

1. Setup VM, Linux, and basic testing – must take screen shots at each step to receive points.
(forgot to take screenshots but below some of the commands that were used along the way)

h. Turn on Firewall and block all ports:

```
sudo ufw enable
```

```
sudo ufw default deny
```

i. Enable SSH access to your new Linux installation; open SSH port in firewall

```
sudo apt update
```

```
sudo apt install openssh-server
```

To verify:

```
sudo systemctl status ssh
```

```
sudo ufw allow ssh
```

2. Show an example of using the following commands (hint: you can use man to find more information about each one); take screen shots of your commands; make sure to clear the screen between each command; explain in your own words what these commands do:

a. **ssh**

Starts a session on a remote machine.

```
C:\Users\barba>ssh -i ~/.ssh/id_rsa varvara@192.168.128.34
The authenticity of host '192.168.128.34 (192.168.128.34)' can't be established.
ECDSA key fingerprint is SHA256:xPBtPKF+EWxOxF8VhQxF8FeoIQFYtwFULjD9+pyX+QglQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.128.34' (ECDSA) to the list of known hosts.
varvara@192.168.128.34's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

12 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```


b. **ssh-keygen**

Generates a public/private key pair and saves it to a specified folder.

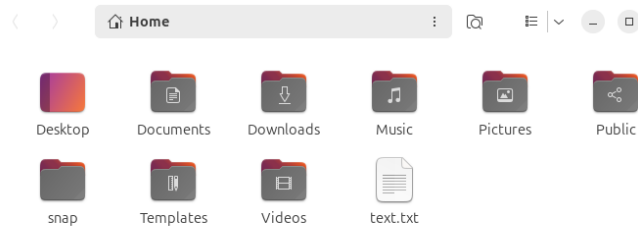
```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ ssh-keygen  
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home/varvara/.ssh/id_ed25519):  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/varvara/.ssh/id_ed25519  
Your public key has been saved in /home/varvara/.ssh/id_ed25519.pub  
The key fingerprint is:  
SHA256:pxK44vXcRuI3PGUSevp3k19UwtcMUV3UFTYcdWcHwnc varvara@varvara-VirtualBox  
The key's randomart image is:  
---[ED25519 256]---+  
      .. =X^|  
      .oo=E|  
      .o.=|  
      .   o.|  
      . ..S.. .|  
      .o.+oo .|  
      . o..*+. .|  
      . o ooo* . + .|  
      .   o+o+ . o. |  
-----[SHA256]-----+  
varvara@varvara-VirtualBox:~$
```

c. **scp**

securely copies a file from a local machine to a remote machine and vice versa.

 Select Command Prompt

```
C:\Users\barba\Desktop>scp text.txt varvara@192.168.138.34  
1 file(s) copied.  
  
C:\Users\barba\Desktop>
```



d. **history**

displays a list of commands used in the terminal session.

```

varvara@varvara-VirtualBox:~$ history
 1 sudo systemctl status ssh
 2 sudo ufw allow ssh
 3 ip a
 4 cat .ssh/authorized_keys
 5 nano .ssh/authorized_keys
 6 exit
 7 touch text.tx
 8 ds
 9 ls
10 rm text.tx
11 ls
12 touch text.txt
13 clear
14 ls
15 vi text.txt
16 clear
17 ip a
18 cd .ssh
19 ls
20 cd ..
21 ssh -i ~/.ssh/id_ed25519 barba@192.168.128.76
22 vim text.txt
23 clear
24 vim text.txt
25 sudo apt install vim
26 vim text.txt
27 clear
28 ls
29 cd Desktop
30 ls
31 cd ..

```

e. **sudo**

runs any command with elevated privileges, which allow to perform administrative tasks.

```

varvara@varvara-VirtualBox:~$ sudo apt install vim
[sudo] password for varvara:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libsodium23 vim-common vim-runtime vim-tiny xxd
Suggested packages:
  ctags vim-doc vim-scripts indent
The following NEW packages will be installed:
  libsodium23 vim vim-runtime
The following packages will be upgraded:
  vim-common vim-tiny xxd
3 upgraded, 3 newly installed, 0 to remove and 14 not upgraded.
Need to get 10.6 MB of archives.
After this operation, 42.0 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim-tiny amd64 2:9.1.0016-1ubuntu7.1 [803 B]
Get:2 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim-common all 2:9.1.0016-1ubuntu7.1 [385 B]
Get:3 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 xxd amd64 2:9.1.0016-1ubuntu7.1 [62.9 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu noble/main amd64 libsodium23 amd64 1.0.18-1build3 [161 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim-runtime all 2:9.1.0016-1ubuntu7.1 [7,214 B]
Get:6 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim amd64 2:9.1.0016-1ubuntu7.1 [1,881 kB]
Fetched 10.6 MB in 1s (8,142 kB/s)
(Reading database ... 150953 files and directories currently installed.)
Preparing to unpack .../0-vim-tiny_2%3a9.1.0016-1ubuntu7.1_amd64.deb ...
Unpacking vim-tiny (2:9.1.0016-1ubuntu7.1) over (2:9.1.0016-1ubuntu7) ...
Preparing to unpack .../1-vim-common_2%3a9.1.0016-1ubuntu7.1_all.deb ...
Unpacking vim-common (2:9.1.0016-1ubuntu7.1) over (2:9.1.0016-1ubuntu7) ...
Preparing to unpack .../2-xxd_2%3a9.1.0016-1ubuntu7.1_amd64.deb ...
Unpacking xxd (2:9.1.0016-1ubuntu7.1) over (2:9.1.0016-1ubuntu7) ...
Setting up vim-tiny (2:9.1.0016-1ubuntu7.1) ...
Setting up vim-common (2:9.1.0016-1ubuntu7.1) ...
Setting up xxd (2:9.1.0016-1ubuntu7.1) ...
Setting up vim-runtime (2:9.1.0016-1ubuntu7.1) ...
Setting up vim (2:9.1.0016-1ubuntu7.1) ...

```

f. **ip**

shows the ip address of the machine.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host noprefixroute  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP  
    group default qlen 1000  
    link/ether 08:00:27:67:d0:a2 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.128.34/24 brd 192.168.128.255 scope global dynamic noprefixroute  
        valid_lft 42855sec preferred_lft 42855sec  
    inet6 fe80::a00:27ff:fe67:d0a2/64 scope link  
        valid_lft forever preferred_lft forever  
varvara@varvara-VirtualBox:~$
```

g. **dd**

reads input from a file or device and writes it to another file or device.

```
varvara@varvara-VirtualBox:~$ dd if=text.txt of=another.txt  
0+1 records in  
0+1 records out  
38 bytes copied, 0.00329331 s, 11.5 kB/s  
varvara@varvara-VirtualBox:~$
```

h. **fdisk**

used for creating and manipulating disk partition table.

