



Implementing Commands



Task



clarusway



lessons



linux.txt

“I love linux”



html.txt

“I can create a website”



materials



lms.txt

pre-class

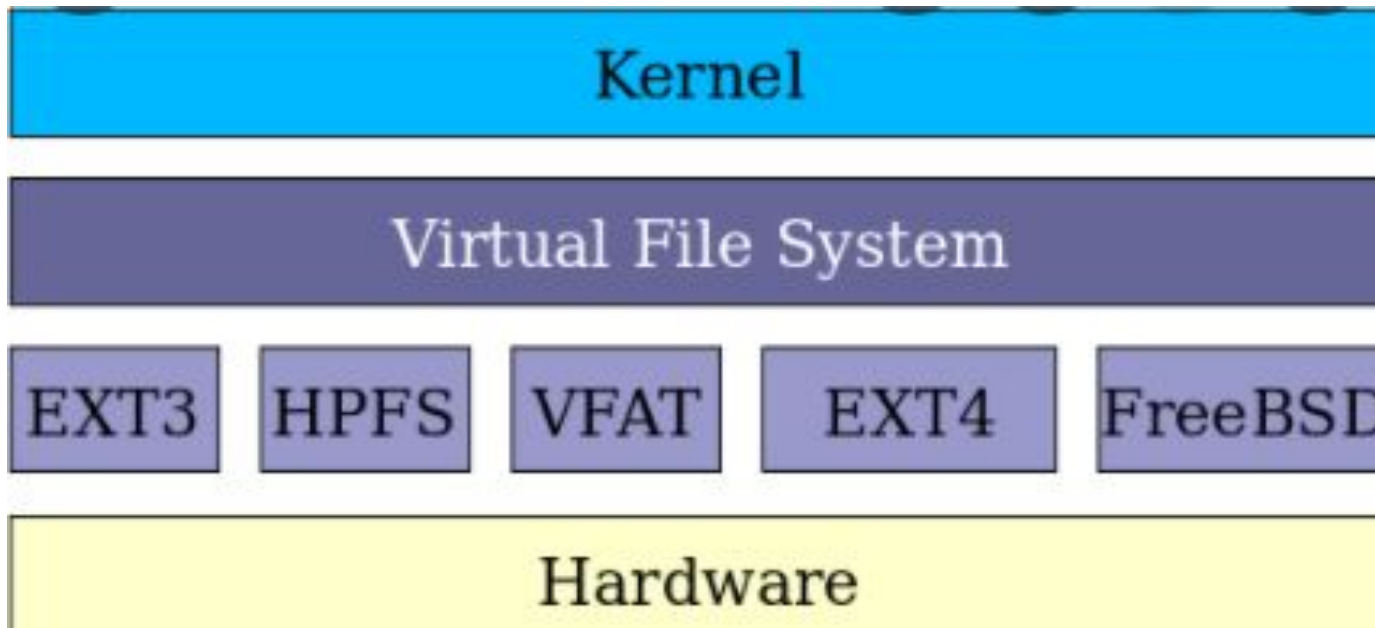


try-it.txt

post-class



Linux file systems



Solution

```
136 ls
137 mkdir clarusway
138 cd clarusway
139 mkdir lessons
140 cd lessons
141 touch linux.txt
142 echo "I love linux" > linux.txt
143 touch html.txt
144 echo "I can create a web site" > html.txt
145 cd ..
146 mkdir materials
147 cd materials
148 mkdir pre-class
149 cd pre-class
150 touch lms.txt
151 cd ..
152 mkdir post-class
153 cd post-class
154 touch try-it.txt
```

