Final Year Project Report

Full Unit – Final Report

NETWORK ATTACKS DETECTION USING PROTOCOL ANALYZERS: PRACTICAL IMPLEMENTATION AND IDENTIFICATION OF THREE ATTACKS

SAHIL NIRWAL AND VIVEK SRIVASTAVA

A report submitted in part fulfilment of the Course of

Post Graduate Diploma in Cyber Security and Law (PGDCSL)

Supervisor: Mr. Rahul Dahiya

SHAHEED SUKHDEV COLLEGE OF BUSINESS STUDIES

Post Graduate Diploma in Cyber Security and Law

Shaheed Sukhdev College of Business Studies (CBS)

JAN 18, 2025

Declaration

This report has been prepared on the basis of my own work. Where other published and unpublished source materials have been used, these have been acknowledged.

Word Count:

Student Name: Sahil Nirwal and Vivek Srivastava

Date of Submission:

Signature:

Table of Content

Abstract

In today's interconnected environment, network security is a critical priority as data breaches and cyberattacks continue to pose significant risks. A key component of ensuring secure networks involves detecting and preventing malicious activities. This report focuses on leveraging protocol analyzers to detect network attacks, presenting a practical implementation and detailed analysis of five common attack types.

The study begins by providing an overview of network attack types and examining the role of protocol analyzers in their detection. It elaborates on the practical setup of a network environment for analysis, encompassing the selection and configuration of protocol analyzers. The project then simulates and evaluates three prominent network attacks: Denial of Service (DoS), ARP Poisoning and Brute Force.

Each attack scenario is carefully designed to replicate real-world conditions, facilitating a thorough analysis of attack patterns, packet signatures, and network behavior. By employing protocol analyzers, the report demonstrates effective techniques for real-time detection and mitigation of these attacks. Additionally, it explores the nuances of identifying attacks, including differentiating between legitimate and malicious network traffic.

The project further evaluates the performance and efficiency of various detection methodologies, examining key metrics such as detection accuracy, false positive rates, and resource consumption. This empirical analysis offers critical insights into the strengths and limitations of different detection approaches, enabling network administrators to develop more resilient security strategies.

In conclusion, this report contributes to advancing network security by presenting a practical framework for utilizing protocol analyzers to detect and mitigate network attacks. By understanding the behavior and characteristics of common attack types, organizations can strengthen their defenses and protect vital assets against evolving cyber threats.

Project Specification

1. Introduction

Network security serves as a cornerstone in ensuring the integrity, confidentiality, and availability of data exchanged across digital networks. As cyber threats continue to evolve in complexity and frequency, the need for advanced and proactive security measures becomes more critical. Among the diverse tools available for securing networks, protocol analyzers stand out for their ability to monitor, inspect, and analyze network traffic in real time. These tools play a pivotal role in identifying anomalous behaviors, which often serve as indicators of cyber attacks.

This project emphasizes the practical application of protocol analyzers for network attack detection, focusing on three prevalent attack types: Denial of Service (DoS), Brute Force, and ARP Poisoning. By simulating these attacks in a controlled laboratory setting, the project aims to explore their unique characteristics, including traffic patterns, packet signatures, and behavioral anomalies. This exploration facilitates the development of effective detection mechanisms designed to protect network infrastructures against such threats.

Through comprehensive experimentation, the project will delve into the capabilities of protocol analyzers, uncovering how these tools can differentiate between benign and malicious traffic. The study will also highlight the importance of understanding the behavioral nuances of each attack to enable swift detection and response.

Ultimately, this project contributes to the field of network security by providing a hands-on approach to attack detection and mitigation. It also aims to enhance the knowledge base on best practices for utilizing protocol analyzers as an integral part of modern cybersecurity strategies.

2. Practical Implementation

The project will involve the establishment of a controlled network environment designed specifically for simulating and analyzing cyber attacks. This environment will be configured to emulate real-world scenarios, enabling a realistic examination of the threats posed by Denial of Service (DoS), Brute Force, and ARP Poisoning attacks. The setup will include a variety of tools and systems to facilitate the detection and study of attack behaviors.

3. Identification of Attacks

Using protocol analyzers, the project will focus on monitoring network traffic to identify the distinct patterns and signatures associated with each of the selected attack types. This step involves detailed traffic analysis to pinpoint specific anomalies, such as unusual packet frequencies, unauthorized access attempts, or malicious ARP broadcasts. These insights will form the foundation for designing effective detection strategies.

4. Data Collection

The project will employ real-time data collection methods to capture and analyze network traffic.

This will involve monitoring various metrics, including packet size, frequency, and source/destination details. The collected data will serve as a basis for identifying deviations from normal traffic behavior, providing crucial insights into potential attack scenarios.

5. Alert System

An integral component of the project is the development and implementation of an alert system. This system will be designed to trigger notifications or alarms whenever malicious activity is detected. By leveraging real-time monitoring capabilities, the alert system will provide network administrators with immediate feedback on security incidents, enabling timely and informed responses to mitigate potential damage.

6. Reporting and Analysis

Comprehensive reporting is a key outcome of this project. Detailed reports will be generated to document the detected attacks, including their characteristics, patterns, and impact on network performance. Additionally, these reports will include an assessment of the effectiveness of the detection mechanisms and offer actionable recommendations for enhancing network security. The analysis will also explore potential improvements to the deployed strategies, ensuring that the findings are applicable to broader cybersecurity contexts.

By addressing the practical challenges of network attack detection, this project not only strengthens defenses against specific threats but also contributes to the development of more resilient and adaptive security frameworks. Through a combination of theoretical understanding and hands-on application, it underscores the critical role of protocol analyzers in modern network security.

Introduction

In an era of increasing digital interconnectedness, network security remains a critical priority for organizations. The persistent evolution of cyber threats, including data breaches, unauthorized access, and service disruptions, underscores the urgent need for robust detection and mitigation mechanisms. Malicious actors exploit vulnerabilities to compromise network infrastructures, necessitating proactive defensive strategies.

Protocol analyzers, also known as network sniffers, are pivotal tools for monitoring and analyzing network traffic. These tools provide insights into packet-level communication, enabling the identification of anomalous behaviors and malicious activities. This project

explores the use of protocol analyzers in detecting and analyzing three prevalent network attacks: Denial of Service (DoS), ARP Poisoning, and Brute Force.

Project Objectives

This study aims to:

1. Simulate three common network attacks in a controlled environment to examine their characteristics and behaviors.
2. Utilize protocol analyzers to monitor network traffic, identifying patterns and indicators of malicious activity.
3. Develop practical detection mechanisms to mitigate security threats effectively.

Methodology

The project involves:

1. **Environment Setup:** Establishing a test network to simulate real-world attack scenarios.
2. **Traffic Analysis:** Deploying protocol analyzers to capture and analyze network traffic, focusing on specific attack signatures.
3. **Attack Detection:** Identifying key patterns, such as traffic spikes (DoS), manipulated ARP tables (ARP Poisoning), and repeated login attempts (Brute Force).
4. **Evaluation:** Assessing the effectiveness of protocol analyzers based on detection accuracy, resource efficiency, and response times.

Significance

The project highlights the importance of protocol analyzers in proactive network defense, demonstrating their utility in identifying and mitigating cyber threats. By providing practical insights into attack detection and analysis, the study contributes to the advancement of network security practices.

This exploration emphasizes the critical role of protocol analyzers in addressing the challenges of evolving cyber threats. By equipping organizations with practical detection strategies, the project underscores the importance of proactive security measures to safeguard critical assets and ensure network resilience.