- Creating space for new partitions
- Organizing space for new drives
- Re-organizing old drives
- Moving data to new disks

```

varvara@varvara-VirtualBox:~$ sudo fdisk -l
[sudo] password for varvara:
Disk /dev/loop0: 4 KiB, 4096 bytes, 8 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop1: 74.24 MiB, 77844480 bytes, 152040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop2: 269.63 MiB, 282722304 bytes, 552192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop3: 10.72 MiB, 11239424 bytes, 21952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop4: 91.69 MiB, 96141312 bytes, 187776 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

```

i. apt

tool for managing deb packages on Linux distributions. Used to install, update, remove different applications.

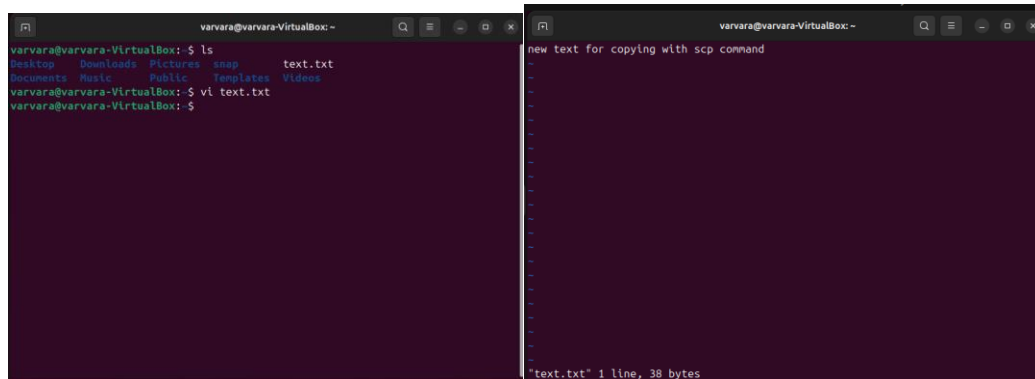
```

varvara@varvara-VirtualBox:~$ sudo apt install vim
[sudo] password for varvara:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libsodium23 vim-common vim-runtime vim-tiny xxd
Suggested packages:
  ctags vim-doc vim-scripts indent
The following NEW packages will be installed:
  libsodium23 vim vim-runtime
The following packages will be upgraded:
  vim-common vim-tiny xxd
3 upgraded, 3 newly installed, 0 to remove and 14 not upgraded.
Need to get 10.6 MB of archives.
After this operation, 42.0 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim-tiny amd64 2:9.1.0016-1ubuntu7.1 [803 B]
Get:2 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim-common all 2:9.1.0016-1ubuntu7.1 [385 B]
Get:3 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 xxd amd64 2:9.1.0016-1ubuntu7.1 [62.9 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu noble/main amd64 libsodium23 amd64 1.0.18-1build3 [161 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim-runtime all 2:9.1.0016-1ubuntu7.1 [7,214 B]
Get:6 http://us.archive.ubuntu.com/ubuntu noble-updates/main amd64 vim amd64 2:9.1.0016-1ubuntu7.1 [1,881 kB]
Fetched 10.6 MB in 1s (8,142 kB/s)
(Reading database ... 150953 files and directories currently installed.)
Preparing to unpack .../0-vim-tiny_2%3a9.1.0016-1ubuntu7.1_amd64.deb ...
Unpacking vim-tiny (2:9.1.0016-1ubuntu7.1) over (2:9.1.0016-1ubuntu7) ...
Preparing to unpack .../1-vim-common_2%3a9.1.0016-1ubuntu7.1_all.deb ...
Unpacking vim-common (2:9.1.0016-1ubuntu7.1) over (2:9.1.0016-1ubuntu7) ...
Preparing to unpack .../2-vim_2%3a9.1.0016-1ubuntu7.1_amd64.deb ...
Unpacking vim (2:9.1.0016-1ubuntu7.1) over (2:9.1.0016-1ubuntu7) ...
Setting up vim-tiny (2:9.1.0016-1ubuntu7.1) ...
Setting up vim-common (2:9.1.0016-1ubuntu7.1) ...
Setting up vim (2:9.1.0016-1ubuntu7.1) ...

```

j. vi

opens a text editor Vi on a file specified.



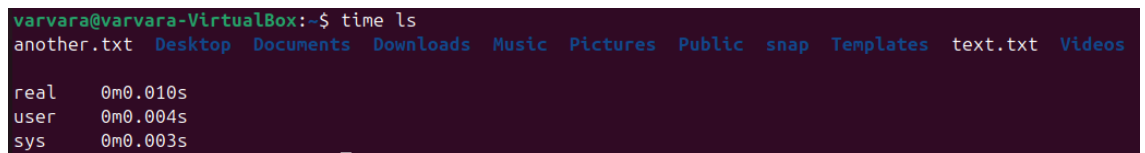
The image shows two terminal windows. The left window shows the execution of `ls` and `vi text.txt`. The right window shows the text editor interface with the text "new text for copying with scp command" and a status bar indicating "1 line, 38 bytes".

```
varvara@varvara-VirtualBox:~$ ls
Desktop  Downloads  Pictures  snap      text.txt
Documents Music      Public    Templates Videos
varvara@varvara-VirtualBox:~$ vi text.txt
varvara@varvara-VirtualBox:~$
```

```
new text for copying with scp command
*text.txt* 1 line, 38 bytes
```

k. **time**

measures how long it takes for a command/program to execute.



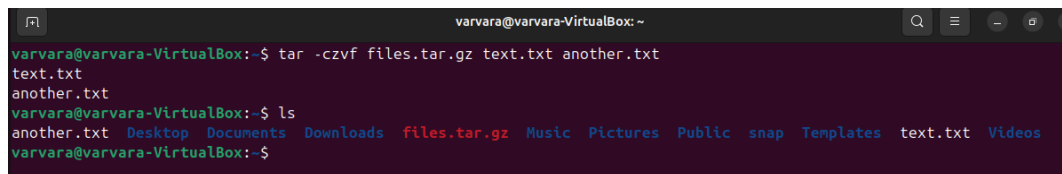
The terminal window shows the `time ls` command being executed, followed by a list of files and the execution time breakdown.

```
varvara@varvara-VirtualBox:~$ time ls
another.txt  Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  text.txt  Videos

real    0m0.010s
user    0m0.004s
sys     0m0.003s
```

l. **tar**

command used for creating, viewing, extracting files from archives.

