UNIX/LINUX Top100 Commands

Version 1.26 — 19.12.2024

© 2003–2024 T. Birnthaler, OSTC GmbH

19. Dezember 2024

Inhaltsverzeichnis

1   The most important UNIX/LINUX commands

`man CMD`	Display description for CMD [**manual**]
`man -f/whatis CMD`	Display 1 line of description for CMD (man *index!*) [**file**]
`man -k/apropos TXT`	Display 1 line of description for topic TXT (man *index!*) [**keyword**]
`info CMD`	Display description for CMD (*GNU programs!*)
`help [CMD]`	List all shell commands/list description for shell CMD
`CMD --help/-h/-?/-.`	Display usage message for CMD (*GNU cmds/shell builtins/other programs!*)
`type [-a] CMD`	Display CMD type/path (*builtin/alias/function/binary!*) [**all**]
`who`	Display all users logged in
`whoami` `who am i`	Display currently (in terminal) logged in user
`id [-a] [USER]`	Display own/USER name + group memberships + IDs [**identity/all**]
`su [-[l]] USER`	Switch to account USER (*without/-=-l=incl. environment!*) [**switch user/login**]
`su` `su -[l]`	Become superuser root (*without/-=-l=incl. environment!*)
`exit logout CTRL-D`	Log out from terminal/switch back to former account/end of file [**EOF**]
`sudo CMD`	Execute one(!) command as root [**superuser do**]
`passwd [USER]`	Change (own/USER) password (*1x old, 2x new one!*) [**password**]
`more/less FILE...`	Display FILE... page by page
`CMD | more/less`	Display CMD output page by page
`lp/lpr FILE...`	Print out FILE... (lpr=BSD, lp=SysV) [**line print**]
`CMD | lp/lpr`	Print out CMD output
`cd PATH` `cd ~` `cd`	Switch to directory PATH/own home directory [**change directory**]
`cd /` `cd ..` `cd -`	Switch to root/parent/previous directory (*back and forth!*)
`pwd`	Display current directory [**print working directory**]
`ls [DIR|FILE]...`	Display content of current dir./DIR/FILE (*sorted alphab. ascending!*)
`ls -l ...`	File names + attributes ... [**long**]
`ls -a ...`	All file names (*hidden ones like .*XXX *too!*) [**all**]
`ls -d ...`	Only directory name, not directory content [**directory**]
`ls -h ...`	File size in unit K/M/G/T/P/E/Z/Y/R/! (*instead of byte!*) [**human readable**]
`ls -i ...`	Index-node number [**inode**]
`ls -F ...`	File type char after name (**=exe,* /=dir, @=link!*) [**Flag**]
`ls -R ...`	Whole directory tree [**Recursive**]
`ls -r ...`	Sorted descending (*instead of ascending!*) [**reverse**]
`ls -S ...`	Sorted by file size (*ascending!*) [**Size**]
`ls -t ...`	Sorted by modification time (*ascending!*) [**time**]
`ls --color=auto`	File type by color (*black=file, blue=dir, turq.=link, green=exec, yellow=device!*)
`tree [DIR]`	Display directory tree graphical from current/DIR directory
`touch FILE...`	Update change date of FILE... or create new file [**touch**]
`*` `*.sh` `?` `[abc]`	File name pattern (*all, extension* sh, *any char, 1 char from* abc*!*) [**globbing**]
`mv FILE NAME`	Rename FILE to NAME [**move**]
`mv FILE... DIR`	Move FILE... to directory DIR [**move**]
`mv FILE DIR/NAME`	Move FILE to directory DIR and rename it to NAME [**move**]
`cp FILE NAME`	Copy FILE to NAME (*overwrite!*) [**copy**]
`cp FILE... DIR`	Copy FILE... to directory DIR [**copy**]
`ln FILE NAME`	Create hard/physical link NAME to FILE [**link**]
`ln -s FILE NAME`	Create symbolic/logical link NAME to FILE [**symbolic**]
`rm FILE...`	Delete FILE [**remove**]
`rm -r DIR...`	Delete complete directory tree starting at DIR... [**recursive**]
`mkdir [-p] DIR...`	Create directory (path) DIR... [**make directory/path**]
`rmdir DIR...`	Delete directory DIR... (*must be empty!*) [**remove directory**]
`cat FILE1 FILE2...`	Concatenate files and output them to stdout [**concatenate**]
`CTRL-D`	End of input [**EOF/End Of File**] or logout (*instead of* exit/logout*!*)
`chmod MODE FILE/DIR`	Change file perm. [ugoa][+-=][rwxst] (*only owner!*) [**change mode**]
`umask NNN`	Set standard permission mask 000-777 for new files/dirs
	(*permissions to be removed,* r=4, w=*2,* x=*1!*) [**usage mask**]
`umask`	Display std. perm. mask: 022 (*SuSE, Redhat!*) or 002 (*Debian, Ubuntu!*)
`chown USER FILE/DIR`	Change owner to USER (*only* root*!*) [**change owner**]
`chgrp GROUP FILE/DIR`	Change group owner to GROUP (*only owner!*) [**change group**]