Chapter 1: ARP Poisoning Attack

1.1 Description

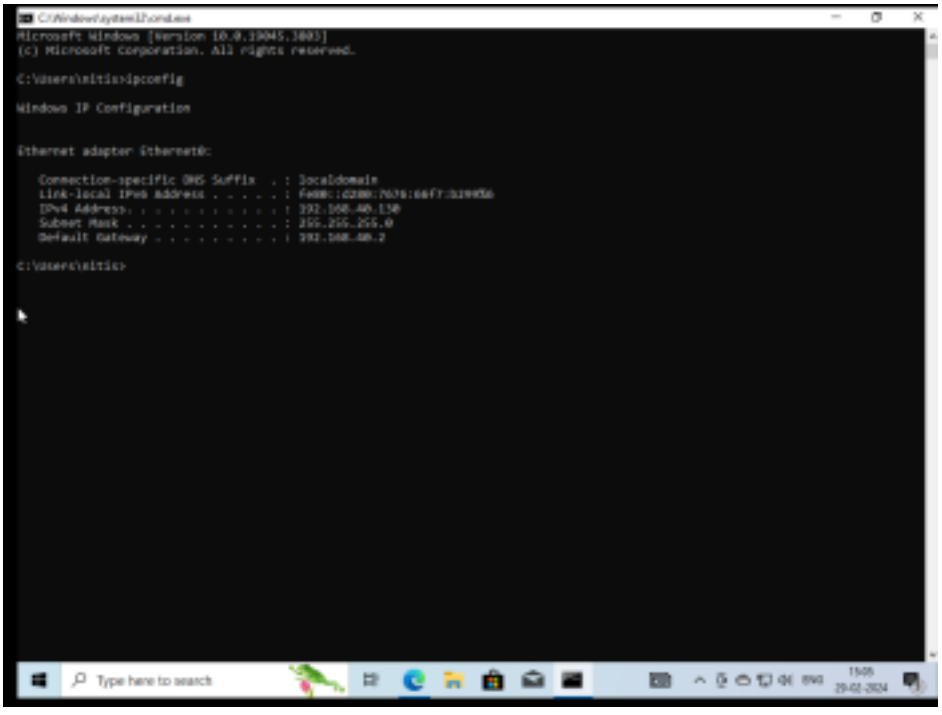
ARP (Address Resolution Protocol) Poisoning, also known as ARP Spoofing or ARP Cache Poisoning, is a type of cyber-attack where an attacker intercepts, modifies, or redirects network traffic between two devices by poisoning the ARP cache of one or both devices. ARP poisoning attacks exploit vulnerabilities in the ARP protocol, which is used to map IP addresses to MAC addresses on a local area network.

The ARP protocol operates by broadcasting ARP request packets to resolve the MAC address associated with a given IP address. The device with the corresponding IP address responds with its MAC address, and this mapping is stored in the ARP cache of the requesting device. In an ARP poisoning attack, the attacker sends falsified ARP packets to one or more devices on the network, associating their own MAC address with the IP address of another device, such as the default gateway or a target victim.

1.2 Practical Implementation

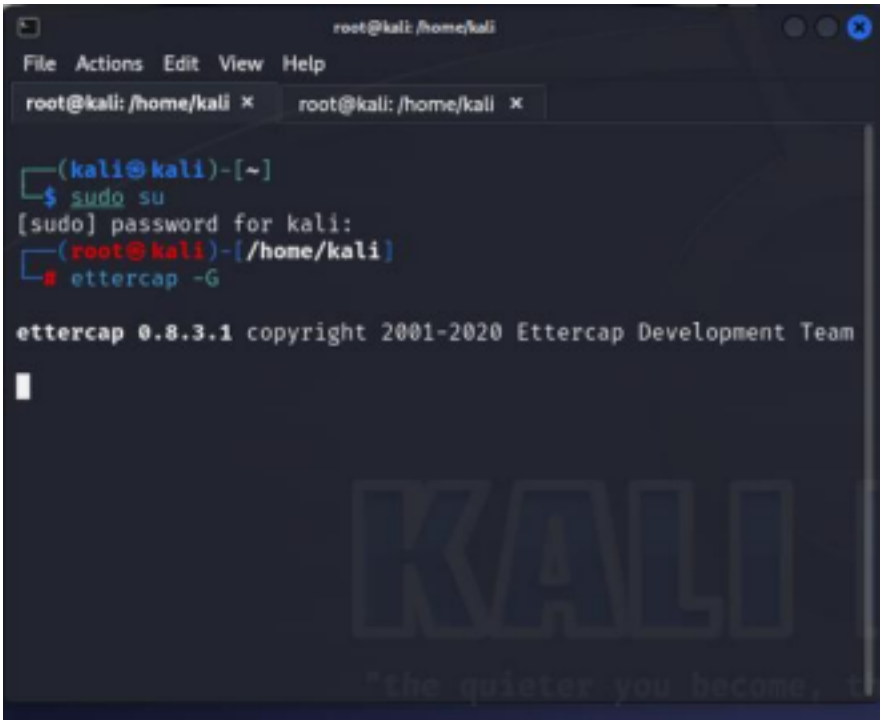
1. Install a protocol analyser such as Wireshark on a computer connected to the network.
2. Start capturing traffic on the network interface that corresponds to the network segment where ARP spoofing may occur.

Step 1: Open command prompt and get Ip address of the target machine.

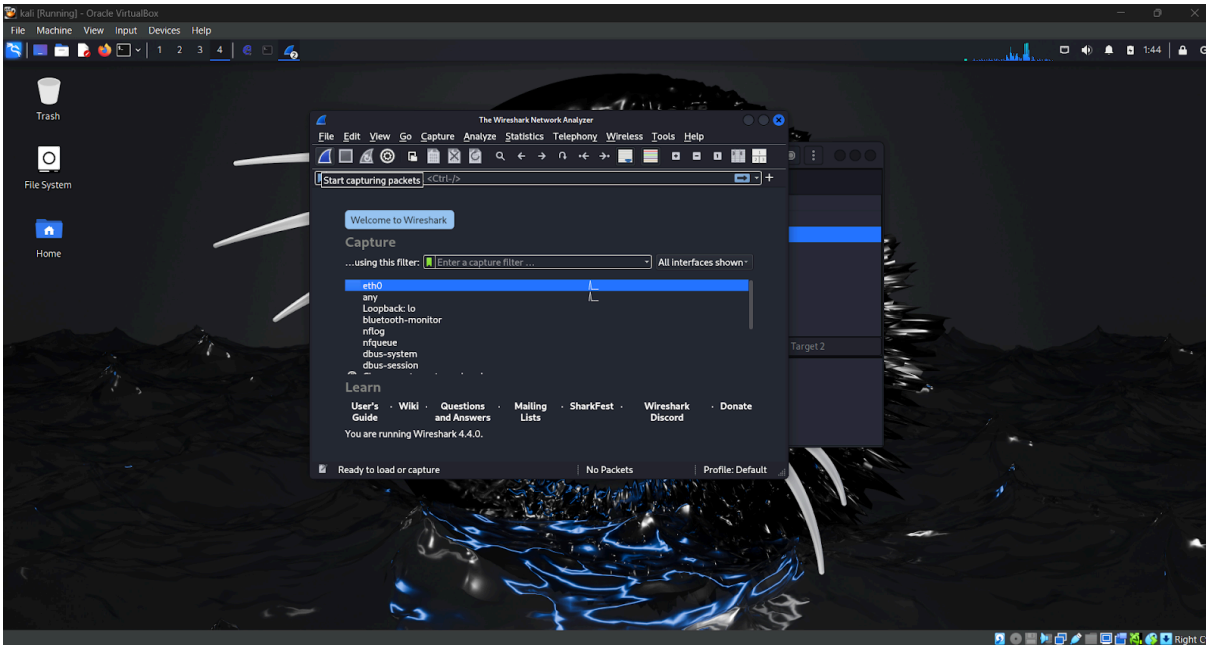


Step 2: Power on kali in VMware, open terminal in kali Linux.

Step 3: Go on root, and write commands to function Ettercap.