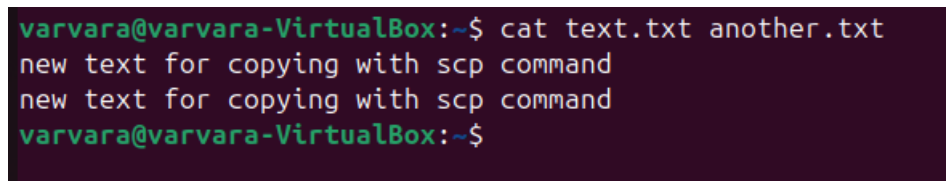


The terminal window shows the `tar -czvf files.tar.gz text.txt another.txt` command being executed, followed by `ls` to verify the archive creation.

```
varvara@varvara-VirtualBox:~$ tar -czvf files.tar.gz text.txt another.txt
text.txt
another.txt
varvara@varvara-VirtualBox:~$ ls
another.txt  Desktop  Documents  Downloads  files.tar.gz  Music  Pictures  Public  snap  Templates  text.txt  Videos
varvara@varvara-VirtualBox:~$
```

m. **cat**

reads files sequentially, displaying the content to the terminal.



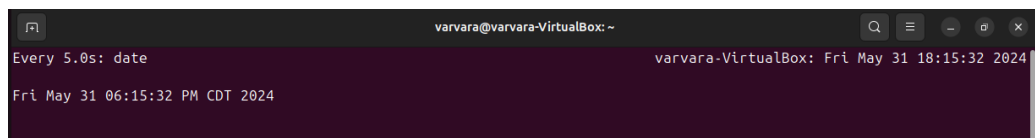
The terminal window shows the `cat text.txt another.txt` command being executed, displaying the contents of both files.

```
varvara@varvara-VirtualBox:~$ cat text.txt another.txt
new text for copying with scp command
new text for copying with scp command
varvara@varvara-VirtualBox:~$
```

n. **watch**

runs user-defined commands at regular intervals.

`watch -n 5 date` (below)



The terminal window shows the `watch -n 5 date` command being executed, displaying the date every 5 seconds.

```
varvara@varvara-VirtualBox:~$ watch -n 5 date
Every 5.0s: date
varvara-VirtualBox: Fri May 31 18:15:32 2024
Fri May 31 06:15:32 PM CDT 2024
```

o. **ps**

displays a list of all current processes running on the system. Regular `ps` will display info about processes related to the current user. Flag `-e` expands the functionality of it to listing info about processes of all users.

```
varvara@varvara-VirtualBox:~$ ps
  PID TTY          TIME CMD
 2735 pts/0    00:00:00 bash
 5342 pts/0    00:00:00 ps
varvara@varvara-VirtualBox:~$ ps -e
```

p. `top`

provides dynamic real-time view of the running system. Processes are sorted by CPU usage in descending order by default.

```
top - 18:26:19 up 3:50, 1 user, load average: 0.07, 0.04, 0.00
Tasks: 222 total, 1 running, 221 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.1 us, 0.1 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.1 st, 0.0 st
MiB Mem : 3915.4 total, 2062.7 free, 1050.9 used, 1060.1 buff/cache
MiB Swap: 3820.0 total, 3820.0 free, 0.0 used, 2864.6 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM     TIME+ COMMAND
 1950 varvara   20   0 5006112 393072 138316 S    2.0   9.8   3:44.18 gnome-shell
   1 root       20   0 23104 13536  9184 S    0.0   0.3   0:02.93 systemd
   2 root       20   0      0      0   0 S    0.0   0.0   0:00.06 kthreadd
   3 root       20   0      0      0   0 S    0.0   0.0   0:00.00 pool_workqueue_release
   4 root       0 -20      0      0   0 I    0.0   0.0   0:00.00 kworker/R-rcu_g
   5 root       0 -20      0      0   0 I    0.0   0.0   0:00.00 kworker/R-rcu_p
   6 root       0 -20      0      0   0 I    0.0   0.0   0:00.00 kworker/R-slub
   7 root       0 -20      0      0   0 I    0.0   0.0   0:00.00 kworker/R-netns
   8 root       20   0      0      0   0 I    0.0   0.0   0:13.73 kworker/0:0-cgroup_destroy
  10 root       0 -20      0      0   0 I    0.0   0.0   0:00.00 kworker/0:0H-events_highpri
  11 root       20   0      0      0   0 I    0.0   0.0   0:00.00 kworker/u8:0-ext4-rsv-conversion
  12 root       0 -20      0      0   0 I    0.0   0.0   0:00.00 kworker/R-mm_pe
  13 root       20   0      0      0   0 I    0.0   0.0   0:00.00 rcu_tasks_kthread
  14 root       20   0      0      0   0 I    0.0   0.0   0:00.00 rcu_tasks_rude_kthread
  15 root       20   0      0      0   0 I    0.0   0.0   0:00.00 rcu_tasks_trace_kthread
  16 root       20   0      0      0   0 S    0.0   0.0   0:02.82 ksoftirqd/0
  17 root       20   0      0      0   0 I    0.0   0.0   0:04.32 rcu_preempt
  18 root       rt   0      0      0   0 S    0.0   0.0   0:00.27 migration/0
  19 root      -51   0      0      0   0 S    0.0   0.0   0:00.00 idle_inject/0
  20 root       20   0      0      0   0 S    0.0   0.0   0:00.00 cpuhp/0
  21 root       20   0      0      0   0 S    0.0   0.0   0:00.00 cpuhp/1
  22 root      -51   0      0      0   0 S    0.0   0.0   0:00.00 idle_inject/1
  23 root       rt   0      0      0   0 S    0.0   0.0   0:01.15 migration/1
  24 root       20   0      0      0   0 S    0.0   0.0   0:00.22 ksoftirqd/1
  27 root       20   0      0      0   0 S    0.0   0.0   0:00.00 cpuhp/2
```

