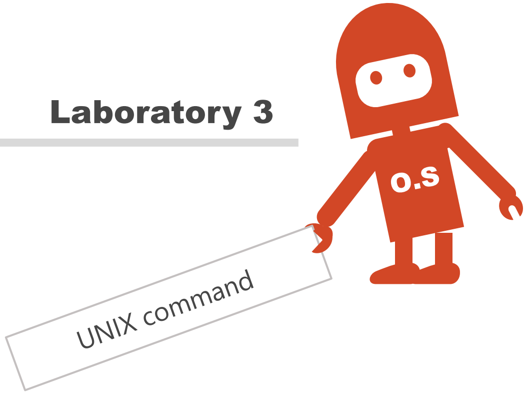
OPERATING SYSTEMS UNIX / LINUX

Stefan Bartha



Structure of a UNIX command

```
command [options] [value]
```

- `command` the first word in a command line. It’s a small program that completes a task
- `options` represent the option for executing the command
  - **short option** that means a **single letter** preceded by a single hyphen -
  - **long option** that means a **word** preceded by a single double hyphen --

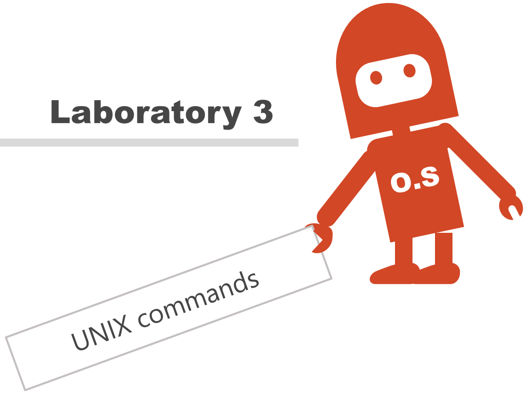
Examples :

- command only: `pwd, ls, history`
- command + option: `ls -l, ls -a ls --all`
- command + value: `mkdir dir_name touch file_name`
- command + option + value: `ls -l /etc`

UNIX Command

Operating Systems

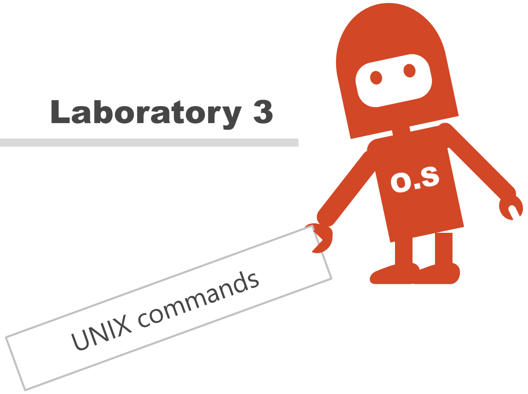
Laboratory 3



Let’s see some commands without option or value

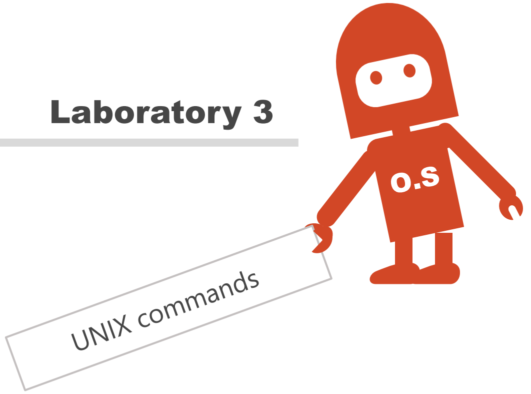
pwd	return current directory
ls	list content of current folder
history	provide a list of commands used for an account
cd	Change the current directory. Used without parameters it will jump to home directory cd / pwd cd pwd
exit	close current session
whoami	return current user name
who	return all ussers logged into system.
date	return current date
uptime	Time passed from system up
df	Reports the system's disk space usage for all mounted filesystems
top	Provides a dynamic real-time view of a running system’s processes
uname	Shows system information, such as the operating system name. Without parameters, it typically prints the kernel name.