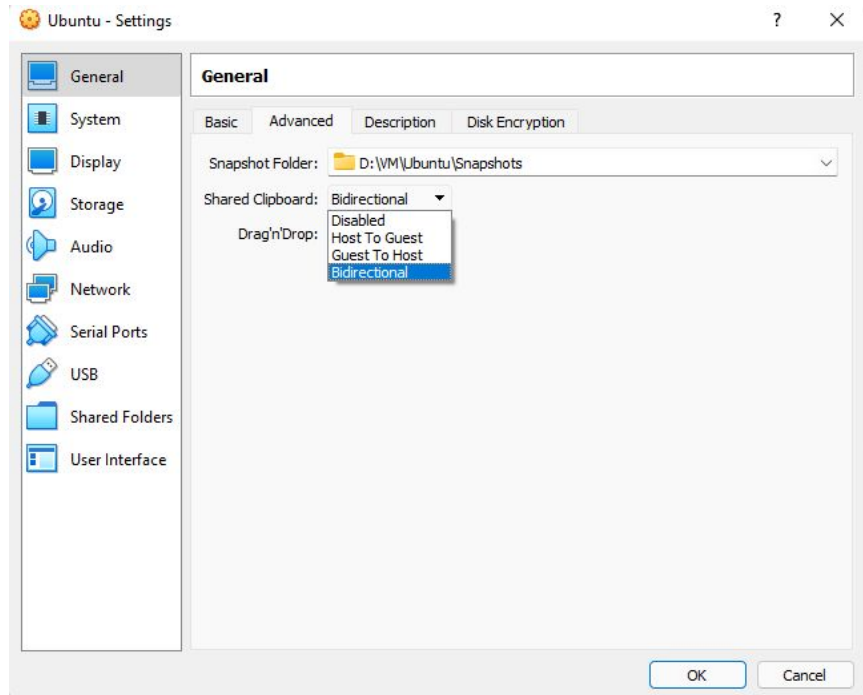


Kahoot!



Preparation

- If not open; Boot-up Ubuntu in VBox, login.
- Select and find Devices\Insert Guest Additions CD image.
- Open up a terminal and execute the command “Reboot”
- After reboot login to VBox, open terminal.
- `cd /`
- `sudo rmdir ~/Projects`
- `mkdir` under `~/Projects`, these folders : Common DB Docs
- Output Test Web Worklog
- `clear`





- Session 14 : 5pts
Timeout : 5 min

Investigation of Linux directory Structure :

/ this is the root directory cd /

/bin : Essential user binaries

/sbin : Essential system administration binaries.

/boot : Files needed to boot the system.

/cdrom: Temporary location for the cd-rom content (not standard)

/dev : Device files, linux treats devices as files. for ex (sda : sata drive), dev/null is special, no input-output, can be ignored.

/etc : configuration files exist here.

/home: Home folder for each user exists here. Every user contains a folder in.

/lib : Shared libraries, all applications use this folder as a common share folder.

/lost+found : if a system crash happens, linux saves the content here if possible, next boot it should be checked if needed.

/media : Removable media treated as file again.

/tmp : Temporary mount points. For instance if you mount a windows partition, it would be /mnt/windows.

/opt : The optional packages for apps that do not obey the standart file hierarchy.

/proc : Similar to dev, but more specific to running processes in memory, files created by those processes.

/root : It is the home directory for the current user. which is distinct from the / system root directory.

/run : it is a fairly new standard. it gives the apps to store transient storage area, like sockets and process ID's.

/srv : Service data, just like IIS in windows Apache HTTP Server's data can be stored here.

/usr : User binaries, read-only data. Non-essential applications, user specific applications.

/var : This is a writable counterpart of /usr directory. /var/logs are pretty much used to check application logs.

Filesystem Hierarchy Standard (FHS) defines this structure, but it doesn't mean that non-standard folders can't exist.