Step 4: open wireshark and click on “start scanning”

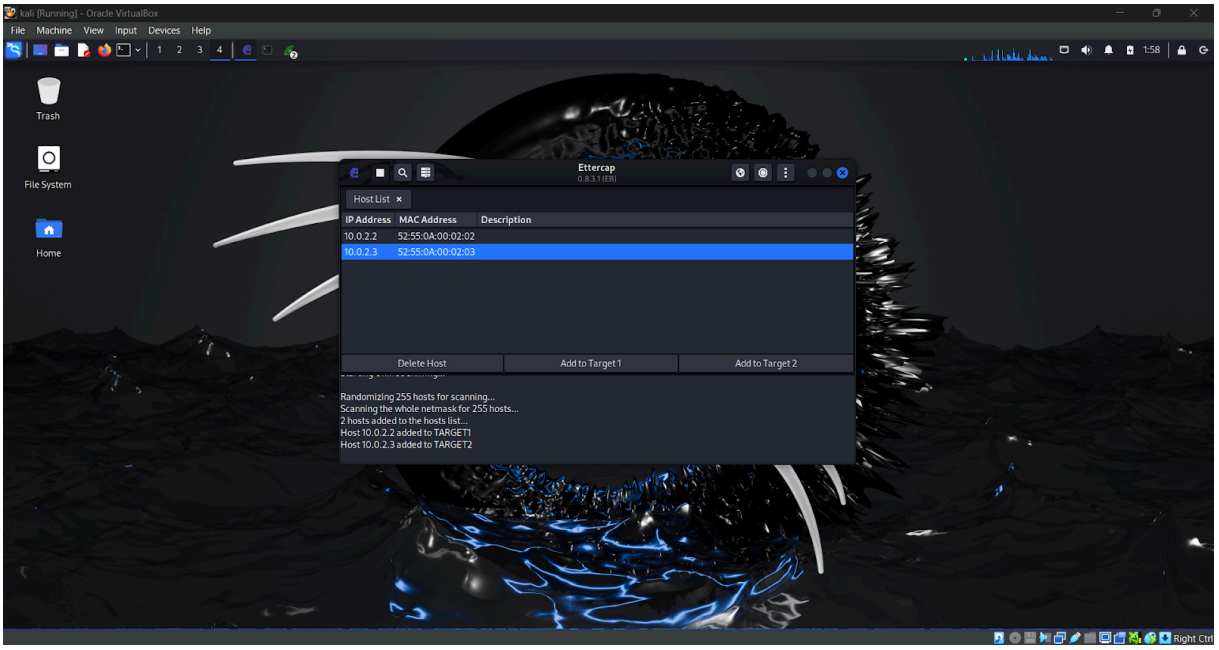


Step 5: Once Ettercap is open, click on to start unified sniffing and choose the interface as per connection and click ok.

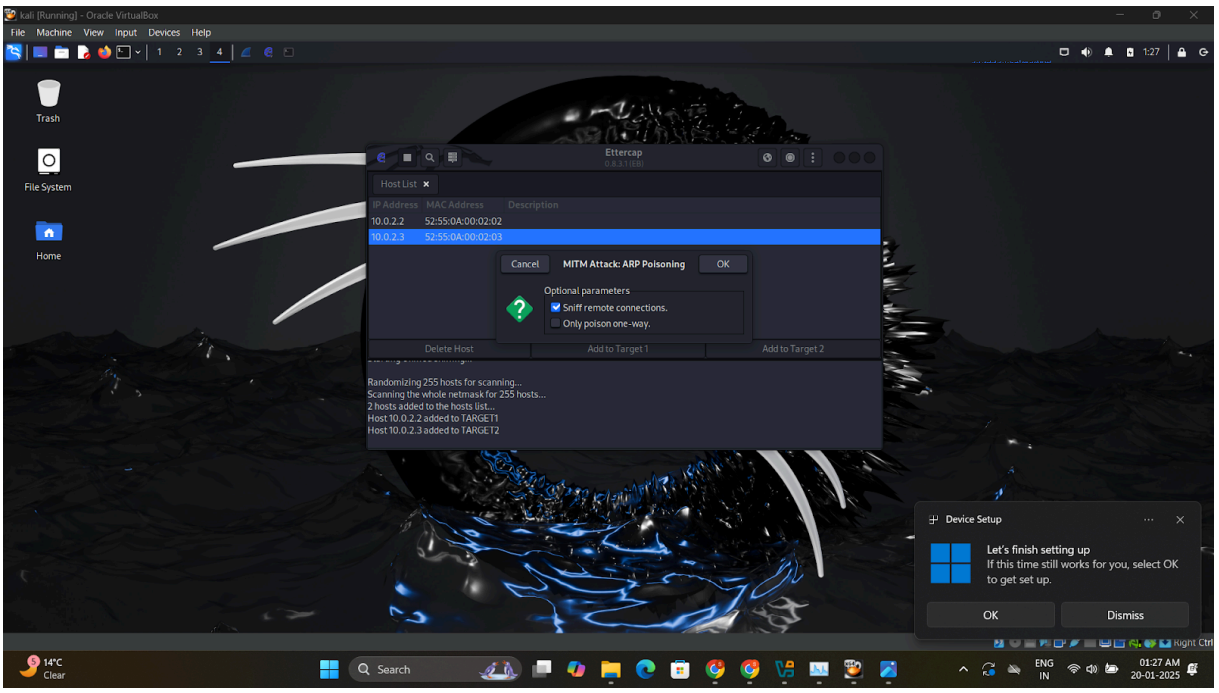
Step 6: Click on host to scan for hosts.

Step 7: click on the host list to check the host connection.

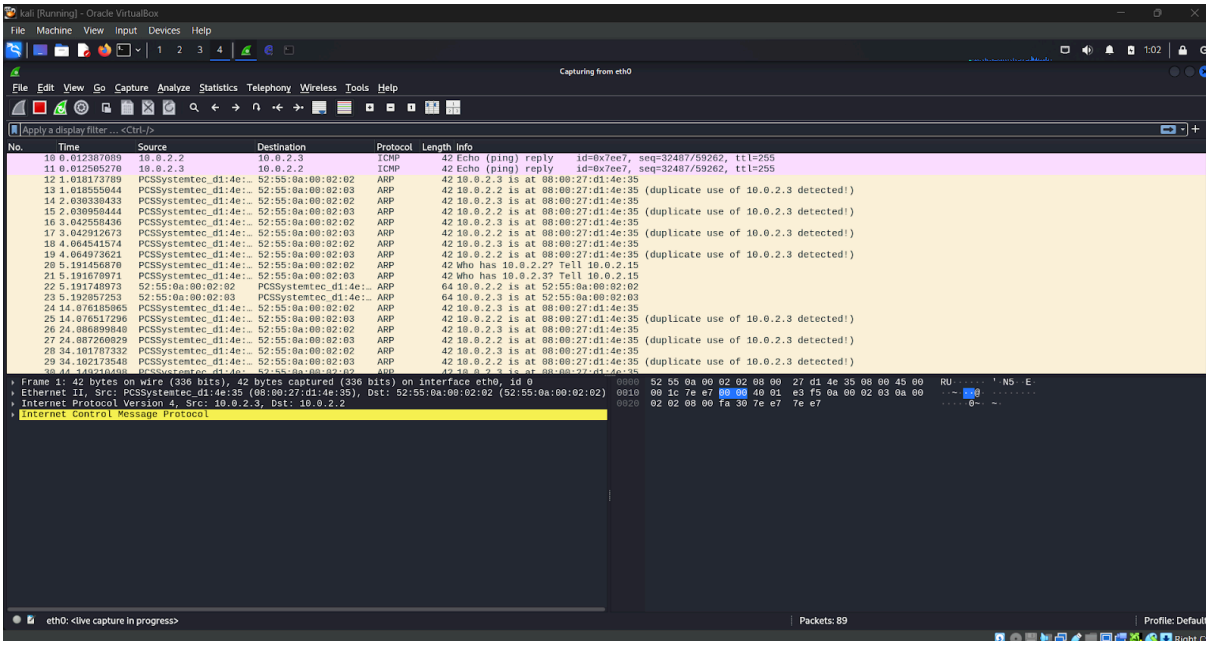
Step 8: Set Target 1 and Target 2 according to your victim machines and gateway.



