

## Linux fundamentals

### Introduction

Learning is essential for security professionals to protect, defend and secure systems and networks against cyber threats. It provides the foundation for mastering security concepts, tools and techniques critical for success in the cybersecurity field.

As I was working out my assignment, I came along a couple of new concepts that I believe with practice will lay a very strong foundation for me in this realm of security.

Questions and solution I found for them.

For I to solve this question I had to connect to the target machine via secure shell(ssh) as shown below;

```
(root@kali)~[/home/mwabe]
# ssh htb-student@10.129.172.204
The authenticity of host '10.129.172.204 (10.129.172.204)' can't be established.
ED25519 key fingerprint is SHA256:PHs.jpBEAl6hSCzjVohppUybupbLXdBZy8FqtwlMpmjU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.172.204' (ED25519) to the list of known hosts.
htb-student@10.129.172.204's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-123-generic x86_64)

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

System information as of Wed May  8 05:05:52 UTC 2024

System load:  0.64               Processes:           151
Usage of /:   64.7% of 6.76GB    Users logged in:    0
Memory usage: 20%               IP address for ens192: 10.129.172.204
Swap usage:   0%

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Last login: Wed Sep 23 22:09:41 2020 from 10.10.14.6
htb-student@nixfund:~$
```

+ 0 Find out the machine hardware name and submit it as the answer.

x86\_64

by typing and executing “uname -a” displayed all the information about the machine;name, kernel version, date released.

```
└─$ uname -a
Linux kali 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.15-2kali1 (2024-04-09) x86_64 GNU/Linux
```

+ 1 What is the path to htb-student's home directory?

/home/htb-student

I used the “cd ” command to change directory to home then the htb-student directory.

```
htb-student@nixfund:/home$ cd htb-student
htb-student@nixfund:~$ ls -la
total 32
drwxr-xr-x 4 htb-student htb-student 4096 Aug  3  2021 .
drwxr-xr-x 5 root         root         4096 Aug  3  2021 ..
-rw-r--r-- 1 htb-student htb-student  5 Sep 23  2020 .bash_history
-rw-r--r-- 1 htb-student htb-student  220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 htb-student htb-student 3771 Apr  4  2018 .bashrc
drwxr-xr-x 2 htb-student htb-student 4096 Aug  3  2021 .cache
drwxr-xr-x 3 htb-student htb-student 4096 Aug  3  2021 .gnupg
-rw-r--r-- 1 htb-student htb-student  807 Apr  4  2018 .profile
htb-student@nixfund:~$
```

+ 0 What is the path to the htb-student's mail?

/var/mail/htb-student

```
/usr/src/linux-headers-4.15.0-122-generic/include/config/megaraid/mailbox.h
/var/mail
/var/lib/ucf/cache/etc:dovecot:conf.d:10-mail.conf
/var/lib/ucf/cache/etc:dovecot:conf.d:15-mailboxes.conf
/var/lib/ucf/cache/etc:dovecot:conf.d:auth-vpopmail.conf.ext
/var/log/mail.log
/var/log/mail.log.1
htb-student@nixfund:~$
```

+ 0 Which shell is specified for the htb-student user?

/bin/bash

By executing “which bash” the results displayed showed which shell environment we are using.

```
htb-student@nixfund:/$ which bash
/bin/bash
htb-student@nixfund:/$
```

+ 0 Which kernel version is installed on the system? (Format: 1.22.3)

4.15.0

To achieve this, I ran the command as shown below and there I found the kernel version. Though this can also be achieved by running “uname -a”.

```
htb-student@nixfund:/$ uname -r
4.15.0-123-generic
htb-student@nixfund:/$
```

+ 1 What is the name of the network interface that MTU is set to 1500?