Review :

```
cd / takes us to the root folder
cd ~/ takes us to the home folder of the user
cd .. One directory up
cd [Path] takes us the the desired directory
man find
user@host:/$
pwd
ls -a (all items including . and ..)
ls -l (long listing)
mkdir dir1 dir2 dir3 dir4 dir5
rm -r dir1
touch filename
rm filename
cp filename -n (With or without interaction switches i and n)
mv filename
echo '2022 Log File contents' > SystemLog.log
cat filename
sudo find -name '*.log'
sudo find ~/ /var/ -name '*.log'
open 1.html
alias ll='ls -al'
alias mytop='top -u root' // IF another user needed than set the name.
tail -f /var/log/auth.log // -f monitors follows the file for changes
grep pam_unix /var/log/auth.log // displays if pam_unix exists in the file.
```




New Commands:

`dmesg` : displays the boot messages, this is just like to loggers in programming languages.

`cat /proc/cpuinfo` : see the CPU information in detail.

`free -h` : display the free memory in detail

`lshw` : hardware configuration

`lsblk` : some information about block devices

`lspci -tv` : PCI devices in tree structure

`lsusb -tv` : USB devices in tree structure

`dmidecode` : hardware information in BIOS

`hdparm -i /dev/disk` : display disk data

`hdparm -tT /dev/[device]` : read speed test

`badblocks -s /dev/[device]` : display bad blocks on disk

 : `badblocks -s /dev/disk/by-id/ata xxx` choose the disk.

`fsck [disk-or-partition-location]` : run a disk check



- Create file, folder, modify and save using NANO editor
- cd /
- mkdir ~/Projects
- touch ~/Projects/data.json
- nano ~/Projects/data.json
- write "This is data.json file. It will contain json elements" and save the contents.
- cat data.json to display the output.



- Create file, folder, modify and save in VIM Editor
- cd /
- touch ~/Projects/config.json
- vi ~/Projects/config.json
- write "This is config.json file. It will contain configuration elements" and save the contents.
- cat ~/Projects/config.json to display the output.



By using an editor in GUI environment
modify the contents of data.json.
Write “This is from GUI Editor” inside it and save it please.



- `cd /`
- `w` (display the logged in users)
- `whoami` (display the current user)



Find how to display date time values in Terminal in Ubuntu.
You may check web, or man pages.



- Find your local IPv4 address in Ubuntu Terminal
- Find your local IPv4 address in Settings\Network



- date
- !!
- type in !! 5 times to see what it does.
- ip a
- tell where is your local IP address in the figure.
- !!



- Your linux environment has some issues, you can't figure out what's wrong with it. You decided to investigate the boot-up logs, how do you do that?
dmesg displays the list of logs, please find an error message
(Red one's are the error logs)



- Your python application crashes several times while working, you need to check if it's because no free RAM space available in your PC.
How do you do that from Terminal easily?



- There is a utility called `htop` in Linux. It displays the running processes and corresponding consumed memory and details.
- Install this application by using `sudo apt-get install htop`
- Find the name of the process that has the highest CPU usage
- Check the memory and CPU usage percentages.



- `cd /`
- We had created `data.json` under `/Projects` folder. Stay in root, do the actions below ;
- `rm /Projects/ data.json, config.json`
- `mkdir in /Projects Worklog Output`
- `touch /Projects data.json config.json`
- edit contents in VI editor and NANO editor, save
- `cp data.json to Worklog`
- `mv config. json to Output`
- try to remove `rmdir, rm Worklog and Output` from root directory.



- Install bashtop yourself and see the free memory in your system.
You know it already that `sudo apt-get install bashtop` will do it.
- After installing find how to open-up the bashtop tool.



- `cd /`
- `ls`
- `lspci`
- `lshw`
- `lsblk`
- `lsusb`

Observe the outputs of each command above.

- After observing the outputs please find the model of your PCI Ethernet Controller by using one of the commands above.



Find a way to display the environment variables



Find KDE partition manager, observe your disks partitions,
which one is the major partition?



Investigate these commands `id`, `last`, `w`, `who` and take notes to explain them.



Investigate the difference between info and man, drop me some lines about it. You may check “cp” command in both documentation and tell the difference.



File Permissions

File Permission



Ownership

User

- A user is the owner of the file.

Group

- A user- group can contain multiple users.

Other

- Any other user who has access to a file.

