

linux CheatSheet

Top-level Directories

- / : root
 - /etc : Program Configuration Files
 - /var : Frequently changing content (ex. logs)
 - /home : User account files
 - /sbin : System binary files
 - /bin : User binary files
 - /lib : Shared libraries
 - /usr : Third-party libraries

Basic Commands

ls

- lists content of a directory

```
ls [/aDirectoryName]
```
- lists content with the long flag `-l` and human-readable flag `h` displays file permissions and size information

```
$ ls -lh /users/max/documents
```
- lists with the recursive flag `-R` everything in and under a directory

```
$ ls -R /users/max/documents
```

pwd (present working directory)

- print name of current/working directory
- using flag Logical `-L` to include symlinks

cd (change directory)

- ex. : moving to root directory :

```
$ cd \
```

rm (remove)

- removing a file :

```
$ rm myFile
```
- removing an empty directory:

```
$ rmdir myEmptyDirectory
```
- removing a non-empty directory and its contents recursively :

```
rm -r myDirectory
```

cp (copy)

- copy a file to a specified directory

```
$ cp myFile /home/max/anotherDirectory
```

mv (move)

- move a file to a specified directory

```
$ cp myFile /home/max/anotherDirectory
```

Other Commands

cat (concatenate)

- prints file content to output

less

- quickly display less than a complete file contents
- scroll document's content with arrows
- quit by typing 'q'

man

- brings up a one-page interface to reference manuals about a linux command

```
ex. : $ man ls
```
- type `\` (backslash) followed by keywords to search into document

```
move forward to next keyword by typing 'n'
```
- quit by typing 'q'

touch

- updates file timestamps
- if file doesn't exist it will create it

stat (status)

- displays file or file system status

```
displays file's inode (metadata) information
```

| (pipe)

- creates a unidirectional data channel
- takes the output of a command to feed it as the input of another

```
$ journalctl | grep myFile.php
```
- you can chain as many pipes as you wish

```
$ journalctl | grep myFile.php | grep error
```

info

- reading documentation in Info format
- useful, when you don't know the name of a command you want to use
- click Enter on underlined text to follow links

```
'u' will bring you back one level
```

```
'q' to exit
```

journalctl

- tool to query the contents of the systemd journal

```
ideally do filter query with grep
```

```
$ journalctl | grep myPage.php
```

arch

- prints your system architecture (ex: x86_64)

Keyboard Shortcuts

TAB (completion)

- press TAB to complete a command

```
$ touch myNewFile
```

```
$ rm my<TAB>
```

Special Characters

* (gobbling)

- wildcard used to designate all files in a directory

```
$ mv * /home/max/archive
```
- wildcard used to designate any characters

```
$ git add linux*.tex, linux*.pdf
```

? (question mark)

- represents or matches a single occurrence of any character

```
there are files named 'file1', 'file2', 'file3'
```

```
$ rm file? /home/max/documents
```

\ (backslash)

- The backslash character can be used to conveniently break a long command into multiple lines on the command line.

```
# yum install lxc lxc-templates \libcaps-devel libcgrouper busybox
```

System Administration

systemd

- first process to run on a system

- show all services and processes running

```
$ systemctl list-units -type service  
-state running
```

or

```
$ systemctl -no-pager | grep service |
```

```
grep running | column -t
```

- show all installed unit-files

```
$ systemctl list-units -type service  
or
```

```
$ systemctl list-units -type service  
-state running -no-legend
```

- show processes that executed then exited

```
$ systemctl list-units -type service  
-state exited
```

- show processes that have failed

```
$ systemctl list-units -type service  
-state failed
```

- *In order to pass the output to *stdout* instead of a pager

add argument ‘--no-pager’

Last compiled : (April 8, 2020)