q. `htop`

dynamic process viewer like `top` but allows to scroll both vertically and horizontally and provides with an option to kill processes or view them in a tree.

```
varvara@varvara-VirtualBox: ~
0[ ] 3.5% Tasks: 111, 340 thr, 123 kthr; 2 running
1[ ] 4.0% Load average: 0.35 0.68 0.33
2[ ] 0.7% Uptime: 00:03:22
3[ ] 4.7%
Mem[ ] 773M/3.82G
Swp[ ] 0K/3.73G

Main I/O
PID USER      PRI  NI  VIRT  RES  SHR  S  %CPU  %MEM     TIME+ Command
1952 varvara   20   0 4863M 352M 131M S    4.8   9.0   0:08.23 /usr/bin/gnome-shell
1986 varvara  -21   0 4863M 352M 131M S    2.7   9.0   0:02.42 /usr/bin/gnome-shell
2676 varvara   20   0 544M 54664 43096 S    1.4   1.4   0:02.29 /usr/libexec/gnome-te
1994 varvara   20   0 4863M 352M 131M S    0.7   9.0   0:02.85 /usr/bin/gnome-shell
2985 varvara   20   0 11288 4736 3456 R    2.0   0.1   0:00.32 htop
1792 varvara  -21   0 112M 14076 8956 S    0.0   0.4   0:00.36 /usr/bin/pipewire
1996 varvara   20   0 4863M 352M 131M S    0.7   9.0   0:02.71 /usr/bin/gnome-shell
1997 varvara   20   0 4863M 352M 131M S    0.0   9.0   0:03.00 /usr/bin/gnome-shell
2007 varvara   20   0 4863M 352M 131M S    2.0   9.0   0:01.32 /usr/bin/gnome-shell
   1 root       20   0 2:096 13784 9304 S    0.0   0.3   0:01.43 /sbin/init splash
 283 root       20   0 4:640 16768 15488 S    0.4   0.4   0:00.35 /usr/lib/systemd/syst
 345 root       20   0 30416 8192 4864 S    0.0   0.2   0:00.27 /usr/lib/systemd/syst
 395 systemd-oo 20   0 17556 7552 6656 S    0.0   0.2   0:00.13 /usr/lib/systemd/syst
 400 systemd-re 20   0 21580 13056 10880 S    0.0   0.3   0:00.11 /usr/lib/systemd/syst
 402 systemd-tl 20   0 91044 7808 6912 S    0.0   0.2   0:00.05 /usr/lib/systemd/syst
 456 systemd-tl 20   0 91044 7808 6912 S    0.0   0.2   0:00.00 /usr/lib/systemd/syst
 644 root       20   0 8288 2176 2048 S    0.0   0.1   0:00.00 /usr/sbin/anacron -d
 661 avahi      20   0 8668 4480 4096 S    0.0   0.1   0:00.11 avahi-daemon: running

F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8NICE F9Kill F10Quit
```

r. `gcc`

C/C++ compiler. Normally does preprocessing, compilation, assembly, and linking. Additional options allow to stop the compiler process at one of the above steps.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ ls  
another.txt  Documents  files.tar.gz  Pictures  Public  Templates  Videos  
Desktop      Downloads  Music         program.c  snap    text.txt  
varvara@varvara-VirtualBox:~$ gcc program.c  
varvara@varvara-VirtualBox:~$ ls  
another.txt  Desktop  Downloads  Music  program.c  snap  text.txt  
a.out        Documents  files.tar.gz  Pictures  Public  Templates  Videos  
varvara@varvara-VirtualBox:~$ a.out  
a.out: command not found  
varvara@varvara-VirtualBox:~$ ./a.out
```

s. **tail**

prints the last lines of a file. By default, 10 lines

```
varvara@varvara-VirtualBox:~$ tail program.c  
#include <stdio.h>  
int main() {  
    printf("Hello world!");  
    return 0;  
}
```

t. **grep**

allows to search for specified words in text.

```
varvara@varvara-VirtualBox:~$ grep "Linux" another.txt  
Welcome to Linux!  
Linux is a free and open source Operating system that is mostly used by develop  
ers and in production servers for hosting crucial components such as web and data  
base servers. Linux has also made a name for itself in PCs.  
varvara@varvara-VirtualBox:~$
```

u. **kill**

sends a signal to a process to terminate it.

```
varvara@varvara-VirtualBox:~$ top &  
[1] 4030  
varvara@varvara-VirtualBox:~$ kill 4030  
  
[1]+  Stopped                  top  
varvara@varvara-VirtualBox:~$
```

v. **killall**

terminates all instances of a specific process. Below command closed opened firefox browser.

```
varvara@varvara-VirtualBox:~$ killall firefox  
varvara@varvara-VirtualBox:~$
```

w. **du**

allows to analyze and report disk usage within directories and files.

- Identify space-hogging directories.
- Manage disk space efficiently.
- Gain insights into storage consumption.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ du  
16      ./ssh  
4       ./Public  
4       ./Downloads  
8       ./local/state/wireplumber  
12      ./local/state  
4       ./local/share/sounds  
4       ./local/share/ibus-table  
4       ./local/share/applications  
12      ./local/share/keyrings  
76      ./local/share/gvfs-metadata  
4       ./local/share/nano  
4       ./local/share/flatpak/db  
8       ./local/share/flatpak  
4       ./local/share/icc  
8       ./local/share/gnome-shell  
4       ./local/share/evolution/addressbook/system/photos  
100     ./local/share/evolution/addressbook/system  
4       ./local/share/evolution/addressbook/trash  
108     ./local/share/evolution/addressbook  
8       ./local/share/evolution/tasks/system  
4       ./local/share/evolution/tasks/trash  
16      ./local/share/evolution/tasks  
4       ./local/share/evolution/memos/trash
```

x. **df**

displays free disk space.

```
varvara@varvara-VirtualBox:~$ df  
Filesystem      1K-blocks    Used Available Use% Mounted on  
tmpfs           400944      1600    399344   1% /run  
/dev/sda2       25623780 9467904  14828928 39% /  
tmpfs           2004716        0    2004716   0% /dev/shm  
tmpfs            5120         8        5112   1% /run/lock  
tmpfs           400940     128    400812   1% /run/user/1000  
varvara@varvara-VirtualBox:~$
```

y. **screen**

```
varvara@varvara-VirtualBox: ~
GNU Screen version 4.09.01 (GNU) 20-Aug-23

Copyright (c) 2018-2023 Alexander Naumov, Amadeusz Slawinski
Copyright (c) 2015-2017 Juergen Weigert, Alexander Naumov, Amadeusz Slawinski
Copyright (c) 2010-2014 Juergen Weigert, Sadrul Habib Chowdhury
Copyright (c) 2008-2009 Juergen Weigert, Michael Schroeder, Micah Cowan,
Sadrul Habib Chowdhury
Copyright (c) 1993-2007 Juergen Weigert, Michael Schroeder
Copyright (c) 1987 Oliver Laumann

This program is free software; you can redistribute it and/or modify it under
the terms of the GNU General Public License as published by the Free Software
Foundation; either version 3, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with
this program (see the file COPYING); if not, see

[Press Space for next page; Return to end.]
```

```
varvara@varvara-VirtualBox:~$ vim text.txt
varvara@varvara-VirtualBox:~$
```



allows to change the access mode of a file. chmod stands for “change mode”.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ ./a.out  
Hello world!  
varvara@varvara-VirtualBox:~$ ls  
another.txt Desktop Downloads Music program.c snap text.txt  
a.out Documents files.tar.gz Pictures Public Templates Videos  
varvara@varvara-VirtualBox:~$ chmod u-x a.out  
varvara@varvara-VirtualBox:~$ ls  
another.txt Desktop Downloads Music program.c snap text.txt  
a.out Documents files.tar.gz Pictures Public Templates Videos  
varvara@varvara-VirtualBox:~$ ./a.out  
bash: ./a.out: Permission denied  
varvara@varvara-VirtualBox:~$
```

bb. chown

allows to change file owner and group. chown stand for “change owner”.