Permission

Read

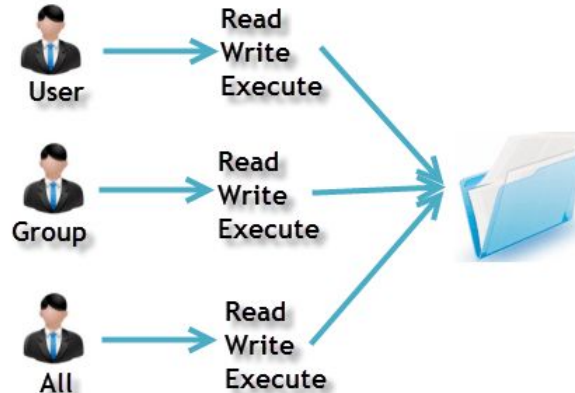
- This permission give you the authority to open and read a file.

Write

- The write permission gives you the authority to modify the contents of a file.

Execute

- you cannot run a program unless the execute permission is set.



File Permission



```
gakeko2018@DESKTOP-JA07K2U:~$ ls
cert.pem
gakeko2018@DESKTOP-JA07K2U:~$ ls -la
.  .. .bash_history .bash_logout .bashrc .local .profile .ssh cert.pem
gakeko2018@DESKTOP-JA07K2U:~$ ls -al
total 12
drwxr-xr-x 1 gakeko2018 gakeko2018 4096 Jan 13 09:41 .
drwxr-xr-x 1 root      root      4096 Dec 25 18:19 ..
-rw-r--r-- 1 gakeko2018 gakeko2018 236 Jan 14 12:21 .bash_history
-rw-r--r-- 1 gakeko2018 gakeko2018 220 Dec 25 18:19 .bash_logout
-rw-r--r-- 1 gakeko2018 gakeko2018 3771 Dec 25 18:19 .bashrc
drwxrwxrwx 1 gakeko2018 gakeko2018 4096 Jan 13 09:38 .local
-rw-r--r-- 1 gakeko2018 gakeko2018 807 Dec 25 18:19 .profile
drwx----- 1 gakeko2018 gakeko2018 4096 Jan 13 09:41 .ssh
-r----- 1 gakeko2018 gakeko2018 1675 Jan 13 09:38 cert.pem
```

File type and Access Permissions

`-rw-r--r-- 1 gakeko2018 gakeko2018 807 Dec 25 18:19 .profile`

indicates File

`drwxr-xr-x 1 gakeko2018 gakeko2018 4096 Jan 13 09:41 .`

d represents directory

User Group Others

r: Read
w: Write
x: Execute

-rw-rw-r--
no execute permission

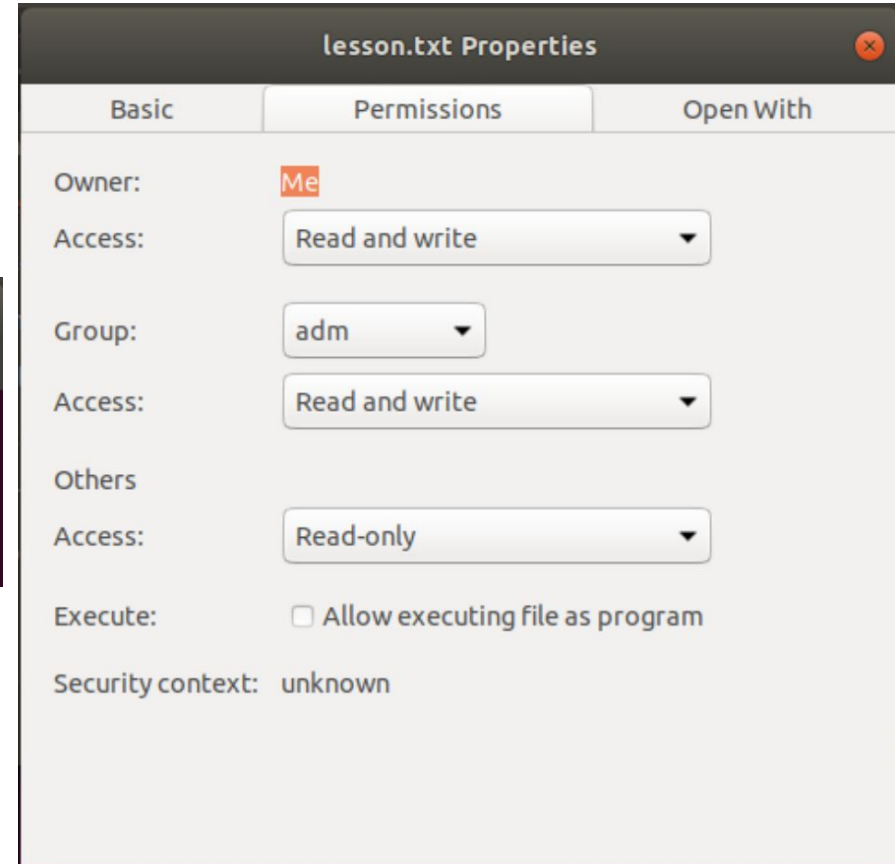
r = read permission
w = write permission
x = execute permission
- = no permission