2 Other important UNIX/LINUX commands

`ps    pstree    top`	Display own processes/process tree/most active processes [**process status**]
`ps auxw   ps -ef`	Display all processes (BSD/SysV) [**all/user/extended/wide/every/full**]
`kill PID/%JOB...`	Stop (background) process PID/%JOB.../ (standard signal `SIGTERM`=15)
`kill -9 PID/%JOB`	Stop (background) process PID/%JOB... in any case (`SIGKILL`=9)
`kill -l`	List all signal names + numbers [**list**]
`CTRL-C`	Cancel `CMD` entry/currently running `CMD` (`SIGINT`=2) [**cancel**]
`CMD &`	Run `CMD` in background (*with number* %JOB*!*)
`nohup CMD &`	Same but don't stop at logout (`SIGHUP`=0) [**no hangup**]
`CTRL-Z   bg`	Stop running `CMD` + switch to background [**background**]
`bg/fg %JOB`	Switch %JOB to background/foreground
`jobs`	Display all background processes of shell
`CMD < FILE`	Read input for `CMD` from `FILE`
`CMD > FILE >> FILE`	Write/append output of `CMD` to `FILE`
`CMD 2> FILE 2>> FILE`	Write/append error messages of `CMD` to `FILE`
`CMD 2> /dev/null`	Ignore error messages of `CMD`
`CMD > FILE 2>&1`	Write output + error messages of `CMD` to `FILE`
`CMD1	CMD2`	Connect output of `CMD1` to input of `CMD2` [**pipe**]
`VAR="TXT"`	Set shell variable `VAR` to value `TXT` (*no space around* = *allowed!*)
`export VAR`	Convert shell variable `VAR` to environment variable
`echo $VAR ${VAR}`	Display content of shell/environment variable `VAR` twice
`set   env   printenv`	Display all shell/environment variables
`HOME   PWD`	Home directory for `cd`/current directory [**print working directory**]
`HOST   HOSTNAME`	Name of machine
`LANG   LC_ALL   LC_*`	Language settings (`C`, `en_US.UTF8`, `de_DE.UTF8`) [**language/locale**]
`PAGER   VIEWER`	Name of standard viewer (e.g. `more`/`less`; autom. called by `man`)
`PATH`	Search path for executables (*dir list separated by* `:` *!*)
`PS1  PS2  PS3  PS4`	Shell prompt: normal/continuation/select/debug [**prompt string**]
`TERM`	Terminal type (*editors,* `more`/`less`*,* `curses` *library!*) [**terminal**]
`USER   LOGNAME`	Name of logged in user [**login name**]
`EDITOR   VISUAL`	Name of standard editor (*e.g. autom. called by* `crontab`/`visudo`*!*)
`PRINTER   LPDEST`	Name of standard printer (*e.g. autom. called by* `lp`/`lpr`*!*)
`find [DIR] ...`	Search/list/process files/dirs by name + attr. starting from curr. dir/DIR
`grep "TXT" FILE...`	Display lines of `FILE` containing `TXT` [**global regex print**]
`CMD	grep "TXT"`	Display lines of `CMD` output containing `TXT`
`head [-N] FILE...`	Display first 10/N lines of `FILE`
`CMD	head [-N]`	Display first 10/N lines of `CMD` output
`tail [-N] FILE...`	Display last 10/N lines of `FILE`
`CMD	tail [-N]`	Display last 10/N lines of `CMD` output
`tail -f FILE...`	Show added lines at end of `FILE` continuously [**follow**]
`tail +N FILE...`	Show lines from line N of `FILE`...
`wc -l/-w/-c FILE...`	Display number of lines/words/characters in `FILE` [**word count/line/word/char**]
`CMD	wc -l/-w/-c`	Display number of lines/words/characters of `CMD` output
`echo [-n] "TXT"`	Display `TXT` on terminal (`-n`=*without linefeed!*) [**no newline**]
`clear   CTRL-L`	Erase terminal content (blank screen) [**formfeed/leeren**]
`date [+FORMAT]`	Display date/time (*according to* `FORMAT`*, e.g.* `%T %D %H:%M:%S %d.%m.%Y`*!*)
`more/less FILE...`	Page through `FILE` by following cmds (`man` *autom. uses* `more`/`less`*!*):
`SPACE   RETURN`	Skip 1 page/line further [**space key/return key**]
`b`	Skip 1 page back (*only* `less`*!*) [**backward**]
`1G   G   NG`	Go to top/bottom/line N of file (*nur* `less`*!*) [**Go**]
`/TXT     ?TXT`	Search `TXT` forward/backward (*only* `less`*!*)
`n        N`	Repeat last search forward/backward (*only* `less`*!*) [**next**]
`:n      :p`	Jump to next/previous (*only* `less`*!*) file [**next/previous**]
`q        h`	Leave `more`/`less` [**quit**]; Display help [**help**]
`history [N]`	List total cmd history/last N cmds
`CTRL-R TXT...`	Search cmd history for last one containing `TXT` [**reverse search**]
`CTRL-R`	Search next cmd in history containing `TXT`
`Cursor-Up/Down`	Goto previous/next cmd in history
`!NR   !TXT   !!`	Execute cmd number NR/beginning with `TXT`/last from history