```
varvara@varvara-VirtualBox:~  
varvara@varvara-VirtualBox:~$ ls  
another.txt Desktop Downloads Music program.c snap text.txt  
a.out Documents files.tar.gz Pictures Public Templates Videos  
varvara@varvara-VirtualBox:~$ ls -l text.txt  
-rw-rw-r-- 1 varvara varvara 38 May 31 14:47 text.txt  
varvara@varvara-VirtualBox:~$ chown root text.txt  
chown: changing ownership of 'text.txt': Operation not permitted  
varvara@varvara-VirtualBox:~$ sudo chown root text.txt  
[sudo] password for varvara:  
varvara@varvara-VirtualBox:~$ ls -l text.txt  
-rw-rw-r-- 1 root varvara 38 May 31 14:47 text.txt  
varvara@varvara-VirtualBox:~$
```

cc. useradd

adds new user accounts to the system.

```
varvara@varvara-VirtualBox:~  
varvara@varvara-VirtualBox:~$ sudo useradd newuser  
varvara@varvara-VirtualBox:~$ cat /etc/passwd | grep newuser  
newuser:x:1001:1001::/home/newuser:/bin/sh  
varvara@varvara-VirtualBox:~$
```

dd. man

displays the user manual of commands.

```
CHMOD(1) User Commands CHMOD(1)  
  
NAME  
chmod - change file mode bits  
  
SYNOPSIS  
chmod [OPTION]... MODE[.MODE]... FILE...  
chmod [OPTION]... OCTAL-MODE FILE...  
chmod [OPTION]... --reference=REFILE FILE...  
  
DESCRIPTION  
This manual page documents the GNU version of chmod. chmod changes the  
file mode bits of each given file according to mode, which can be ei-  
ther a symbolic representation of changes to make, or an octal number  
representing the bit pattern for the new mode bits.  
  
The format of a symbolic mode is [ugoa...][[-+=][perms...]]..., where  
perms is either zero or more letters from the set rwXst, or a single  
letter from the set ugo. Multiple symbolic modes can be given, sepa-  
rated by commas.  
  
A combination of the letters ugoa controls which users' access to the  
file will be changed: the user who owns it (u), other users in the  
Manual page chmod(1) line 1 (press h for help or q to quit)
```

ee. locate

allows to find files by their names.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ locate file.txt  
/home/varvara/Desktop/file.txt  
/usr/share/doc/alsa-base/driver/Procfile.txt.gz  
varvara@varvara-VirtualBox:~$
```

ff. **find**

dynamic utility used for file search within a hierarchical structure.

```
varvara@varvara-VirtualBox: ~/Desktop  
varvara@varvara-VirtualBox:~$ find file.txt  
find: 'file.txt': No such file or directory  
varvara@varvara-VirtualBox:~$ cd Desktop  
varvara@varvara-VirtualBox:~/Desktop$ find file.txt  
file.txt  
varvara@varvara-VirtualBox:~/Desktop$
```

gg. **sed**

stream editor. Can perform various functions on files like searching, finding, replacing, inserting, or deleting.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ cat another.txt  
Welcome to Linux!  
Linux is a free and open source Operating system that is mostly used by develop  
rs and in production servers for hosting crucial components such as web and data  
base servers. Linux has also made a name for itself in PCs.  
varvara@varvara-VirtualBox:~$ sed 's/Linux/unix' another.txt  
sed: -e expression #1, char 12: unterminated 's' command  
varvara@varvara-VirtualBox:~$ sed 's/Linux/unix/' another.txt  
Welcome to unix!  
unix is a free and open source Operating system that is mostly used by develop  
s and in production servers for hosting crucial components such as web and datab  
ase servers. Linux has also made a name for itself in PCs.  
varvara@varvara-VirtualBox:~$
```

hh. **awk**

scripting language used for manipulating data and generating reports.

```
varvara@varvara-VirtualBox:~$ cat lab.txt  
jane doe professor  
john doe phd_student  
emily doe student  
johnson doe student  
varvara@varvara-VirtualBox:~$ awk '/student/ {print}' lab.txt  
john doe phd_student  
emily doe student  
johnson doe student
```

ii. **diff**

helps compare files. Useful for debugging. Stands for 'difference'.

```
varvara@varvara-VirtualBox:~$ diff lab.txt lab2.txt
1,4c1,4
< jane doe professor
< john doe phd_student
< emily doe student
< johnson doe student
---
> john smith professor
> katy smith postdoc
> ana smith student
> ross smith student
varvara@varvara-VirtualBox:~$
```

jj. **sort**

sorts the context of a text file line by line, according to ASCII. Doesn't modify the original file by default.

```
varvara@varvara-VirtualBox:~$ cat lab.txt
jane doe professor
john doe phd_student
emily doe student
johnson doe student
varvara@varvara-VirtualBox:~$ sort lab.txt
emily doe student
jane doe professor
john doe phd_student
johnson doe student
varvara@varvara-VirtualBox:~$
```

kk. **export**

- shows all the exported variables.
- Marks environment variables to be exported to child processes.

```
varvara@varvara-VirtualBox:~$ export
declare -x COLORTERM="truecolor"
declare -x DBUS_SESSION_BUS_ADDRESS="unix:path=/run/user/1000/bus"
declare -x DEBUGINFOD_URLS="https://debuginfod.ubuntu.com "
declare -x DESKTOP_SESSION="ubuntu"
declare -x DISPLAY=":0"
declare -x GDMSESSION="ubuntu"
declare -x GNOME_DESKTOP_SESSION_ID="this-is-deprecated"
declare -x GNOME_SETUP_DISPLAY=":1"
declare -x GNOME_SHELL_SESSION_MODE="ubuntu"
declare -x GNOME_TERMINAL_SCREEN="/org/gnome/Terminal/screen/8ab982f8_3324_4f
bd4e_bbc3a03694e"
declare -x GNOME_TERMINAL_SERVICE=":1.112"
declare -x GSM_SKIP_SSH_AGENT_WORKAROUND="true"
declare -x GTK_MODULES="gail:atk-bridge"
declare -x HOME="/home/varvara"
declare -x IM_CONFIG_PHASE="1"
declare -x LANG="en_US.UTF-8"
declare -x LESSCLOSE="/usr/bin/lesspipe %s %s"
declare -x LESSOPEN="| /usr/bin/lesspipe %s"
declare -x LOGNAME="varvara"
declare -x LS_COLORS="rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35
=40;33;01:cd=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=00:tw=30;42:ow=3
2:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;
```

ll. **pwd**

stands for 'print working directory'. Prints the path of the working directory, starting from the root.