Step 9: Now click on ARP poisoning, start sniffing.



Step 10: Open Wireshark again, and you will see "ARP" mentioned in the protocol section, indicating that ARP poisoning is occurring.



1.3 Identification

1. Filter captured packets to display only ARP traffic.
2. Look for ARP responses with conflicting MAC/IP address mappings or multiple ARP replies for a single IP address.

CHAPTER 2: DOS Attack

2.1 DESCRIPTION

A denial-of-service (DoS) attack is a type of cyberattack in which a malicious actor aims to render a computer or other device unavailable to its intended users by interrupting the device's normal functioning. DoS attacks typically function by overwhelming or flooding a targeted machine with requests until normal traffic is unable to be processed, resulting in denial-of-service to addition users. A DoS attack is characterized by using a single computer to launch the attack.

A distributed denial-of-service (DDoS) attack is a type of DoS attack that comes from many distributed sources, such as a botnet DDoS attack.

The primary focus of a DoS attack is to oversaturate the capacity of a targeted machine, resulting in denial-of-service to additional requests. The multiple attack vectors of DoS attacks can be grouped by their similarities.

Buffer overflow attacks:

An attack type in which a memory buffer overflow can cause a machine to consume all available hard disk space, memory, or CPU time. This form of exploit often results in sluggish behavior, system crashes, or other deleterious server behaviors, resulting in denial-of-service.

Flood attacks: By saturating a targeted server with an overwhelming of packets, a malicious actor is able to oversaturate server capacity, resulting in denial-of-service. In order for most DoS flood attacks to be successful, the malicious actor must have more available bandwidth than the target.

2.2 Practical Implementation

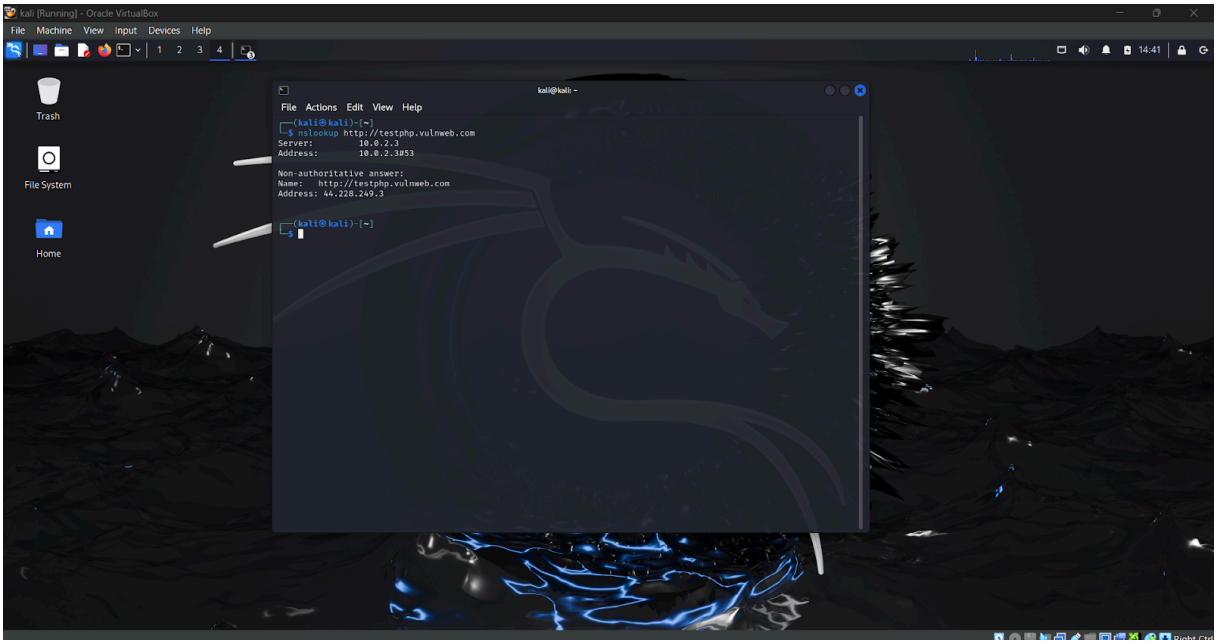
1.This type of attack is usually implemented by hitting the target resource such as a web server with too many requests at the same time.

2.This results in the server failing to respond to all the requests. The effect of this can either be crashing the servers or slowing them down.