UNIX commands



Let’s see some commands with option

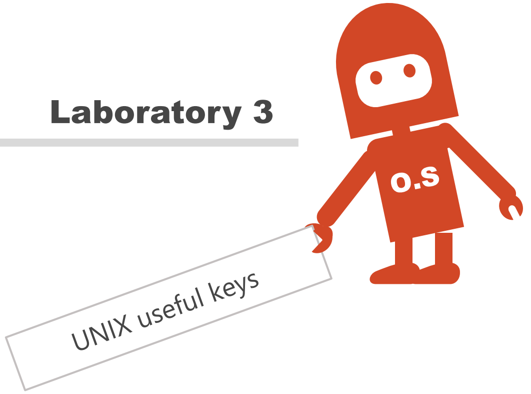
ps	**ps –F** – returns all processes with all info
ls	**ls -a, ls –all** → display all content of folder also folders with . prefix (hidden) **ls -l** → long listing format **ls -ls** → display as list with block size used on disk by file **ls -lh** →human readable. **ls -clt** → display by last change descending **ls -cl** → display by last change ascending **ls -F** → Will add / at the end of each directory **ls -r** → Display files and directories in reverse order. Try **ls -l** and **ls -lr** **ls -R** → Display full path of each file or directory **ls -full-time** → Display full time information about files
history	**history -c** → Delete current session history
df	**df –h** → display info about used space into an human readable format **df –T** → display type of disk foreach disk mounted
uname	**uname –a** → display all informations regarding operating system **uname –r** → display kernel version
lsblk	**lsblk –J** → info about available drives in json format **lsblk –m** → info about available drives with permisions
journalctl	**journalctl –system** → system log **journalctl –user** → log for current user



Let’s see some commands with option and value

`ls -l <directory_name>`	ls -l /etc
`grep -i “<charters>” <destination>`	touch my_file.txt nano my_file.txt → and some text in file that contains “example” grep –i “example” myfile.txt → Search for the string of characters, ignoring the difference between uppercase and lowercase.
`find <location> -name <keyword>`	find / -name myfile.txt → Search the entire file system for a file named myfile.txt
`chmod +x <file_name>`	chmod a-w myfile.txt chmod +w myfile.txt
`cp -r /source/directory /destination/directory`	**uname –a** → display all informations regarding operating system **uname –r** → display kernel version
`rm -f -r <directory_name>`	delete folder recursively
`mv old_location new_location`	move a file from a location to another location

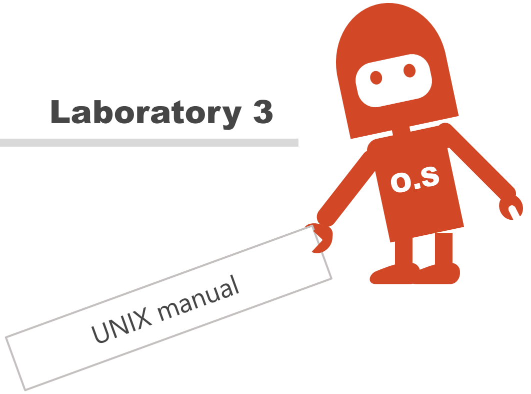
UNIX commands



Let’s see some useful key combinations

`TAB`	autocompletion of the command line (ex. long file name, type start of it then tab).
`Arrows up and down`	navigate through commands history typed by you
`Ctrl-C`	Stop the execution of the current program
`Ctrl-Z`	Suspend the execution of the current program (remains on in background)
`Ctrl-D`	Close the working session (sometimes equivalent with EOF)
`Ctrl-S`	Lock the console (never saves a file) you can’t type anything anymore for putty. No effect in Moba
`Ctrl-Q`	Unlock the console after using Ctrl+S by reflex to try to save a file
`Ctrl-K`	Cut/copy text from current position to end of line
`Ctrl-Y`	Paste copied text with Ctrl-K (you can also use mouse select and then right mouse click to paste – it will paste the selected)
`Ctrl-R`	Search commands history
`Ctrl-A`	Move cursor at the beginning of cmd line
`Ctrl-B`	Move cursor back one char
`Ctrl-F`	Move cursor forward one char
`Ctrl-E`	Move cursor at the end of the cmd line