ens192

```
htb-student@nixfund:/$ ifconfig
ens192: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.129.172.204 netmask 255.255.0.0 broadcast 10.129.255.255
    inet6 dead:beef::250:56ff:fe94:f2dc prefixlen 64 scopeid 0x0<global>
    inet6 fe80::250:56ff:fe94:f2dc prefixlen 64 scopeid 0x20<link>
    ether 00:50:56:94:f2:dc txqueuelen 1000 (Ethernet)
    RX packets 1934 bytes 289154 (289.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 736 bytes 134794 (134.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1100 bytes 86785 (86.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1100 bytes 86785 (86.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

htb-student@nixfund:/$
```

```

htb-student@nixfund:/home$ cd htb-student
htb-student@nixfund:~$ ls -la
total 32
drwxr-xr-x 4 htb-student htb-student 4096 Aug  3  2021 .
drwxr-xr-x 5 root         root         4096 Aug  3  2021 ..
-rw----- 1 htb-student htb-student   5 Sep 23  2020 .bash_history
-rw-r--r-- 1 htb-student htb-student  220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 htb-student htb-student 3771 Apr  4  2018 .bashrc
drwx----- 2 htb-student htb-student 4096 Aug  3  2021 .cache
drwx----- 3 htb-student htb-student 4096 Aug  3  2021 .gnupg
-rw-r--r-- 1 htb-student htb-student  807 Apr  4  2018 .profile
htb-student@nixfund:~$

```

+ 1 📦 What is the index number of the "sudoers" file in the "/etc" directory?

147627

After I had cd into the /etc directory, I used ls together with -i in order to list the inode number of the sudoers file.

```

htb-student@nixfund:/$ cd /etc | ls -i /etc/sudoers
147627 /etc/sudoers
htb-student@nixfund:/$

```

+ 1 📦 What is the name of the config file that has been created after 2020-03-03 and is smaller than 28k but larger than 25k?

00-mesa-defaults.conf

Since I did not have an idea where this file was located, I used the “find” command to locate the file alongside with filtering command such as -newermt to filter the date specified.

```

htb-student@nixfund:~$ find / -type f -newermt 2020-03-03 ! -newermt 2024-05-08 -size +25k -size -28k -name "*.conf" 2>/dev/null
/usr/share/dirc.d/00-mesa-defaults.conf
htb-student@nixfund:~$

```

+ 0 📦 How many total packages are installed on the target system?

737

To check for this, I used the command below as shown in the image.

```
htb-student@nixfund:~$ apt list --installed | wc -l
```

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

```
738
```

+ 0 🗨 How many services are listening on the target system on all interfaces? (Not on localhost and IPv4 only)

7

Netstat -l will list all services listening on our device, but for ipv4 only, are 7 as displayed in the image below.


```
htb-student@nixfund:~$ netstat -l
Active Internet connections (only servers)
```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	localhost:mysql	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:netbios-ssn	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:pop3	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:imap2	0.0.0.0:*	LISTEN
tcp	0	0	localhost:domain	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:ssh	0.0.0.0:*	LISTEN
tcp	0	0	localhost:smtp	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:microsoft-ds	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:imaps	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:pop3s	0.0.0.0:*	LISTEN
tcp6	0	0	:::netbios-ssn	:::*	LISTEN
tcp6	0	0	:::pop3	:::*	LISTEN
tcp6	0	0	:::imap2	:::*	LISTEN
tcp6	0	0	:::http	:::*	LISTEN
tcp6	0	0	:::ftp	:::*	LISTEN
tcp6	0	0	:::ssh	:::*	LISTEN

+ 1 🗨 Use the "systemctl" command to list all units of services and submit the unit name with the description "Load AppArmor profiles managed internally by snapd" as the answer.


```
snapd.apparmor.service
```

```
htb-student@nixfund:~$ systemctl list-units --type=service | grep AppArmor
apparmor.service          loaded active exited AppArmor initialization
snapd.apparmor.service    loaded active exited Load AppArmor profiles managed internally by snapd
htb-student@nixfund:~$
```

+ 0  Submit the full path of the "xxd" binary.