steps to Perform the SYN Flood Attack and Analyze Packet Transmission

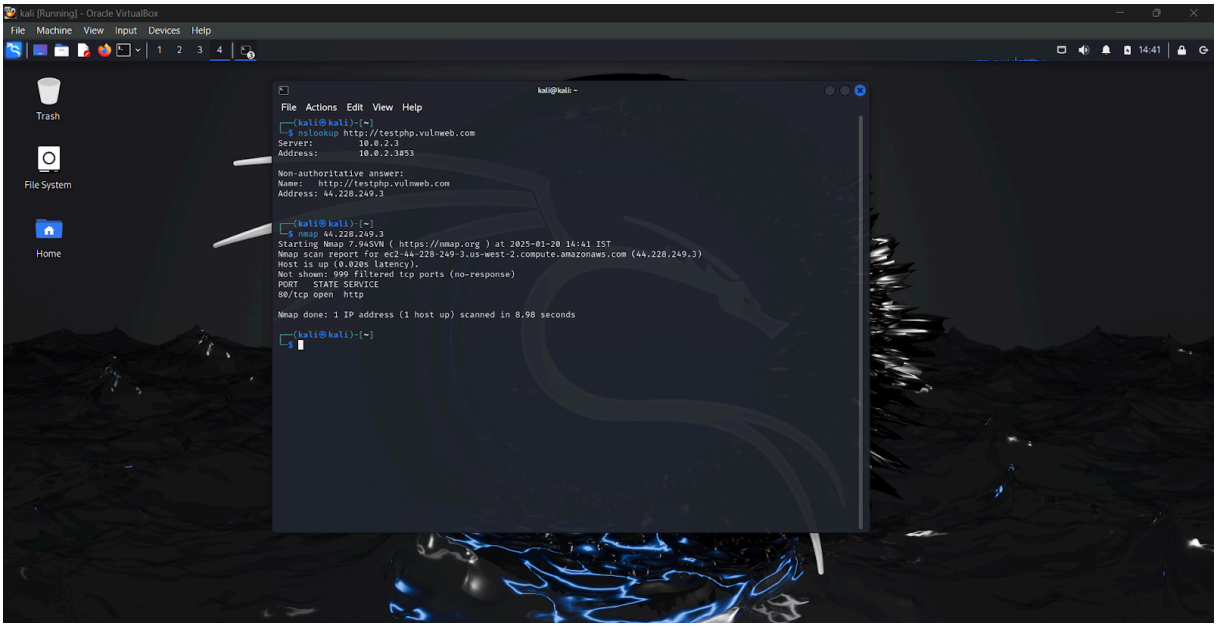
Step 1: DNS Lookup

- Use the `nslookup` command to find the IP address of the target web server



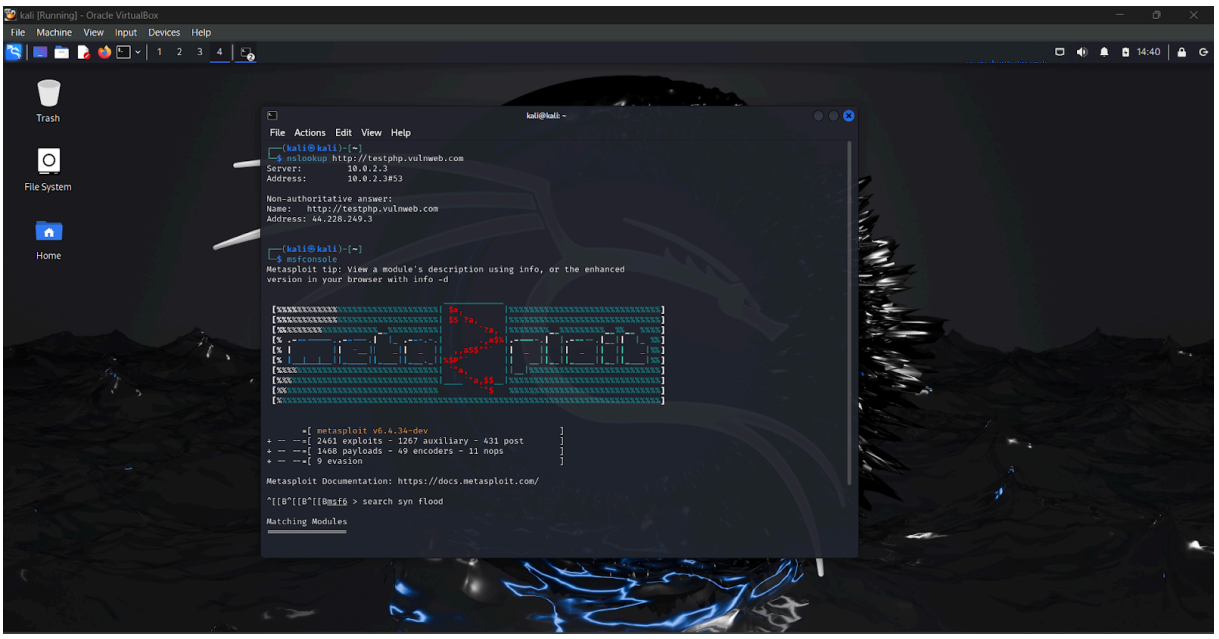
Step 2: Network Scanning

- Perform a port scan on the IP address obtained (44.228.249.3) from Step 1 using the `nmap` command. This will identify open ports on the target web server.



