

Happy UNIX developer life

Quick Reference Card

@laughing_bit — <http://git.0xdeadc0de.fr/>

Bash Basics

!! Refer to the previous command
!* Refer to all the arguments of the last command
!^ Refer to the first word of the last command
!\$ Refer to the last word of the last command
cmd1 |& cmd2 Pipe stdout and stderr of *cmd1* to *cmd2*
echo \$RANDOM Display a random number
echo \$((0x10 + 1)) Compute math formulas
source ~/.bashrc Reload the bash initialization file
pushd / popd Stack working directories
cp myfile{,.bak} Skip some writing
ls -lhd !(*avi|*mkv) List all files which do not match \command Escape any aliases for *command*

Terminal shortcuts

Ctrl-l Clear the screen
Ctrl-a Go to the beginning of line
Ctrl-e Go to the end of line
Ctrl-k Delete from cursor to the end of line
Ctrl-u Delete from cursor to the start of line
Ctrl-w Delete from cursor to the start of word
Ctrl-y Paste text cut using one of the above shortcuts
Esc c / Alt-c Capitalize from cursor to end of word
Esc u / Alt-u Upcase from cursor to end of word
Esc l / Alt-l Downcase from cursor to end of word
Alt-t Swap the current word with the previous one
Ctrl-t Swap the current character with the previous one
Esc Ctrl-e Expand the line content
Ctrl-r Enter the reverse-search history

Useful tools

colorgcc Colorize GCC compilation output
hexedit View and edit binary files, in hex or ASCII
vbindiff Visually compare binary files

kcachegrind Visualize valgrind profiling output
kompare View file differences using a nice GUI
kmag Magnify a part of the screen
screenruler Measure objects on the screen
cloc / sloccount Count physical source lines of code

Shell tricks

cat /proc/self/environ | tr "\0" "\n" Same as env
sed "s#/#/#" Same as sed "s/\\/\\/\\/"/
mount | column -t Show filesystems in a nice layout
getconf LONG_BIT Tell if the system is 32 bits or 64 bits
sed -n 5p file Print the 5th line of *file*
tar tf file | xargs rm -rf Remove all extracted files

GDB - ~/ .gdbinit

set confirm off Disable confirmation requests
set disassembly-flavor intel Set the disassembly style
set print pretty on Print structures in a nice way
set history save Record command history in a file
set print array on Print one item per line

GDB commands

Commands can be run at startup using `gdb -ex cmd`.

target remote :1234 Connect to a local server
b where [if cond] Set a breakpoint (triggered if *cond*)
commands id .. Run commands when the breakpoint *id* is hit
rb regexp Break on functions matching *regexp*
bt [full] [N/-N] Show *N* frames of the call stack
t a a cmd Apply *cmd* to all threads (e.g. *t a a bt 3*)
info f Print verbose info about the current frame
finish Continue till end of function
pt var Print info about the type of *var*
p *array@count Print the *count* first items of *array*
info locals View all local variables
info args View all function arguments

Valgrind options

--trace-children=yes Track wrapped binaries
--vgdb=full Activate the "gdbserver" functionality
--vgdb-error=0 Break the program at startup
--fullpath-after=directory .. Extend path in stack traces
--suppressions=file Specify one list of errors to hide

--gen-suppressions=all Print error definitions
--read-var-info=yes Read variable info from DWARF3

Valgrind --tool=memcheck

--leak-check=full Show each individual leak in detail
--show-reachable=yes Track all kinds of memory leak
--track-origins=yes Track the origin of uninitialised values

Valgrind --tool=callgrind

--callgrind-out-file=dest Choose *dest* as output
--dump-instr=yes Collect at per-instruction granularity
--collect-jumps=yes Collect info for conditional jumps

Valgrind --tool=massif

--massif-out-file=dest Choose *dest* as output
--depth=100 Increase depth of recorded allocation trees
--threshold=0.05 Set threshold as % of total memory
--alloc-fn=g_malloc Handle GLib allocations #0
--alloc-fn=g_malloc0 Handle GLib allocations #1
--alloc-fn=g_realloc Handle GLib allocations #2
--alloc-fn=g_try_malloc Handle GLib allocations #3
--alloc-fn=g_mem_chunk_alloc Handle GLib allocations #4

Use `ms_print --threshold=X` to process stored data.

Alias advices

man ascii Display the ASCII table
gcc -dM -E - < /dev/null Dump preprocessor defines
make -j\$(nproc) Run make using all found CPUs
find -name "*~" -delete Remove backup files
find -type d -empty -delete Remove empty directories
xclip -selection clipboard Copy from stdin
xclip -selection clipboard -o Paste to stdout
kill -STOP \$(pidof firefox) Save some CPU usage
kill -CONT \$(pidof firefox) Go to surf on the Web
grep --color=always Enjoy colors using less -R
grep --include="*.[ch]" Search in code source only
readelf -aW file Display wide info about *file*