UNIX useful keys



How we can find details about a command ?

man command_name

Command_name is the command name that we are looking for info

Examples :

- Getting help for manual : **man --help**
- Getting details about a command : **man pwd**
- How to get just a section of manual. Get only command description : **man – f pwd, man –whatis pwd**
- If we remember just some part of command but not entire name we can use : **man --apropos pw**

man –k pw

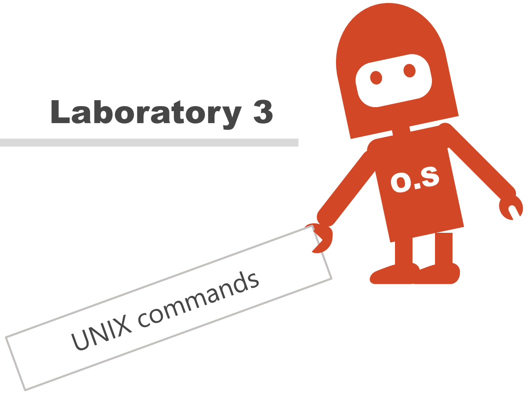
Practice 5 min :

- See what is displaying on your machine: **man --help**
- Search for commands that contains an keyword: **man --apropos** …
- Get short description for a command: **man --whatis** …
- Try something like: **man –K ip –o pwd.** Try to explain what this command do
- Get details about history command from man and use it. What it's doing

UNIX Manual

Operating Systems

Laboratory 3



Let’s see some usual commands

Commands for navigating through the file system

pwd	**p**rint **w**orking **d**irectory	Print the path cu current directory
ls	**l**i**s**t	List the content of the current directory
cd dir	**c**hange **d**irectories	Change current directory to specified dir

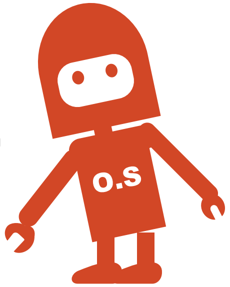
Commands for working with directories and files

mkdir dir_name	Create a directory named **dir_name**
touch file_name	Create empty file name named **file_name**
rmdir dir_name	Remove directory named **dir_name**
rm file_name	Remove file named **file_name**
cp src_dir dst_dir	Copy the **src_dir** into **dst_dir**
cp src_f dest_f	Create a copy of **src_f** named **dest_f**
mv src_dir dst_dir	Move/rename directory **src_dir** into **dst_dir**
mv src_f dest_f	Move/rename src_f in dst_f
cat file_name	List the content of file **file_name**

UNIX Commands

Operating Systems

Laboratory 3



Let’s practice some usual commands

5 min practice navigation commands

- Read about commands (`pwd , ls, cd`) by using man command and `cmd_name --help`
- Navigate to root folder `cd ..` Type `ls -ls` Try command `cd ~` followed by `pwd`. What you seen ?. Now try `cd /` followed by `pwd`
  Not sure where you are ? Try ls –ls. Now let’s try `cd~; ls /bin; ls bin` . What you find and why ?
- Try different `ls`

`Ls -l`	shows file or directory, size, modified date and time, file or folder name and owner of the file, and its permission
`ls -a`	List all files including hidden files starting with . (hidden files). Try it in your user folder
`ls -lh`	shows sizes in a human-readable format. Try `ls -ls`
`ls -F`	Will add / at the end of each directory
`ls -r`	Display files and directories in reverse order. Try `ls -l` and `ls -lr`
`ls -R`	will list very long listing directory trees
`ls -lt`	List files / directory by last modify date ordered desc . Try after that `ls -ltr`
`ls -i (ls -li)`	List files/directory with index node (inode)
`ls -full-time`	List the content with full time information