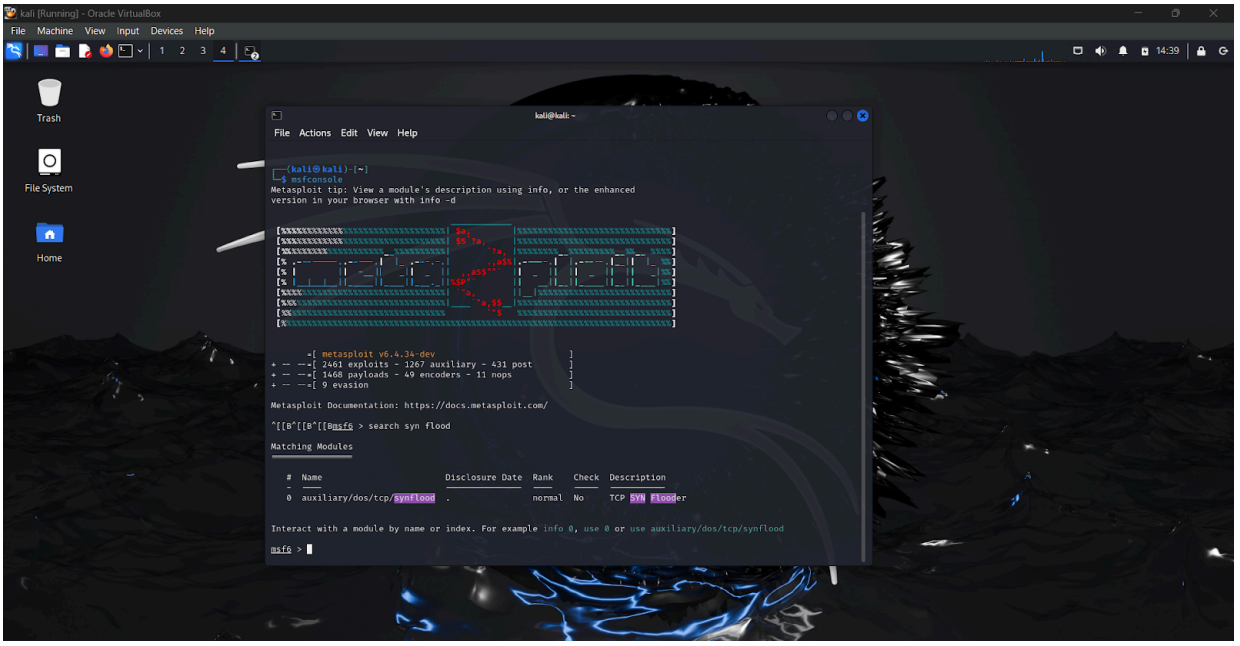
Step 3: Start Metasploit Framework

- Launch the Metasploit Framework by typing: sudo msfconsole



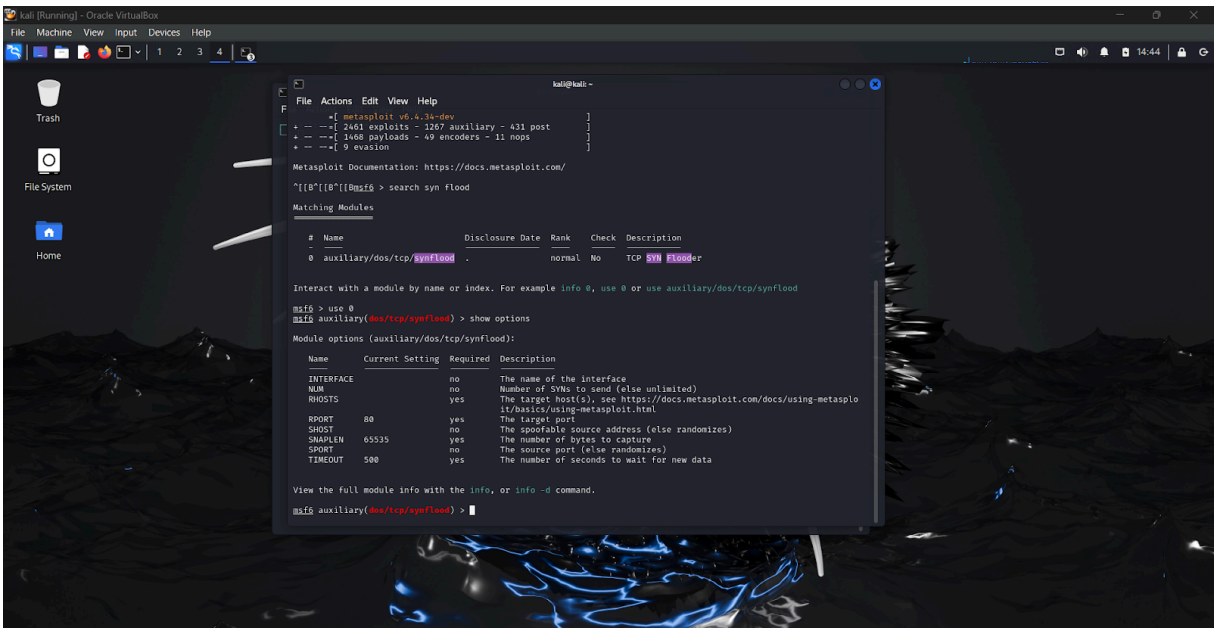
Step 4: Search for SYN Flood Module

- Search for the SYN Flood attack module in Metasploit



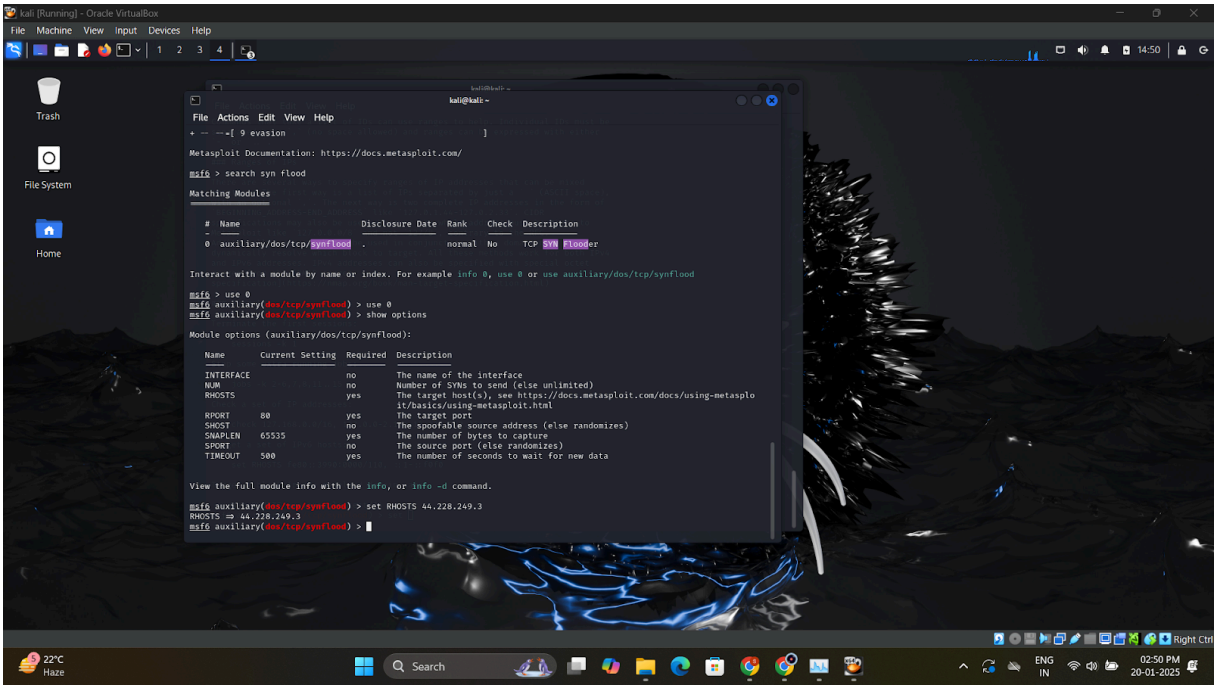
Step 5: Select and Configure the Module

- Use the desired SYN Flood module



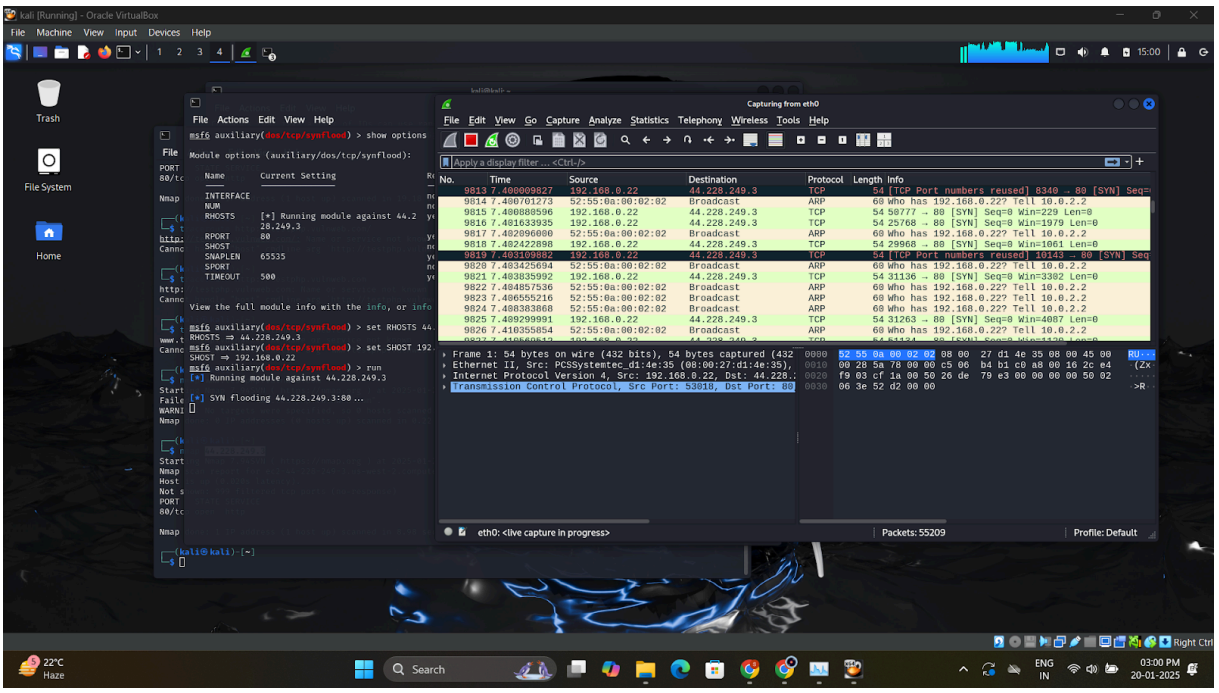
Step 6: Configure Target and Attack Parameters

- Set the `RHOSTS` parameter to the IP address of the target web server
- Set the `SHOST` parameter to your attacking machine's IP address (e.g., `192.168.0.22`)
- Start the SYN Flood attack:



Step 7: Monitor Packet Transmission

- Open **Wireshark** to monitor network traffic and analyze the number of packets being sent from the attacking machine (`192.168.0.22`).
    - Apply a filter in Wireshark for the source IP (`192.168.0.22`) to isolate relevant packets.
    - Observe the packet count and traffic behavior during the SYN Flood attack.



2.3 Identification

The best way to detect and identify a DoS attack would be via network traffic monitoring and analysis. Network traffic can be monitored via a firewall or intrusion detection system.

CHAPTER 3: BRUTE FORCE ATTACK

3.1 Description

A brute force attack is a hacking method that uses trial and error to crack passwords, login credentials, and encryption keys. It is a simple yet reliable tactic for gaining unauthorized access to individual accounts and organizations' systems and networks. The hacker tries multiple usernames and passwords, often using a computer to test a wide range of combinations, until they find the correct login information.

The name "brute force" comes from attackers using excessively forceful attempts to gain access to user accounts. Despite being an old cyberattack method, brute force attacks are tried and tested and remain a popular tactic with hackers.

3.2 Practical Implementation

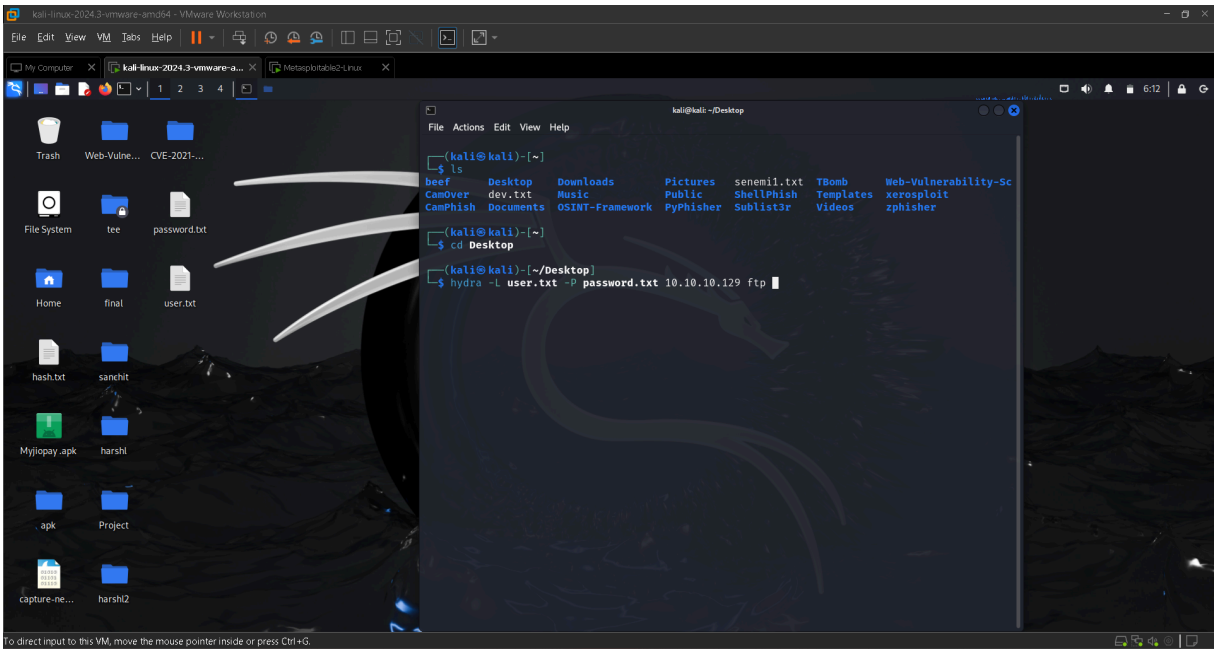
1.To hack gds password of the server in which we use ip address and username of the server which we know and used for hacking the password of the server.