`/usr/bin/xxd`

```
htb-student@nixfund:~$ locate xxd
/snap/core/10126/usr/bin/xxd
/snap/core/10126/usr/share/bash-completion/completions/xxd
/snap/core/10185/usr/bin/xxd
/snap/core/10185/usr/share/bash-completion/completions/xxd
/snap/core18/1885/usr/bin/xxd
/snap/core18/1885/usr/share/bash-completion/completions/xxd
/snap/core18/1885/usr/share/doc/xxd
/snap/core18/1885/usr/share/doc/xxd/changelog.Debian.gz
/snap/core18/1885/usr/share/doc/xxd/copyright
/snap/core18/1932/usr/bin/xxd
/snap/core18/1932/usr/share/bash-completion/completions/xxd
/snap/core18/1932/usr/share/doc/xxd
/snap/core18/1932/usr/share/doc/xxd/changelog.Debian.gz
/snap/core18/1932/usr/share/doc/xxd/copyright
/usr/bin/xxd
/usr/lib/git-core/mergetools/xxdiff
/usr/share/bash-completion/completions/xxd
/usr/share/doc/xxd
/usr/share/doc/xxd/NEWS.Debian.gz
/usr/share/doc/xxd/changelog.Debian.gz
/usr/share/doc/xxd/copyright
/usr/share/man/fr/man1/xxd.1.gz
/usr/share/man/it/man1/xxd.1.gz
/usr/share/man/ja/man1/xxd.1.gz
/usr/share/man/man1/xxd.1.gz
/usr/share/man/pl/man1/xxd.1.gz
/usr/share/man/ru/man1/xxd.1.gz
/usr/share/vim/vim80/syntax/xxd.vim
/var/lib/dpkg/info/xxd.list
/var/lib/dpkg/info/xxd.md5sums
htb-student@nixfund:~$
```

+ 0  Determine what user the ProFTPD server is running under. Submit the username as the answer.

`proftpd`

In order to find the user proftpd server is running under, I had to use "ps" to list all the processes running on our target machine, then piped the result to grep the proftpd keyword.



```
htb-student@nixfund:~$ ps aux | grep proftpd
proftpd    1690  0.0  0.1 126444 3636 ?        Ss   12:34   0:00 proftpd: (accepting connections)
htb-stu+   6237  0.0  0.0 13144 1148 pts/0    S+   13:03   0:00 grep --color=auto proftpd
htb-student@nixfund:~$
```

+ 1 📦 How many files exist on the system that have the ".bak" extension?

4

Here I used the “find” command alongside other filtering commands and then piped the output to “wc -l” in order to count the number of “.bak” files found.

```
htb-student@nixfund:~$ find / -name *.bak 2>/dev/null | wc -l
4
htb-student@nixfund:~$
```

+ 1 📦 How many files exist on the system that have the ".log" file extension?

32

```
htb-student@nixfund:~$ find / -type f -name *.log 2>/dev/null | wc -l
32
htb-student@nixfund:~$
```

+ 0 📦 What is the type of the service of the "syslog.service"?


notify

For me to find out the type of service I had to use the “—property=Type” flag to display the type of service of the syslog.service

```
htb-student@nixfund:/proc$ systemctl show syslog.service --property=Type
Type=notify
htb-student@nixfund:/proc$
```


+ 1 📦 Find a way to start a simple HTTP server inside Pwnbox or your local VM using "npm". Submit the command that starts the web server on port 8080 (use the short argument to specify the port number).

http-server -p 8080

+ 0  Find a way to start a simple HTTP server inside Pwnbox or your local VM using "php". Submit the command that starts the web server on the localhost (127.0.0.1) on port 8080.

`php -S 127.0.0.1:8080`

```
htb-student@nixfund:/$ php -S 127.0.0.1:8080
PHP 7.2.24-0ubuntu0.18.04.7 Development Server started at Sun May 12 06:35:40 2024
Listening on http://127.0.0.1:8080
Document root is /
Press Ctrl-C to quit.
```

+ 0  How many partitions exist in our Pwnbox? (Format: 0)

3

“fdisk” is a command line utility that is used for disk partitioning on Unix-like systems. It allows users to create, delete, modify, and display information about disk partitions on their system.

