

Some useful UNIX Commands written down by Razor
for newbies to get a start in UNIX

15th Jan. 2000 / 3:55 am

Part 1: Working with files and rights

```
cp <source> <dest>          --- Copies the source file to dest
mv <source> <dest>          --- Moves source to dest (if a directory,
                             else it renames source to dest)

cd <directory>              --- Changes the current directory
pwd                          --- Shows the current directory
mkdir <directory>           --- Creates a directory
rm <directory>              --- Deletes a directory, if empty
rm -R <directory>           --- Deletes a whole directory with subdirectories
                             (like DELTREE <directory>)
rm <file>                   --- Deletes a file

chmod userMODErights        --- Changes the mode of a file
                             user=(u)ser   (current user)
                             (g)roup   (group of current user)
                             (o)ther   (all, except user&group)
                             (a)ll     (all :) )
                             MODE=(+) add rights
                             (-) remove rights
                             rights=(r)eadable
                             (w)ritable
                             e(x)ecutable

df                            --- Shows free disk space

find / -name "<file>"        --- Searches the whole tree for a file (>
DIR <FILE> /S)
whereis <file>               --- Searches the path for a file
grep -l "<Text>" <files>    --- Shows all files, which contain the
given <Text>
cmp <file1> <file2>         --- Compares <file1> with <file2>
diff <file1> <file2>        --- Compares <file1> with <file2> and show
the difference

                             between them
head <file>                  --- Shows the first 10 lines of the file
tail <file>                  --- Shows the last 10 lines of the file
```

```

nl <file>          --- Shows the lines of the file numbered
sort <file>       --- Shows the lines of the file in sorted order
strings <file>    --- Shows only the readable strings of a
file
wc <file>        --- Shows lines, words, bytes of a file

file <file>       --- Gives information about a file
touch <file> -a -m -c -t MMDDhhmmCCYY --- Changes the timestamp of a file
-a (Change Access-time)
-m (Change Modification-time)
-c (Don't create files, that don't exist)

pico [file]      --- An easy to use ASCII-Editor.
script <file>    --- Copies the text, that is entered
between this command
and the "exit"-command in <file>

lpr <file>       --- Prints a file
lprm <file>      --- Removes file from Printing Queue

sudo <file>      --- Executes file with SuperUser-Rights (Only
when
reported
user is in SUID-List, else it will be
to root)

```

Part 2: Working with archives

```

-----
tar -cvf <archive.tar> <files>          --- Creates <archive.tar> and stores
the files (uncompressed)                in this archive

tar -xpvf <archive.tar> <files>        --- Extracts the files from the
archive

tar -tf <archive.tar> | less            --- Shows the contents of an archive

gzip <file>                             --- Compresses the <file>
gzip can only compress one file, so you'll
have to                                  archive the files with tar and then gzip
them.                                     This will give you a file.tar.gz

tar -zxf <file.tar.gz>                  --- Decompresses and Extracts the files

tar -M -cvf /dev/fd0h1440 <files>      --- Creates a Multi-Volume-Archive (for
floppy discs)
tar -M -xpvf /dev/fd0h1440             --- Extracts a Multi-Volume-Archive

```

Part 3: Working with filesystems

```

-----
Formatting a Floppy Disc.

```

Step 1 -- Format the disc

```
fdformat /dev/fd0H1440
```

Step 2 -- Create a Filesystem on the disc

```
mkfs -t <fs> -c /dev/fd0H1440
```

<fs> stands for the filesystem. This can be ext2/minix/msdos (for floppys normal is minix)

To use a floppy disc or a cd-rom, you'll have to mount them before.

```
mount <device> <directory>
```

Floppy: mount /dev/fd0 /floppy (you can use any directory, but floppy would be normal, I think.)

CD-ROM: mount /dev/hdc /cdrom

Part 4: Working in Networks

Connect to a remote machine, and execute programs on it.

```
telnet remote.host.org      --- Connect to a remote machine  
                             You can then execute programs on the remote  
machine in the              terminal.
```

```
uname -a                    --- Gives information about the current system
```

```
w                            --- Shows who is currently logged in and what he
```

```
is running
```

```
finger                      --- Shows who is logged in.
```

```
                             With "finger <user>" you can get more
```

```
information about
```

```
specific users
```

```
passwd                      --- Change User Password
```

```
write <user> [<tty>]        --- Write a message to <user>.
```

```
terminal, you can
```

```
If <user> is logged on more than one
```

```
<user> must
```

```
specify it in <tty>. To answer you, the
```

```
also start "write"
```

```
chsh                        --- Changes the login shell
```

Part 5: Programming

```
cc -o <Output file> <Source file> --- compiles the Source file into Output
file
gcc -o <Output file> <Source file> --- same
gcc -o <Output file> -l<libname> --- links the library <libname> to the file
```

The C functions are declared in manual 3 ==> man 3 <function>

Part 6: Doing jobs in the background

If you add an & at the end of a command, it will start in the background.
Exmpl: fdformat /dev/fd0h1440 &

```
ps -a --- Shows all current processes
kill <PID> --- End process
kill -9 <PID> --- Forces the process to end
```

You can stop the processes on two ways and make them to jobs.

```
CTRL-C --- Stop process
CTRL-Z --- Stop process temporarily

bg <job> --- Brings a job into background
fg <job> --- Brings a job into foreground
kill <%job> --- End job

jobs --- Shows all current jobs
```

Part 7: Escape Sequences

\a Bell
\b backspace
\f feed
\n linefeed
\r carriage return
\t tab horizontal
\v tab vertical

Ending

This text should just provide some basic skills, so that you can start using UN*X. I know it isn't declared very much, but i like short tables, being able to lookup a certain command. If you need to know more specific switches you can always consult the manuals with: man <command>.

If you are interested in more help or have ideas which should be inserted contact me at razor99@gmx.de.

If there is anyone interested in this, I will perhaps rewrite this doc and write more to the commands.

And I hope you excuse my style, but it is now 5:05 in the morning and I'm really tired...

RAZOR