- `ls | less` ls with pipe operator means redirecting of output to another command.
  List the content of directories : `/bin /usr /etc usr/include` by using `less`

UNIX commands

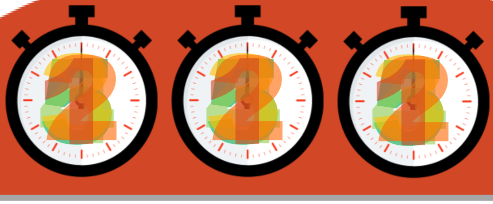
Operating Systems

Laboratory 1

Let’s practice some usual commands

15 min practice commands for working with directories and files

- Read about commands (**mkdir**, **cp**, **rmdir**, **rm**, **touch**, **cat**, **echo**) by using man command and **cmd_name --help**
- In your user dir create a new directory called **lab_2** by using **mkdir**. In **lab_2** create **dir1**. Go to your home directory in one command by using **cd ~**. From your home directory create in one line directory **dir1_1**. Now try to create in **lab_2** in one line **dir2/dir2_1**. See what is happening. Read the manual search for a solution
- Verify that above structure is ok by using **ls**. Ok it’s becoming complicate ? Try **tree** command. It’s not working ? Install the app and try again
- In lab_2 use the following command **mkdir –m 000 dir2** and **mkdir –m 777** dir3. Try ls –l. What you see different to these directories. Try to enter in **dir2** by using **cd** command
- In **dir1** create a new file. We’ll call it **file1.txt** (.txt doesn’t have much sense in UNIX but we are using it in this case to be a more friendly name). Use **touch.**
- In above created file I want some text. Use **cat > file1.txt. Ctrl-D** for exit. Again **cat file1.txt**
- Ok above is the hard way. More simple **nano file2.txt.** Another text editor is **joe.** Try it
- Go to **lab2->dir1.** Remove **dir1_1.** Back to **lab2.** Try to remove **dir2.** If you can’t search in manual
- Go to lab2 and create **dir3 dir4** and **dir5** in one command
- Create 1 file in each directory named **dir3_file1.txt, dir4_file2.txt** …
- Copy dir3->**dir3_file1.txt** to **dir4** without specify a name. Again and specify name
- Move a file from **dir4** to **dir3**
- Move **dir3** to **dir4**



UNIX commands

Operating Systems

Laboratory 3

Let’s practice some usual commands

5 min practice see some text editors. Copy / remove files or directories

1. Go to dir1. Try **vi file1.txt.** Press **INS** key to change text. Press **ESC** key to finish editing. **:w** to save changes **:q** to exit or **:q!** to exit without changes. Verify your changes with **cat file1.txt**

2. Try **vim file1.txt.**Use the same commands like above

3. Try **joe file1.txt.** Out from it **Ctr+C**

4. Try **emacs file1.txt.** Is not working install if you want sudo apt-get install emcas

5. Try **nano file1.txt. Ctrl+X** for exit. Details in bottom side of screen

6. Go to **lab2** and create **dir5 dir6** and **dir7** in one command **mkdir** …

7. Create 1 file in each directory named **dir5_file1.txt, dir6_file1.txt, dir7_file1.txt.** Can be done in single command