```
varvara@varvara-VirtualBox:~$ pwd
/home/varvara
varvara@varvara-VirtualBox:~$ cd Desktop/
varvara@varvara-VirtualBox:~/Desktop$ pwd
/home/varvara/Desktop
varvara@varvara-VirtualBox:~/Desktop$
```

mm. **crontab**

- lists commands that run on a regular schedule.
- Allows to manage the list of those commands.

```
varvara@varvara-VirtualBox:~/Desktop$ crontab -l
no crontab for varvara
varvara@varvara-VirtualBox:~/Desktop$
```

nn. **mount**

used to mount the filesystem found on a device to big tree structure (Linux filesystem) rooted at '/'.

```
varvara@varvara-VirtualBox:~$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=1969100k,nr_inodes=492275,node=755,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=400944k,node=755,inode64)
/dev/sda2 on / type ext4 (rw,relatime)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,inode64)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=32,pgrp=1,timeout=0,noinitproto=5,maxproto=5,direct,pipe_ino=5340)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,nosuid,nodev,relatime,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
/var/lib/snapd/snaps/bare_5.snap on /snap/bare/5 type squashfs (ro,nodev,relatime,errors=continue,threads=single,x-gdu.hide,x-gvfs-hide)
/var/lib/snapd/snaps/core22_1380.snap on /snap/core22/1380 type squashfs (ro,nodev,relatime,errors=continue,threads=single,x-gdu.hide,x-gvfs-hide)
/var/lib/snapd/snaps/firefox_4173.snap on /snap/firefox/4173 type squashfs (ro,nodev,relatime,errors=continue,threads=single,x-gdu.hide,x-gvfs-hide)
```

oo. **passwd**

used to change user passwords.

```
varvara@varvara-VirtualBox:~$ passwd
Changing password for varvara.
Current password:
New password:
Retype new password:
passwd: password updated successfully
varvara@varvara-VirtualBox:~$
```

pp. **uname**

displays system information.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ uname -a  
Linux varvara-VirtualBox 6.8.0-31-generic #31-Ubuntu SMP PREEMPT_DYNAMIC Sat Apr 20 00:40:06 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux  
varvara@varvara-VirtualBox:~$
```

qq. **whereis**

shows the location of the file.

```
varvara@varvara-VirtualBox: ~  
varvara@varvara-VirtualBox:~$ whereis file.txt  
file.txt:  
varvara@varvara-VirtualBox:~$
```

rr. **whatis**

explains what program's functionality is.

```
varvara@varvara-VirtualBox:~$ whatis ls  
ls (1) - list directory contents  
varvara@varvara-VirtualBox:~$
```

ss. **su**

used to switch to another user. 'switch user'.

Running without specifying username turns current user into super user.

```
varvara@varvara-VirtualBox:~$ su  
Password:  
su: Authentication failure
```

tt. **ping**

checks network connectivity.

```
varvara@varvara-VirtualBox:~$ hostname -I  
192.168.0.170  
varvara@varvara-VirtualBox:~$ ping 192.168.0.170  
PING 192.168.0.170 (192.168.0.170) 56(84) bytes of data.  
64 bytes from 192.168.0.170: icmp_seq=1 ttl=64 time=0.039 ms  
64 bytes from 192.168.0.170: icmp_seq=2 ttl=64 time=0.031 ms  
64 bytes from 192.168.0.170: icmp_seq=3 ttl=64 time=0.048 ms  
64 bytes from 192.168.0.170: icmp_seq=4 ttl=64 time=0.087 ms  
64 bytes from 192.168.0.170: icmp_seq=5 ttl=64 time=0.084 ms  
64 bytes from 192.168.0.170: icmp_seq=6 ttl=64 time=0.050 ms  
64 bytes from 192.168.0.170: icmp_seq=7 ttl=64 time=0.048 ms  
64 bytes from 192.168.0.170: icmp_seq=8 ttl=64 time=0.047 ms  
64 bytes from 192.168.0.170: icmp_seq=9 ttl=64 time=0.029 ms  
64 bytes from 192.168.0.170: icmp_seq=10 ttl=64 time=0.093 ms  
^C  
--- 192.168.0.170 ping statistics ---  
10 packets transmitted, 10 received, 0% packet loss, time 10841ms  
rtt min/avg/max/mdev = 0.029/0.055/0.093/0.022 ms
```

uu. **traceroute**

network diagnostic tool used to trace the route taken by packets from a source to a destination over an IP network.

```
varvara@varvara-VirtualBox:~$ traceroute google.com
traceroute to google.com (172.217.1.110), 30 hops max, 60 byte packets
 1  ArcherA7v5 (192.168.0.1)  2.161 ms  2.056 ms  1.988 ms
 2  100.91.152.1 (100.91.152.1)  2.341 ms  2.268 ms  2.440 ms
 3  10.255.6.203 (10.255.6.203)  2.375 ms  2.298 ms  2.235 ms
 4  10.255.6.195 (10.255.6.195)  3.023 ms  2.958 ms  3.154 ms
 5  10.255.8.9 (10.255.8.9)  3.091 ms  3.341 ms  3.259 ms
 6  10.255.11.141 (10.255.11.141)  8.452 ms  9.139 ms  6.100 ms
 7  10.255.10.14 (10.255.10.14)  4.276 ms  3.080 ms  3.335 ms
 8  b1.366w.everywherewireless.com (204.14.39.95)  2.465 ms  2.809 ms  2.729 ms
 9  * * *
10  142.251.60.14 (142.251.60.14)  3.050 ms  142.251.60.8 (142.251.60.8)  3.529 ms  142.2
51.60.4 (142.251.60.4)  3.976 ms
11  108.170.243.165 (108.170.243.165)  3.889 ms  108.170.243.219 (108.170.243.219)  3.85
9 ms  142.251.231.245 (142.251.231.245)  5.551 ms
12  mia09s17-in-f14.1e100.net (172.217.1.110)  2.330 ms  2.222 ms  209.85.250.144 (209.8
5.250.144)  4.629 ms
```