2.After this we use ssh for remote access which allow you to login in the server/computer.

3. For hacking the password of we use hydra password cracking tool in the bruteforce technique to crack the password we use wordlist.

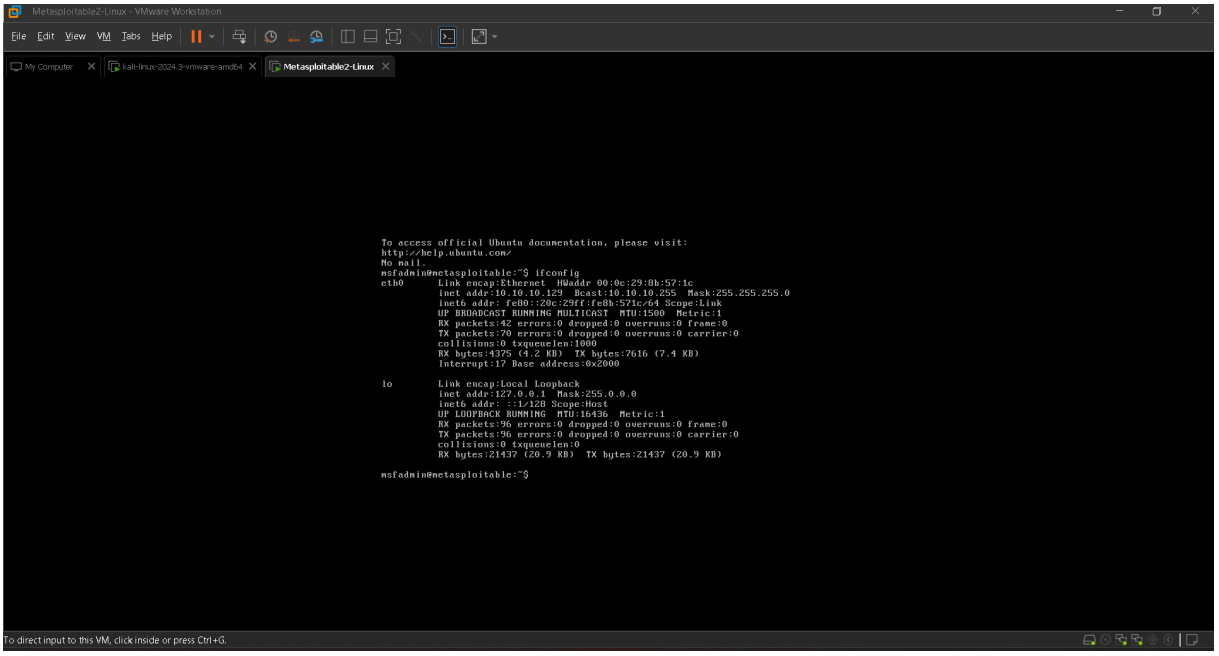
Step 1: Firstly, found the ip address of the server.

>cd Desktop (In Kali)



Step 2:

> ifconfig (In metasploitable-2)

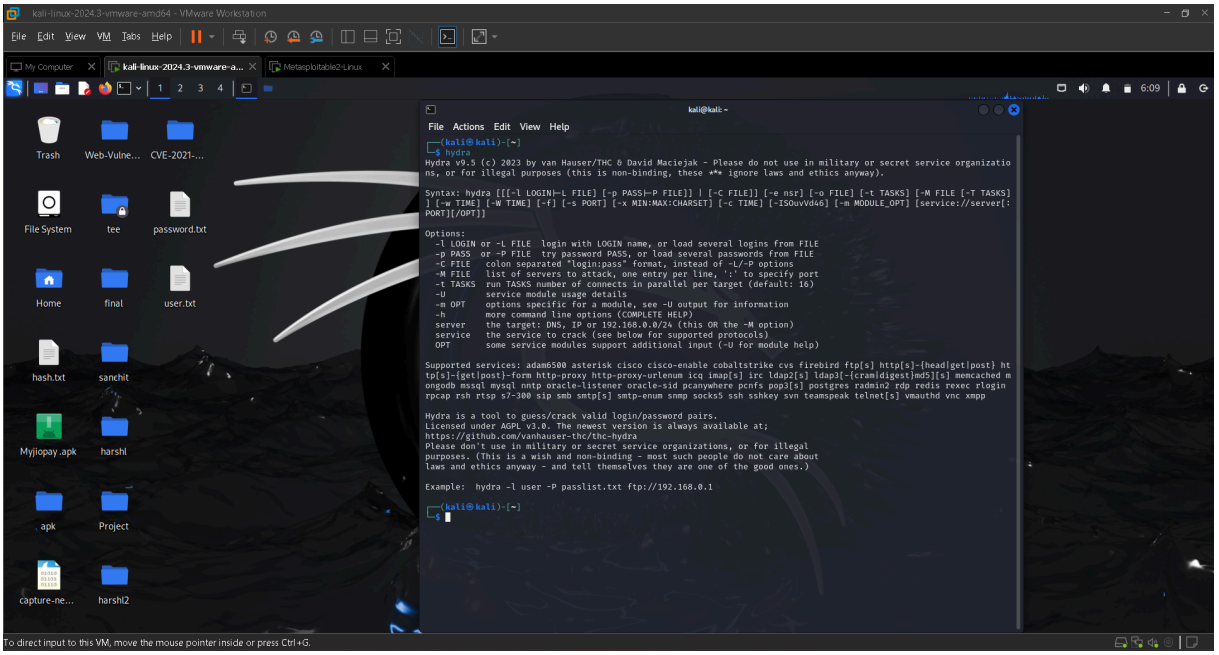


Step 3: Use ip address of the server and username of the server and the mission is hacking the password of the server.

>#ip address 10.10.10.129 (Metasploitable-2 machine)