File Permission



```
raymond@clarusway-linux: ~  
File Edit View Search Terminal Help  
raymond@clarusway-linux:~$ ls -l lesson.txt  
-rw-rw-r-- 1 raymond adm 8 Mar  2 21:19 lesson.txt  
raymond@clarusway-linux:~$
```

Please find data.json and investigate the permission details in UI and Terminal interface, make a comparison, does it match?





File Permission

Changing Permission with chmod Command

We can use the **chmod** command which stands for **change mode**.
we can set permissions (read, write, execute) on a file/directory for the owner, group and the world.

```
chmod permissions filename
```

```
chmod u=rwx,g=rx,o=r  
data.json
```

Symbol	Permission Type
---	No Permission
--x	Execute
-w-	Write
-wx	Execute+Write
r--	Read
r-x	Read+Execute
rw-	Read+Write
rwX	Read+Write+Execute

File Permission



```
root@DESKTOP-4QQ1S5L:~# ls -l
total 0
-rw-rw-rw- 1 root root 0 Dec 29 17:53 file1
-r--r--rwx 1 root root 0 Dec 29 17:53 file2
root@DESKTOP-4QQ1S5L:~# chmod 754 file2
root@DESKTOP-4QQ1S5L:~# ls -l file2
-rwxr-xr-- 1 root root 0 Dec 29 17:53 file2
root@DESKTOP-4QQ1S5L:~#
```

754 code says;

- Owner can read, write and execute
- User's group can read and execute
- Other can only read

Permissions

read=4;

rwX

4 +2 +1

7

write=2;

r-X

4 +0 +1

5

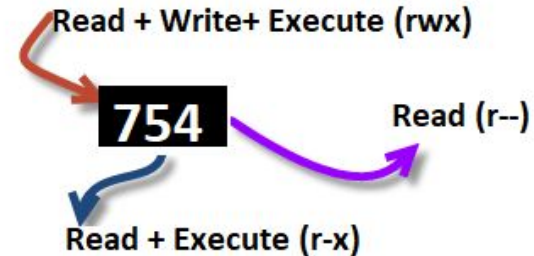
execute=1

r--

4 +0 +0

4

```
chmod u=rwx,g=rx,o=r myfile
chmod 754 myfile
```



File Permission



	Owner			Group			Other Users		
- or d	r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1	
7			7			7			

Read + Write + Execute (rwx)

764

Read (r--)

Read + Write (rw-)

d	r	w	x	r	-	x	r	-	-
	read	write	exec	read	write	exec	read	write	exec
File type	Owner permissions			Group permissions			User permissions		
(directory)	4	2	1	4	2	1	4	2	1
	7			5			4		



Set permissions of data.json to;

owner : full access

group : read and execute

others : no access

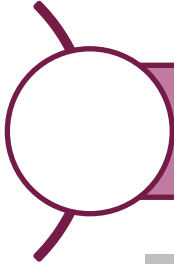


Students, write your response!



