

Obtaining information on commands

man command manual page of command
info command info page of command (if it exists)
which command complete path to command

Path syntax and commands

Path syntax:

. the current directory
.. the parent directory
~ your home directory
/ root of the system
/home/alice/prog the absolute path to the prog directory in Alice's home directory
../..prog a relative path to a prog directory situated two levels upwards wrt the current directory

Commands:

pwd print the current directory
cd PATH change to directory PATH

File manipulation commands

cp FILE DEST copy FILE in DEST
cp -R DIR DEST copy DIR in DEST
ls DIR list files in DIR
ls -l DIR list files in DIR with long listing format
ls -a DIR list all files in DIR (including hidden)
mkdir DIR create DIR directory
mkdir -p DIR create DIR directory with parents if needed
mv SRC DEST move SRC in DEST (or rename SRC if DEST does not exist)

File rights modification

chmod who+/-what file

adds/removes access right what for user who on file where who can be:

u user
g group of the user
o others
a all

and what can be:

r read
w write
x execute

Environment variables

Environment variables can be defined for the *current shell* and its sons:

export MY_LECTURES=~lectures

To get the value of a variable, use the \$ operator:

export PATH=\$PATH:~/bin

Already defined variables:

PATH the directories to be searched for executables
HOME your home directory
LANG the language (english, french, ...) used in shell
SHELL the shell you are using (bash, zsh, ...)

The .bashrc file

Every command in the .bashrc file situated in your home directory will be executed at the start of a shell (particularly **export** commands).

Process management

top view interactively the processes
ps list running processes for the user
ps -ef more verbose
nice -n adj command adjust process niceness by adding adj (-20 niceness most favorable, 19 niceness less favorable)
nohup command run command ignoring hangup signals
kill -SIGNAL PID send SIGNAL to process PID
command1 && command2 launch command1 and if it succeed launch command2

Signals and their identifiers:

SIGINT	2	interrupt process
SIGSTOP	19	stop temporarily process
SIGTSTP	20	send stop from terminal
SIGCONT	18	continue a stopped process
SIGKILL	9	terminate a process
SIGUSER1	10	user signal
SIGUSER2	12	user signal

Foreground and background processes

command & start command in background
fg put a background process foreground
bg put a stopped process background

References

- [1] Alexis de Lattre et al. *Formation Debian GNU/Linux*. <http://formation-debian.via.ecp.fr/>.
- [2] *Bash Hackers Wiki*. <http://wiki.bash-hackers.org/doku.php>.

Redirection and pipes

command < file use file as input
command > file use file as standard output
command >> file use file as standard output (append)
command 2>&1 file send error output to standard output
command1 | command2 send command1 output to command2 input

Connecting to remote machines

ssh user@computer connect as user on computer
scp file user@computer:PATH copy file as user on computer in PATH
rsync -avz dir user@computer:PATH copy directory dir as user on computer in PATH