So for me to know the number of partitions in the pwnbox, I had to use the “-l” flag just as shown in the image below.



```
[eu-academy-1]-[10.10.15.97]-[htb-ac-1287818@htb-lpa0m2gjma]-[~]
[*]$ sudo fdisk -l
Disk /dev/vda: 50 GiB, 53687091200 bytes, 104857600 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
Disklabel type: gpt
Disk identifier: BEDCBC1A-6B76-B743-8F21-D32E9B1310E5

Device            Start      End  Sectors  Size Type
/dev/vda1         2048    96471039 96468992   46G Linux filesystem
/dev/vda2    96471040    96473087    2048    1M BIOS boot
/dev/vda3    96724992   104855551 8130560   3.9G Linux swap

[eu-academy-1]-[10.10.15.97]-[htb-ac-1287818@htb-lpa0m2gjma]-[~]
[*]$
```

Besides the questions I had to respond to, I also came across other linux concepts that were so crucial for a security analyst to know.

1. chmod : this command allows users to modify what they can do with certain files or directories by the changing their read, write, and execute privileges, here is a sample of a modification I made;

```
(mwabe@kali)-[~/Documents]
$ chmod u+x burpcert.der && ls -l burpcert.der
-rwxr--r-- 1 mwabe mwabe 940 Feb 21 08:25 burpcert.der

(mwabe@kali)-[~/Documents]
$ chmod 754 burpcert.der && ls -l burpcert.der
-rwxr-xr-- 1 mwabe mwabe 940 Feb 21 08:25 burpcert.der
```

initially I did not have the execute permission but by running the command above, from the output you can see I managed to have the execute privileges.

2. apt : I used the advanced package manager to install packages and their dependencies in my local machine.

3.dpkg :  
I used  
this as  
an

```
(root@kali)~/home/mwabe
# apt install eog
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
eog is already the newest version (45.2-1).
eog set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 55 not upgraded.

Upgrade plan:
#
```

alternative of apt, and below is a sample image;

```
root@kali ~/home/mwabe
# dpkg --add-architecture i386 && apt-get update && apt-get install wine32:i386
Hit:2 http://http.kali.org/kali Kali-rolling InRelease
Hit:3 https://download.docker.com/linux/debian bullseye InRelease
Hit:4 https://artifacts.elastic.co/packages/8.x/apt stable InRelease
Hit:5 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:6 https://deb.librewolf.net focal InRelease
Hit:1 https://packages.microsoft.com/repos/code stable InRelease
Hit:7 https://ngrok-agent.s3.amazonaws.com buster InRelease
Reading package lists... Done
N: Skipping acquire of configured file 'stable/binary-i386/Packages' as repository 'https://download.docker.com/linux/debian bullseye InRelease' doesn't support architecture 'i386'
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

4. git : after installing the git in our machine, I was able to download files from the internet directly to my local machine using the “git clone \*\*\*\*\*” command.

```
(root@kali)~/home/mwabe
# git clone https://github.com/danielmiessler/SecLists.git
Cloning into 'SecLists'...
remote: Enumerating objects: 18899, done.
remote: Counting objects: 100% (46/46), done.
remote: Compressing objects: 100% (33/33), done.
^Cceiving objects: 11% (2101/18899), 2.93 MiB | 466.00 KiB/s