Ping & SSH Command

Ping Command



Ping or Packet Internet Groper is a network administration utility used to check the connectivity status between a source and a destination device.

ping host-name/IP

```
ping 54.93.34.220
```

```
gakeko2018@DESKTOP-JA07K2U:~$ ping 54.93.34.220
PING 54.93.34.220 (54.93.34.220) 56(84) bytes of data.
64 bytes from 54.93.34.220: icmp_seq=1 ttl=243 time=62.6 ms
64 bytes from 54.93.34.220: icmp_seq=2 ttl=243 time=93.5 ms
64 bytes from 54.93.34.220: icmp_seq=3 ttl=243 time=66.8 ms
64 bytes from 54.93.34.220: icmp_seq=4 ttl=243 time=67.6 ms
64 bytes from 54.93.34.220: icmp_seq=5 ttl=243 time=62.7 ms
64 bytes from 54.93.34.220: icmp_seq=7 ttl=243 time=84.6 ms
64 bytes from 54.93.34.220: icmp_seq=8 ttl=243 time=64.6 ms
64 bytes from 54.93.34.220: icmp_seq=9 ttl=243 time=72.0 ms
```

Ping Command



```
$ ping clarusway.com

Pinging clarusway.com [54.164.151.235] with 32 bytes of data:
Reply from 54.164.151.235: bytes=32 time=132ms TTL=237
Reply from 54.164.151.235: bytes=32 time=130ms TTL=237
Reply from 54.164.151.235: bytes=32 time=130ms TTL=237
Reply from 54.164.151.235: bytes=32 time=130ms TTL=237

Ping statistics for 54.164.151.235:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 130ms, Maximum = 132ms, Average = 130ms
```

```
$ ping www.google.com

Pinging www.google.com [172.217.169.132] with 32 bytes of data:
Reply from 172.217.169.132: bytes=32 time=19ms TTL=116
Reply from 172.217.169.132: bytes=32 time=18ms TTL=116
Reply from 172.217.169.132: bytes=32 time=18ms TTL=116
Reply from 172.217.169.132: bytes=32 time=19ms TTL=116

Ping statistics for 172.217.169.132:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 18ms, Maximum = 19ms, Average = 18ms
```



SSH Command

- * ssh stands for “Secure Shell”.
- * It is a protocol used to securely connect to a remote server/system.

`ssh user@host(IP/Domain_name)`

```
ssh -i cert.pem ec2-user@54.93.34.220
```

```
gakeko2018@DESKTOP-JA07K2U:~$ ssh -i cert.pem ec2-user@54.93.34.220
The authenticity of host '54.93.34.220 (54.93.34.220)' can't be established.
ECDSA key fingerprint is SHA256:lvCnUtJiig4s2U4aojBonZOSbzGPBMOpB9yPPoGjVEo.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '54.93.34.220' (ECDSA) to the list of known hosts.

 _ | _ | _ |
_| ( _ | _ | /   Amazon Linux 2 AMI
_| \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 13 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-35-15 ~]$
```



Ping Command

```
$ ping 54.164.151.235
```

```
Pinging 54.164.151.235 with 32 bytes of data:
```

```
Reply from 54.164.151.235: bytes=32 time=131ms TTL=237
```

```
Reply from 54.164.151.235: bytes=32 time=130ms TTL=237
```

```
Reply from 54.164.151.235: bytes=32 time=130ms TTL=237
```

```
Reply from 54.164.151.235: bytes=32 time=130ms TTL=237
```

```
Ping statistics for 54.164.151.235:
```

```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
```

```
    Minimum = 130ms, Maximum = 131ms, Average = 130ms
```



THANK YOU

Any questions?