# PENTEST 2

# IRON CORP

# TATAKAE

# T12L

| ID | NAME | ROLE |
| --- | --- | --- |
| 1201102691 | Ahmed Muzaffar Arat | Everything |

# Step 1
# Recon and Enumeration

Tools Used: burp suite, dig, nmap.

As always, we begin by running a basic nmap on the provided IP Address. Port **8080** returned a result along with port **11025**.

We began with the ubiquitous **8080**. A control panel that seemed interesting is all we found. Despite spending some time trying to dig into the menus, nothing noteworthy was found.


Port **11025** was tried next but it was only a landing page for a “coming soon” web page.

An afxr scan was run next to try and find alternate URLs that could give us access to web pages that would otherwise be hidden/unknown.

```
└─$ dig ironcorp.me @10.10.226.101 axfr

; <<>> DiG 9.18.1-1-Debian <<>> ironcorp.me @10.10.226.101 axfr
;; 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: 204 msec
;; SERVER: 10.10.226.101#53(10.10.226.101) (TCP)
;; WHEN: Tue Aug 02 19:21:19 +08 2022
;; XFR size: 5 records (messages 1, bytes 238)
```

We still won't be able to access these newfound URLs though so we have to add them to "**/etc/hosts/**" first.

Now we can try typing "**admin.ironcorp.me:11025**" and the result is a login popup asking for a username and password. I could've tried brute forcing at that point but it was best to see if there were any protections against it like IP blacklists or timeouts. I decided to also check for common passwords as that would be much easier and a huge time saver. I tried a few combinations before the combination "**admin**" and "**password123**" worked in their respective positions. This means I've successfully avoided any possible IP bans and timeouts, and saved time.

# Step 2
# Reverse Shell

Tools Used: Netcat, Developer Tools, TinEye.

After entering the correct username:password combination, we gain access to a webpage that has a search bar and a background. I reverse searched the image on **TinEye** and it seems to be a picture from the anime "**OnePiece**" but I doubt that has any use.


I looked through the page source using "**Ctrl+U**" and found an interesting snippet of code.

```
<form method="GET"action="#">
<span>Search:
      <input name="r"type="text"placeholder="******"/>
      <input type="submit"/>
</span>
```

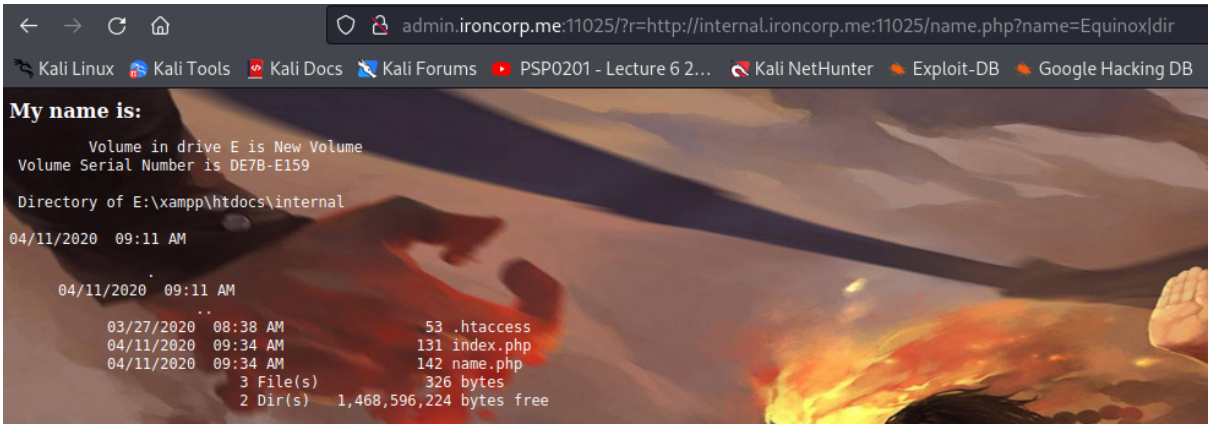
Through the same page source:

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

The URL in the snippet above won't return a useful output on its own since it seems to relay a command.

Seemingly, we need a name to authorise the search as well. Brute forcing returned "**Equinox**" as the name we can use for commands which means we place after "**..?name=**". We also use "|" to separate the name from our command.

We can start by taking a look at the directory tree using "**dir**".




This confirms that we can run commands.

Next we can use a reverse shell to gain access to the files on the server. The server is definitely a Windows machine judging from the drive letter we were provided with after running the "**dir**" command. Linux doesn't use drive letters after all. What we need is a Powershell script for this case.

I used a script from nishang on github.com. I added the machine's IP Address and target port number to the script. We also run a python server at this point so we can upload our reverse shell script to the victim's machine.

Netcat listener is set to port **6666**.

We are almost set to execute our script but before that we need to double encode the command into a URL to bypass any protections that may have been deployed.

Burpsuite is the easiest and most common encoder so I went with that.

After double encoding the command and using the name "Equinox" on the URL, Netcat receives a response on port 6666 which means we have successfully infiltrated the victim's machine.

# Step 3
# Obtaining User Flag

Tools Used: N/A

Now that we have access to the victim's machine, we can cd into the C: drive and then into "C:\Users\Administrator". We can run a "tree" scan then copy paste the result into a text file. Using "Ctrl+F" search for "user" or "flag." One of the results is "user.txt" on "C:\Users\Administrator\Desktop". Now use the "cat" command to read the file and we have found our first flag.

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

**thm{09b408056a13fc222f33e6e4cf599f8c}**

## Step 4
## Obtaining Root Flag

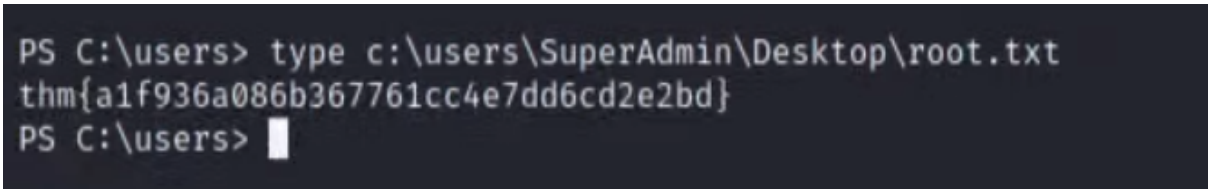
Tools Used: N/A

Checking the other users on the machine is rather simple in this regard. We can “cd ..” from the Administrator account and back into “C:\Users\” where we can access every user except “Admin” and “SuperAdmin”. I went with “SuperAdmin” first since it seemed the last flag would definitely be in that account if it was **Super.**

I initially tried to access the account using simple ls commands to get to the root flag in a step by step fashion but this failed since I didn’t have the privileges required.

Tryhackme’s page shows that the root flag is contained in a .txt flag and the previous user flag was placed on the desktop so perhaps we can access it directly using the “type” command and the full file address.

To put this theory to the test, I used “type C:\Users\SuperAdmin\Desktop\root.txt” and it worked.




**thm{a1f936a086b367761cc4e7dd6cd2e2bd}**

# SUCCÈS

Contributions:
There are no other contributors to this pentesting as was in the first pentest.

| ID | Name | Contribution | Signature |
|---|---|---|---|
| 1201102691 | Ahmed Muzaffar Arat | Everything |  |