#
```

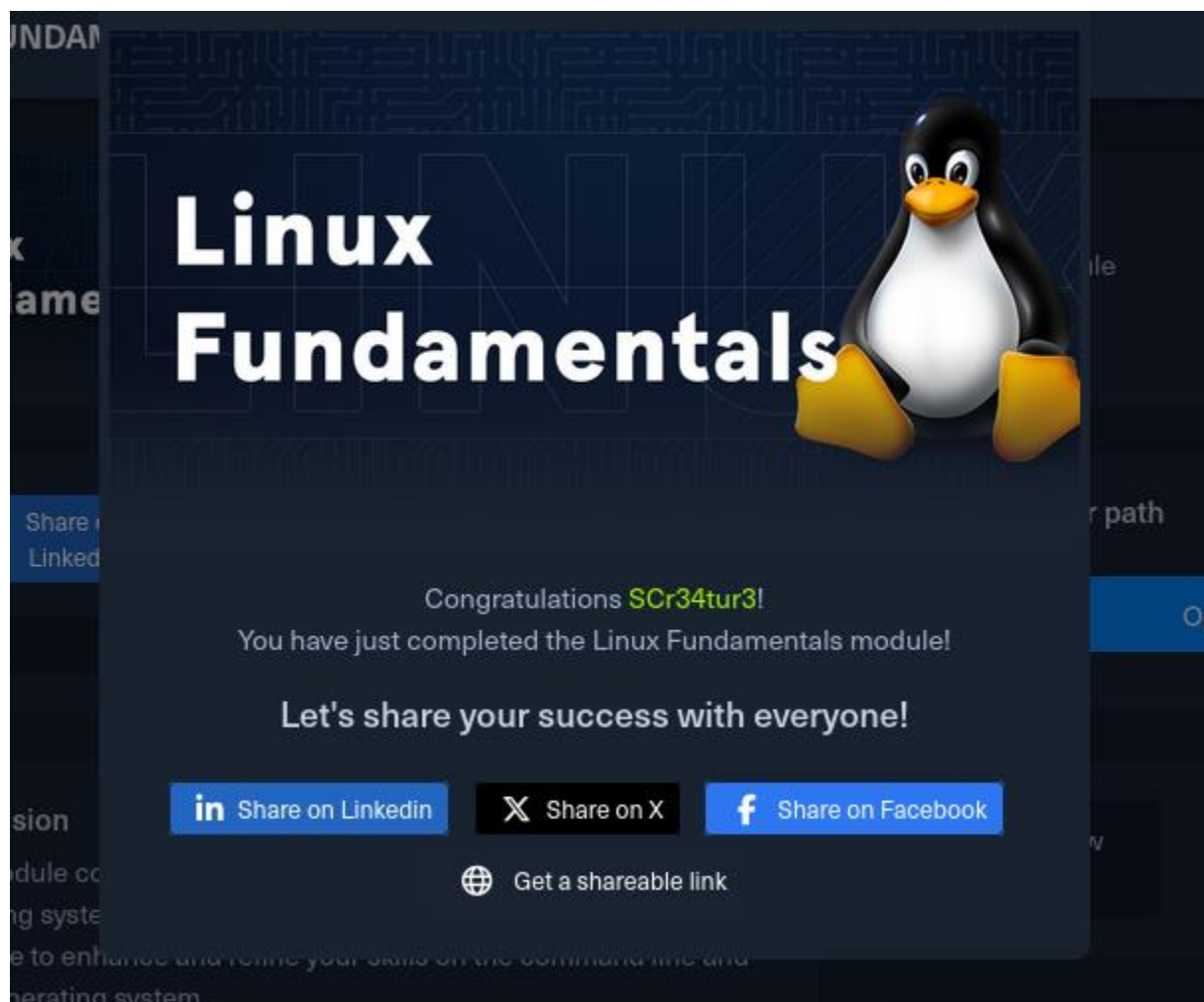
Lastly, is  
the  
daemon;  
this are  
services  
that do  
run at the

background, and they can be started and stopped or killed. Here is a sample image of me starting an ssh service in my machine.

```
(root@kali)~[/home/mwabe]
# systemctl start ssh

(root@kali)~[/home/mwabe]
# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: disabled)
   Active: active (running) since Sun 2024-05-12 07:52:20 EAT; 17s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 9037 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Main PID: 9039 (sshd)
    Tasks: 1 (limit: 9196)
   Memory: 2.7M (peak: 2.9M)
      CPU: 77ms
   CGroup: /system.slice/ssh.service
           └─9039 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

May 12 07:52:19 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
May 12 07:52:20 kali sshd[9039]: Server listening on 0.0.0.0 port 22.
May 12 07:52:20 kali sshd[9039]: Server listening on :: port 22.
May 12 07:52:20 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
```



<https://academy.hackthebox.com/achievement/1287818/18>

#### CONCLUSION:

This module has really cemented my linux knowledge alongside the one I had initially. With this module, I went to a greater depth to learn setting up firewall, vpn and how to configure networking files(network troubleshooting).

Actually it has been a fascinating module that I believe will lay a strong foundation for me in this journey of becoming a security analyst.