vv. **date**

displays current date and time.

```
varvara@varvara-VirtualBox:~$ date
Fri May 31 08:23:21 PM CDT 2024
varvara@varvara-VirtualBox:~$
```

ww. **time**

measures how long it takes for a command/program to execute.

```
varvara@varvara-VirtualBox:~$ time ls
another.txt  Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  text.txt  Videos

real    0m0.010s
user    0m0.004s
sys     0m0.003s
```

xx. **wget**

non-interactive network downloader which is used to download files from the server even when the user has not logged on to the system and it can work in the background without hindering the current process.


```

varvara@varvara-VirtualBox:~$ wget https://letsenhance.io/static/8f5e523ee6b2479e26ecc9
1b9c25261e/1015f/MainAfter.jpg
--2024-06-01 20:16:20-- https://letsenhance.io/static/8f5e523ee6b2479e26ecc91b9c25261e
/1015f/MainAfter.jpg
Resolving letsenhance.io (letsenhance.io)... 151.101.66.132, 151.101.2.132, 151.101.194
.132, ...
Connecting to letsenhance.io (letsenhance.io)|151.101.66.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 86771 (85K) [image/jpeg]
Saving to: 'MainAfter.jpg'

MainAfter.jpg          100%[=====] 84.74K  --.-KB/s   in 0.01s

2024-06-01 20:16:20 (6.46 MB/s) - 'MainAfter.jpg' saved [86771/86771]

varvara@varvara-VirtualBox:~$ ls
another.txt  Documents  lab2.txt      Music      Public      text.txt
a.out        Downloads  lab.txt       Pictures    snap        Videos
Desktop      files.tar.gz MainAfter.jpg program.c   Templates
varvara@varvara-VirtualBox:~$

```

yy. **wc**

‘word count’.

```

varvara@varvara-VirtualBox:~$ ls
another.txt  Documents  lab2.txt      Music      Public      text.txt
a.out        Downloads  lab.txt       Pictures    snap        Videos
Desktop      files.tar.gz MainAfter.jpg program.c   Templates
varvara@varvara-VirtualBox:~$ wc lab.txt
 4 12 80 lab.txt
varvara@varvara-VirtualBox:~$ wc another.txt
 2 42 240 another.txt
varvara@varvara-VirtualBox:~$

```

zz. **pwgen**

password generator.

```

varvara@varvara-VirtualBox:~$ pwgen -s 25
2vLHRiNqzu4zRfvNDzYgbDR3S  S4Pm9ljCRPzzLE8IerEHIMldc  0QJFWPCQdpJTU4IfYjqdn3Qux
utw6SvImFNT3UCN47lwInJd4P  a9rRHKPha4Ex83QWR4q9TX5Q  VY2fvt80puSnIcbe8CPn0Dnx4
wAvZdi958jizcdZ95ihCtiYvf  mU9F65yVh08f4eqCilJX6DerR  sERqBrFZFHHbD66P3d3D15Htv
MEjg6VsvRpnce2K1p0uagJ39h  6Gco2pyH2Ri3Sctvthaxyj5jc  HtSef9XKFcdidMk5nDy9lHgjn
4nIeLq3L07GgSZWnUkUy5yHg5  NnwNe23JjBqqZFCNS4D1ygiOE  6ul5eZnOrvtqJ7V5d530f2aj6
lK90fWKGKzKsZLv14rLB5POCW  sfgNQ3lou9WQFWj9vcI1uWJCQ  wLQZMGMm303MdSmMp9IU3Cmp9
Q7E2IxyPjJPPRjAhn1uiEdhPe  B4E92QpA1oE6MD4ydVt6mEjpZ  ehFrWtgNvwMu40MD98WmvJzSA
q4k2QGS4e4Wx66Xjk0ISBITok  vcM0Z5BiTKiQd16S802dlunIR  4YmJfDYnBFRZkr5goI3b6J0E6
brf9EPLwVJR5sKj3i0khW4mcR  BXfjKxBgLYkpcEbr537V8UHEL  JmmL3Ek8FrVDbgXGTU52xPkjr
s6Lx7mORstGRZcp34mNOF0S4g  ehGXnIYD3h0V7eHKrzzC5sWqH  EDjypXUKYvTdGLzx2hibqGLVG
LCkHvi9lULXV0VHm1QqC58y2  lQ6g7c8wBz7UtntscacPHyr45  ioASBfJqI7zdafrnQsaBwYaPi
BZueruJuUB7fzhLFqE3pYDsc4  5FLHhSnFuP2lybqchmN3ALqXT  JEHbajR1uh1uCUWRXGrmxvml6
M56Bz10drzly2oTY3cfbpDR7Q  VbCEXN0EIHgBJuTI53LIWszEw  wT01gtRfMC1Vv14iHJU17qTpC
9EwL0ujzAPL7qTWx4l74QfgD8  QbsSzztIebYjVMx2zx0cx5zSC  irpwCSKw01Ap3rCKS5QqbVZqi
DUmPsTQXn3BghrH67A7zBn3LY  swYdwGS6nEwEM7swuWgli4vbT  DbHOFLG65KZCX4bdYRmxTgs4
WG8DlwwQ2a6nC00gDQNTypIeZ  y7AdlBYHODXFRppHEdYeeFIMi  SiHmqb1MF40gNcMhsHZsHXak6
gPXAq5ckxe8l38AQ4fxX7XJyU  YTQdZfdb3f4uPydXk7Uy9rPVQ  YJTAnZxsRIMKopJJVHFOlVt1J
eYHxRcFHi4ZETgS48opKRhMrK  QKiriBVEodSwoKceBhU4UcNLW  a7JTUsUz9MER6MCFZva9WuF0E
m0ZHeB6J67dHLh6kcmw4lyfM  6uXIBQGe9X0gedg5yGBPIJbg  RKpHNpebJK0LAUVSW33huA3yV
mUODwHRW0z0QnIz0wsNhdModR  9kyfy89vXUgZH3UPVaIB0sWXU  2hHeSAMPmq7Jd4jgc1MpS6lWN
varvara@varvara-VirtualBox:~$

```

3. Write bash scripts to do the following:

- Write a script called “generate-dataset.sh <filename> <num_records>” with two command line arguments specifying the file name to output and the number of records, where each record is separated by new line character, and each has the following format: <integer> <integer> <ASCII_string>. The integer should be any random number up to a 32-bit integer. The ASCII_string should be any string using

ASCII of exactly 100 bytes long. Use the “time” command to show how long the benchmark took to complete. The benchmark should run for at least 10 seconds, and it should complete even if the ssh (or bash) session is terminated. How many records does your file contain after running it? You must write this script entirely with existing Linux commands (which you can install if they don’t exist on your system), without using any other programming languages like C, C++, Java, or Python.

