

GNU/Linux most wanted

Summary of most useful commands

©Copyright 2006-2005, Free Electrons.

Free to share under the terms of the Creative Commons Attribution-ShareAlike 2.5 license

(<http://creativecommons.org>)

Sources, translations, updates, command and concepts details on our free training materials:

http://free-electrons.com/training/intro_unix_linux

Thanks to Michel Blanc, Hermann J. Beckers and Thierry Grellier.

Handling files and directories

Create a directory:

```
mkdir dir
```

Create nested directories:

```
mkdir -p dir1/dir2
```

Changing directories:

```
cd newdir
```

```
cd .. (parent directory)
```

```
cd - (previous directory)
```

```
cd (home directory)
```

```
cd ~bill (home directory of user bill)
```

Print the working (current) directory:

```
pwd
```

Copy a file to another:

```
cp source_file dest_file
```

Copy files to a directory:

```
cp file1 file2 dir
```

Copy directories recursively:

```
cp -r source_dir dest_dir  
rsync -a source_dir/ dest_dir/
```

Create a symbolic link:

```
ln -s linked_file link
```

Rename a file, link or directory:

```
mv source_file dest_file
```

Remove files or links:

```
rm file1 file2
```

Remove empty directories:

```
rmdir dir
```

Remove non-empty directories:

```
rm -rf dir
```

Listing files

List all “regular” files (not starting with `.`) in the current directory:

```
ls
```

Display a long listing:

```
ls -l
```

List all the files in the current directory, including “hidden” ones (starting with `.`):

```
ls -a
```

List by time (most recent files first):

```
ls -t
```

List by size (biggest files first)

```
ls -S
```

List with a reverse sort order:

```
ls -r
```

Long list with most recent files last:

```
ls -ltr
```

Displaying file contents

Concatenate and display file contents:

```
cat file1 file2
```

Display the contents of several files (stopping at each page):

```
more file1 file2
```

```
less file1 file2 (better: extra features)
```

Display the first 10 lines of a file:

```
head -10 file
```

Display the last 10 lines of a file:

```
tail -10 file
```

File name pattern matching

Concatenate all “regular” files:

```
cat *
```

Concatenate all “hidden” files:

```
cat .*
```

Concatenate all files ending with `.log`:

```
cat *.log
```

List “regular” files with `bug` in their name:

```
ls *bug*
```

List all “regular” files ending with `.` and a single character:

```
ls *.*?
```

Handling file contents

Show only the lines in a file containing a given substring:

```
grep substring file
```

Case insensitive search:

```
grep -i substring file
```

Showing all the lines but the ones containing a substring:

```
grep -v substring file
```

Search through all the files in a directory:

```
grep -r substring dir
```

Sort lines in a given file:

```
sort file
```

Sort lines, only display duplicate ones once:

```
sort -u file (unique)
```

Count the number of words in a file::

```
wc -w file
```

Count the number of characters in a file:

```
wc -c file
```

Changing file access rights

Add write permissions to the current user:

```
chmod u+w file
```

Add read permissions to users in the file group:

```
chmod g+r file
```

Add execute permissions to other users:

```
chmod o+x file
```

Add read + write permissions to all users:

```
chmod a+rw file
```

Make executable files executable by all:

```
chmod a+rX *
```

Make the whole directory and its contents accessible by all users:

```
chmod -R a+rX dir (recursive)
```

Comparing files and directories

Comparing 2 files:

```
diff file1 file2
```

Comparing 2 directories:

```
diff -r dir1 dir2
```

Command help

Basic help (works for most commands):

```
grep --help
```

Access the full manual page of a command:

```
man grep
```

Looking for files

Find all files in the current (`.`) directory and its subdirectories with `log` in their name:

```
find . -name "*log*"
```

Find all the `.pdf` files in `dir` and subdirectories and run a command on each:

```
find . -name "*.pdf" -exec xpdf  
{ } ';'
```

Quick system-wide file search by pattern

(caution: index based, misses new files):

```
locate "*pub*"
```

Redirecting command output

Redirect command output to a file:

```
ls *.png > image_files
```

Append command output to an existing file:

```
ls *.jpg >> image_files
```

Redirect command output to the input of another command:

```
cat *.log | grep error
```

Job control

Show all running processes:

```
ps -ef
```

Live hit-parade of processes (press `P`, `M`, `T`: sort by Processor, Memory or Time usage):

```
top
```

Send a termination signal to a process:

```
kill <pid> (number found in ps  
output)
```

Have the kernel kill a process:

```
kill -9 <pid>
```

Kill all processes (at least all user ones):

```
kill -9 -1
```

Kill a graphical application:

```
xkill (click on the program window  
to kill)
```