touch dir5/dir5_file1.txt dir6/dir6_file1.txt dir7/dir7_file1.txt

8. Copy **dir5_file1.txt** to directory dir6 without name. Use **cp** command. Use ls or tree to see results

9. Move **dir6/dir6_file1.txt** to **dir7** without specify a name. Use **mv dir6/dir6_file1.txt dir7**

10. Move file **dir5_file1.txt** to **dir6** with name dir5indir6_file.txt. Use tree to see results

11. Delete **dir5** use **rmdir** command. Remove **dir6**

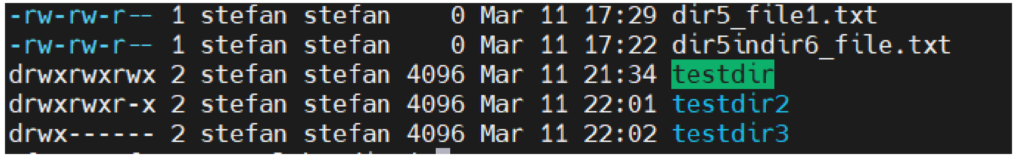
UNIX commands

Operating Systems

Laboratory 3

UNIX access rights

What are UNIX permissions ?



```
-rw-rw-r-- 1 stefan stefan    0 Mar 11 17:29 dir5_file1.txt
-rw-rw-r-- 1 stefan stefan    0 Mar 11 17:22 dir5_indir6_file.txt
drwxrwxrwx 2 stefan stefan 4096 Mar 11 21:34 testdir
drwxrwxr-x 2 stefan stefan 4096 Mar 11 22:01 testdir2
drwx------ 2 stefan stefan 4096 Mar 11 22:02 testdir3
```

drwxrwxrwx

Specify if is a file (**-**) or directory (**d**)

Set of permissions for owner **u**

Set of permissions for users group **g**

Set of permissions for others **o**

- **r** (read) - right for reading the file/dir contents
- **w** (write) - to write/modify in the file or dir content (create/delete files in directory)
- **x** (execute) - right to execute the file or have dir access

r w x

b b b

- **000 → 0** no rights
- **001 → 1** execution rights
- **010 → 2** write rights
- **100 → 4** read rights

- **011 → -wx** =000+ 010+001 = 0+2+1 =3
- **111 → rwx** = 100+010+001 = 4+2+1 =7
- **101 → r-x** = 100+000+001 = 4+0+1 =5
- **110 → rw-** = 100+010+000 = 4+2+0 =6

UNIX access rights

Operating Systems

Laboratory 3

How to change permissions ?

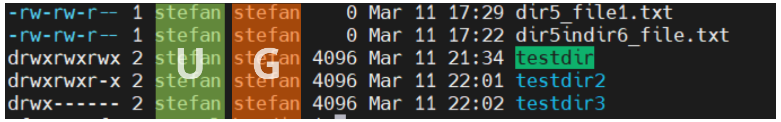
chmod [permision] [file/dir]

Permissions can be set in 2 ways :

- symbolic change **chmod [group][operator] [description] [file|directory]**
  chmod u=rx,g+x,o-r myfile explicit rx for owner, add x to group, remove r for others
- numeric numeric **chmod [number_description] [file|directory]**
  chmod 731 myfile means rwx-wx—x full owner, rx group, x others

Groups and users

- (the information about groups is stored in /etc/group **cat /etc/group**.



```
-rw-rw-r-- 1 stefan stefan    0 Mar 11 17:29 dir5_file1.txt
-rw-rw-r-- 1 stefan stefan    0 Mar 11 17:22 dir5indir6_file.txt
drwxrwxrwx 2 stefan stefan 4096 Mar 11 21:34 testdir
drwxrwxr-x 2 stefan stefan 4096 Mar 11 22:01 testdir2
drwx------ 2 stefan stefan 4096 Mar 11 22:02 testdir3
```

- **groups** display current user groups **groups [user_name]** user groups for specifdied user
- **groupadd** used to add a new group.
- **usermod -a -G group_name user_name** is used to add user to an group. Also can be added by editing etc/group . To remove user from an group **gpasswd -d [user_name] [group]**
- **adduser [user_name]** used to add a new user
- **chown [user_name] [file/dir] chgrp [user_name] [file/dir]**