For the benchmark to complete even if the ssh session is terminated, we should use a *nohup* command.

- b. Write a script called “sort-data.sh” that takes input a file from part (a) above and sorts the file based on the first column data (make sure to only sort based on the first column data, and not on the entire line of data; also make sure you are treating the data in column 1 as numbers and not text). Use the “time” command to show how long the sort script took to complete.
 - c. Use the script in part (a) and generate 3 data files with different number of records (1000, 100000, 10000000); measure time taken to generate these records. Sort the data with your script from part (b) and measure the time. Write a Python matplotlib script to generate a graph for the time taken to generate the data and the time taken to sort the data at the 3 different scales. The graph should automatically adjust to the number of entries, and the scale of the data.
4. Answer the following questions about VMs:
- a. In the system configuration of the VM, explain how changing the number of processors changes the behavior of your VM. Explain a scenario where you want to set this to the minimum, and a scenario where you want to set it to the maximum. Why is setting it to the maximum potentially a bad idea?

Changing the number of processors used in the virtual machine affects the performance of the virtual machine. The more CPUs are dedicated to the virtual machine, the faster it operates. In a scenario where the VM is used for simple calculation tasks, such as coding or running lightweight services, it would be okay to

set the number of CPUs to minimum. In a scenario where the VM is used for tasks such as big data processing or machine learning, the number of CPUs should be increased. However, setting the number of processors to maximum possesses a risk of host processes losing performance.

- b. In the system configuration of the VM, under the Acceleration Tab, explain the difference between the paravirtualization options: None, Legacy, Minimal, HyperV, and KVM. Explain which one would be best to use with Ubuntu Linux, and why.

Paravirtualization Interface – is a comm channel for virtualization-aware OS's to talk to whatever hypervisor is running them, so they can run more efficiently.

- **None:** Disables paravirtualization interface. Generally not recommended, as it can result in poorer performance compared to other paravirtualization options.
- **Legacy:** This mode provides basic paravirtualization that is compatible with older hypervisors and guest operating systems. Suitable for older guest operating systems that do not support more advanced paravirtualization interfaces.
- **Minimal:** Announces the presence of a virtualized environment. Additionally, reports the TSC and APIC frequency to the guest operating system. This provider is mandatory for running any Mac OS X guests.
- **KVM:** Presents a Linux KVM hypervisor interface which is recognized by Linux kernels version 2.6.25 or later. Oracle VM VirtualBox's implementation currently supports paravirtualized clocks and SMP spinlocks. This provider is recommended for Linux guests.
- **Hyper-V:** Presents a Microsoft Hyper-V hypervisor interface which is recognized by Windows 7 and newer operating systems. Oracle VM VirtualBox's implementation currently supports paravirtualized clocks, APIC frequency reporting, guest debugging, guest crash reporting and relaxed timer checks. This provider is recommended for Windows guests.

KVM paravirtualization option is the most suitable for Ubuntu Linux because it is specifically designed to work with the Linux kernel.

c. In storage devices when configuring the VM, there are multiple types of storage controllers: explain the difference between the IDE, SATA, and NVMe controller. Give an example for each type of storage controller of a scenario where you may want to use this type of controller.

- **IDE (Integrated Drive Electronics) controller:** Were prevalent in older computing systems and are relatively simple in terms of functionality. They support up to two devices per controller and are commonly used for connecting hard disk drives (HDDs) and optical drives. *May be preferred for legacy systems that lack support for modern controllers or systems with low performance requirements, for example basic file servers or development environments.*
- **SATA (Serial ATA) controller:** This is a modern type of storage controller for higher hard disk data throughput, to which the virtual hard disks are attached. Provide improved performance and flexibility and support multiple devices per controller. SATA controllers can handle a mix of HDDs and SSDs, making them suitable for VMs with diverse storage requirements. *Suitable for general-purpose VMs that require moderate to high performance storage, for example web servers, database servers, and desktop environments.*
- **NVMe (Non-Volatile Memory Express) controller:** Designed specifically for SSDs and provides significantly higher performance compared to SATA controllers. They leverage the NVMe protocol, which is optimized for low-latency, high-speed access to flash-based storage. *Ideal for systems that require high-performance storage or combination of low-latency and high IOPS (Input/Output Operations Per Second), such as real-time analytics, high-frequency trading, data caching applications.*

d. In the network configuration of the VM, there are multiple types of network adapters: explain the difference between NAT, Bridged Adapter, Internal Network, and Host-only Network. Give an example for each type of network of a scenario where you may want to use this type of network.

- **NAT:** Allows the VM to communicate with the external network using the host's IP address. The host acts as a gateway, translating internal IP addresses to the host's IP address. *Scenarios: 1) When VMs need access to the internet but don't require direct communication with other VMs or devices on the same network. 2) For scenarios where VMs need internet access but should remain isolated from other devices on the local network.*
- **Bridged Adapter:** Connects VM directly to the physical network, allowing it to appear as a separate device on the same network segment as the host. *Scenarios: 1) When VMs need to be part of the same network as other devices, such as servers or printers. 2) When VM needs to be accessed by other devices on the same network.*
- **Internal Network:** Creates a private network that is isolated from the external network and accessible only and accessible only by VMs running on the same host. *Scenarios: 1) When VMs need to communicate with each other but don't require internet access or connectivity to external devices.*
- **Host-Only Network:** Host-only Network creates a private network between the host and VMs, allowing communication between them while isolating them from the external network. *Scenarios: 1) When VMs need to exchange data or services with the host system securely without exposing them to external networks. 2) When setting up development or testing environments where VMs need to communicate with the host system but don't require internet access or connectivity to external devices.*

e. For the USB configuration of the VM, explain the difference between USB 1.1, 2.0, and 3.0 controllers.

- **USB 1.1 controller**
 1. Data Transfer Rate: Up to 12 Mbps
 2. Backward Compatibility: Compatible with USB 2.0 and USB 3.0 devices but operates at USB 1.1 speeds when connected to them.
- **USB 2.0 controller**

1. Data Transfer Rate: Up to 480 Mbps.
2. Backward Compatibility: Compatible with USB 1.1 devices, operating at USB 1.1 speeds when connected to them.

- **USB 3.0 controller**

1. Data Transfer Rate: Up to 5 Gbps (5,000 Mbps).
2. Backward Compatibility: Compatible with USB 2.0 and USB 1.1 devices, operating at their respective speeds when connected to them.