="list-style-type: none"> • <code>seq 100 (tr '\n' +; echo 0) bc</code> 	Add a column of numbers. See also add and funcpy
calendar	
<ul style="list-style-type: none"> • <code>cal -3</code> 	Display a calendar
<ul style="list-style-type: none"> • <code>cal 9 1752</code> 	Display a calendar for a particular month year
<ul style="list-style-type: none"> • <code>date -d fri</code> 	What date is it this friday. See also day
<ul style="list-style-type: none"> • <code>[\$(date -d "tomorrow" +%d) = "01"] exit</code> 	exit a script unless it's the last day of the month
<ul style="list-style-type: none"> • <code>date --date='25 Dec' +%A</code> 	What day does xmas fall on, this year
<ul style="list-style-type: none"> • <code>date --date='@2147483647'</code> 	Convert seconds since the epoch (1970-01-01 UTC) to date
<ul style="list-style-type: none"> • <code>TZ='America/Los_Angeles' date</code> 	What time is it on west coast of US (use <code>tzselect</code> to find TZ)
<ul style="list-style-type: none"> • <code>date --date='TZ="America/Los_Angeles" 09:00 next Fri'</code> 	What's the local time for 9AM next Friday on west coast US
locales	
<ul style="list-style-type: none"> • <code>printf "%'d\n" 1234</code> 	Print number with thousands grouping appropriate to locale
<ul style="list-style-type: none"> • <code>BLOCK_SIZE=\1 ls -l</code> 	Use locale thousands grouping in <code>ls</code> . See also l
<ul style="list-style-type: none"> • <code>echo "I live in `locale territory`"</code> 	Extract info from locale database
<ul style="list-style-type: none"> • <code>LANG=en_IE.utf8 locale int_prefix</code> 	Lookup locale info for specific country. See

	also ccodes
• locale cut -d= -f1 xargs locale -kc less	List fields available in locale database
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.qp > file.txt	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
CDs	
gzip < /dev/cdrom > cdrom.iso.gz	Save copy of data cdrom
mkisofs -V LABEL -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 -pad *.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
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 dutup
• <code>du -hs /home/* sort -k1,1h</code>	Sort paths by easy to interpret disk usage
• <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>lsof -p \$\$</code>	List paths that process id has open
• <code>lsof ~</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 (KB) 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

• 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 -n.1 'cat /proc/interrupts'	Watch changeable data continuously
• udevadm monitor	Monitor udev events to help configure rules
system information (see also sysinfo) ('#' means root access is required)	
• uname -a	Show kernel version and system architecture
• head -n1 /etc/issue	Show name and version of distribution
• 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
• mount column -t	List mounted filesystems on the system (and align output)
• grep -F capacity: /proc/acpi/battery/BAT0/info	Show state of cells in laptop battery
# dmidecode -q less	Display SMBIOS/DMI information
# smartctl -A /dev/sda grep Power_On_Hours	How long has this disk (system) been powered on in total
# hdparm -i /dev/sda	Show info about disk sda
# hdparm -tT /dev/sda	Do a read speed test on disk sda
# badblocks -s /dev/sda	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
• xdg-open .	open a file or url with the registered desktop application