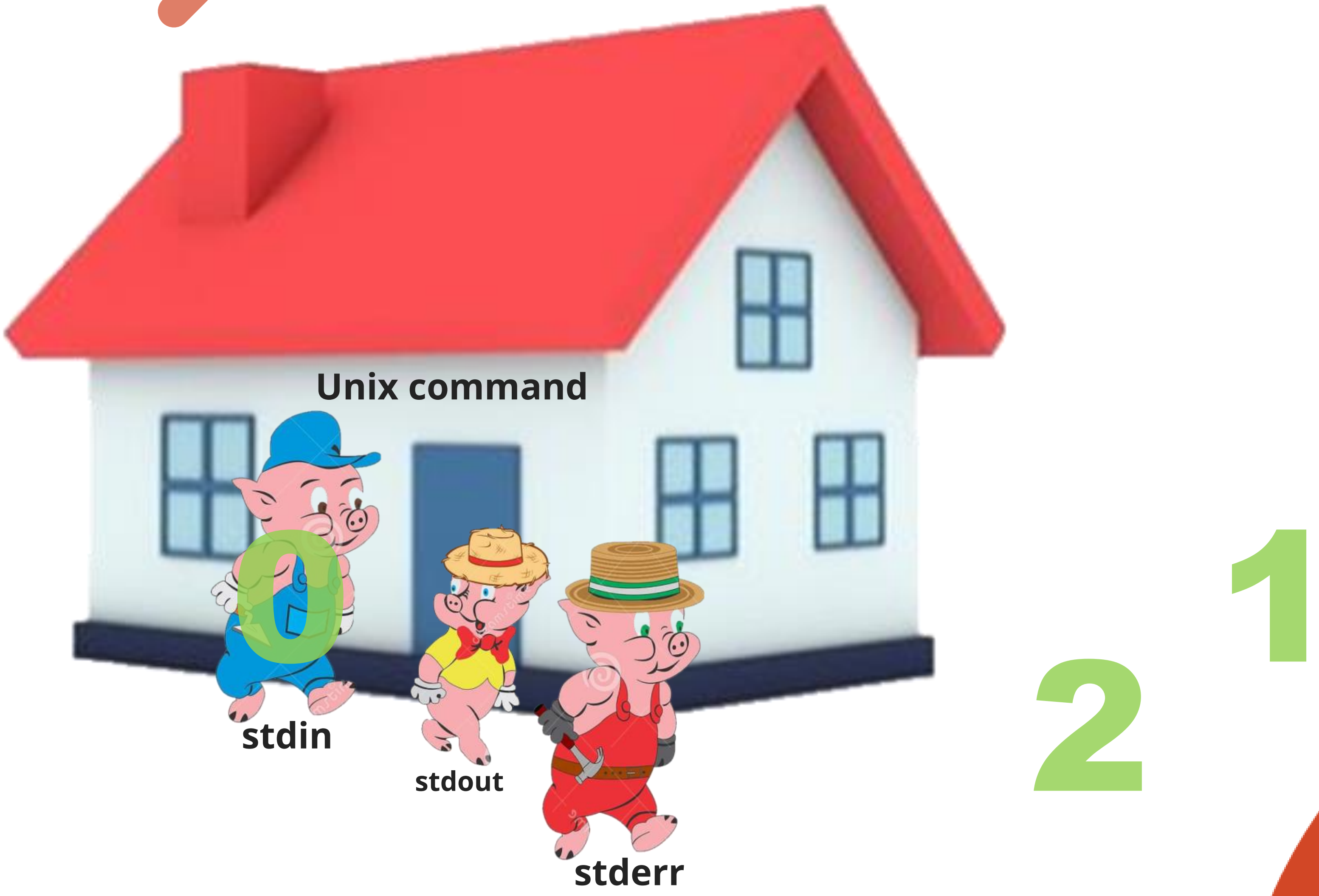
UNIX access rights

Operating Systems

Laboratory 1

Data streams

stdin , stdout , stderr



Data streams