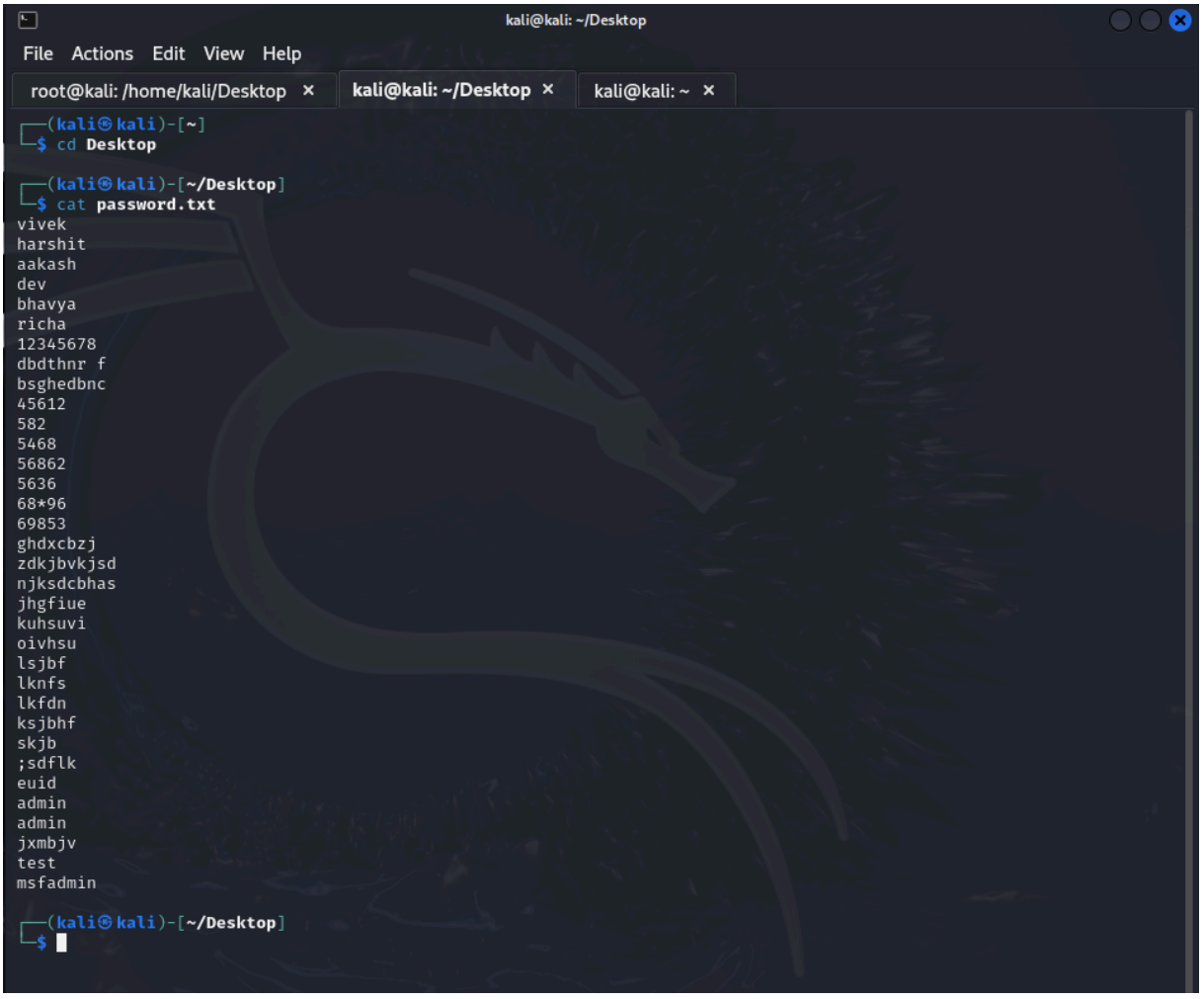
>#username -L <user.txt> User name list

>#password -P <password.txt> Password list



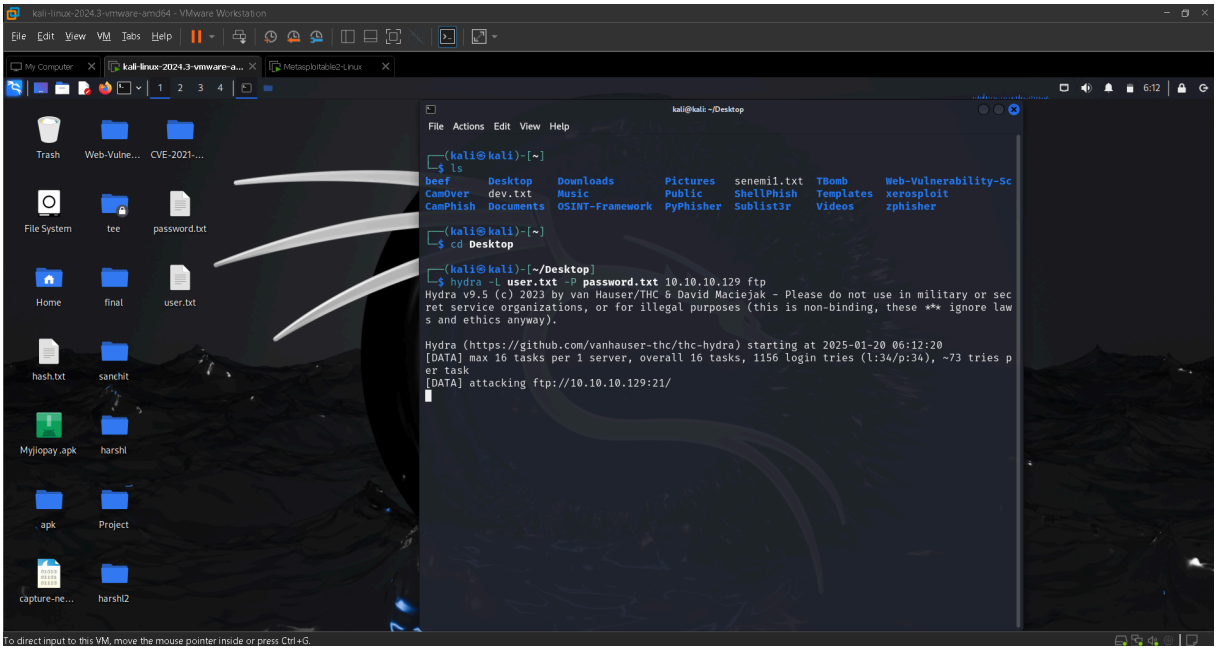
Step-4 : Create a Password List & User List

> Check file using command “cat user.txt” & “cat password.txt”.

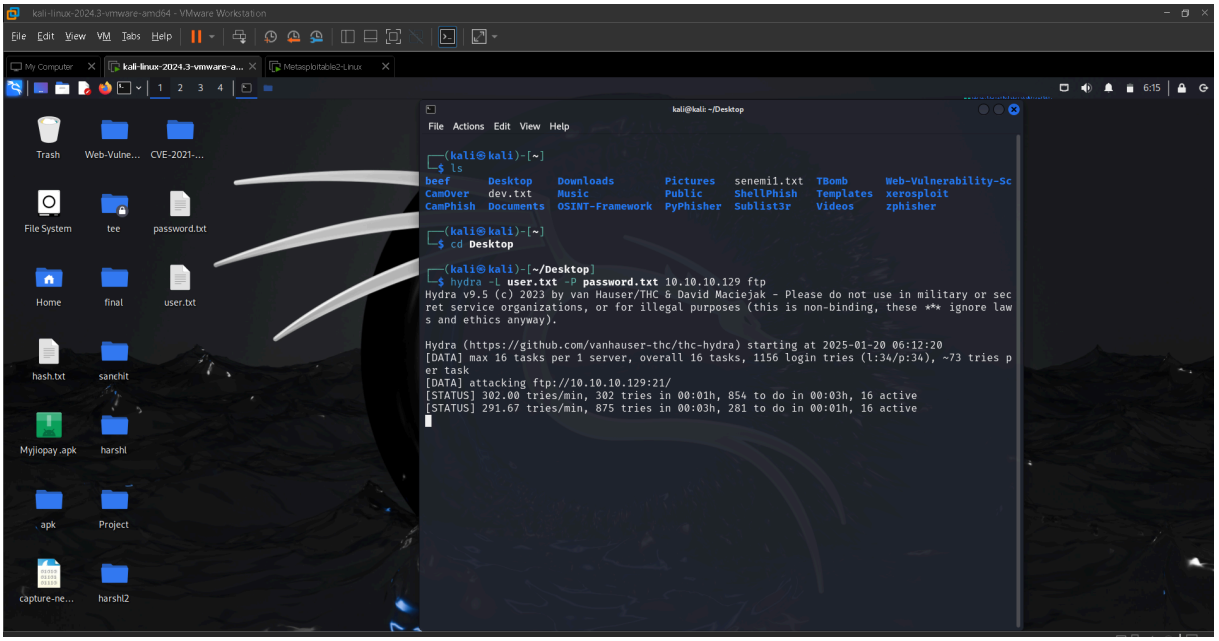


Step 5: For remotely login in the sever we use the command which ask the username & password of metasploitable-2 if you do invalid or wrong username then it denied the permission.

>ftp 10.10.10.129



Step 6: we hack the password by hydra tool which gave the wordlist for cracking the password.
>#hydra

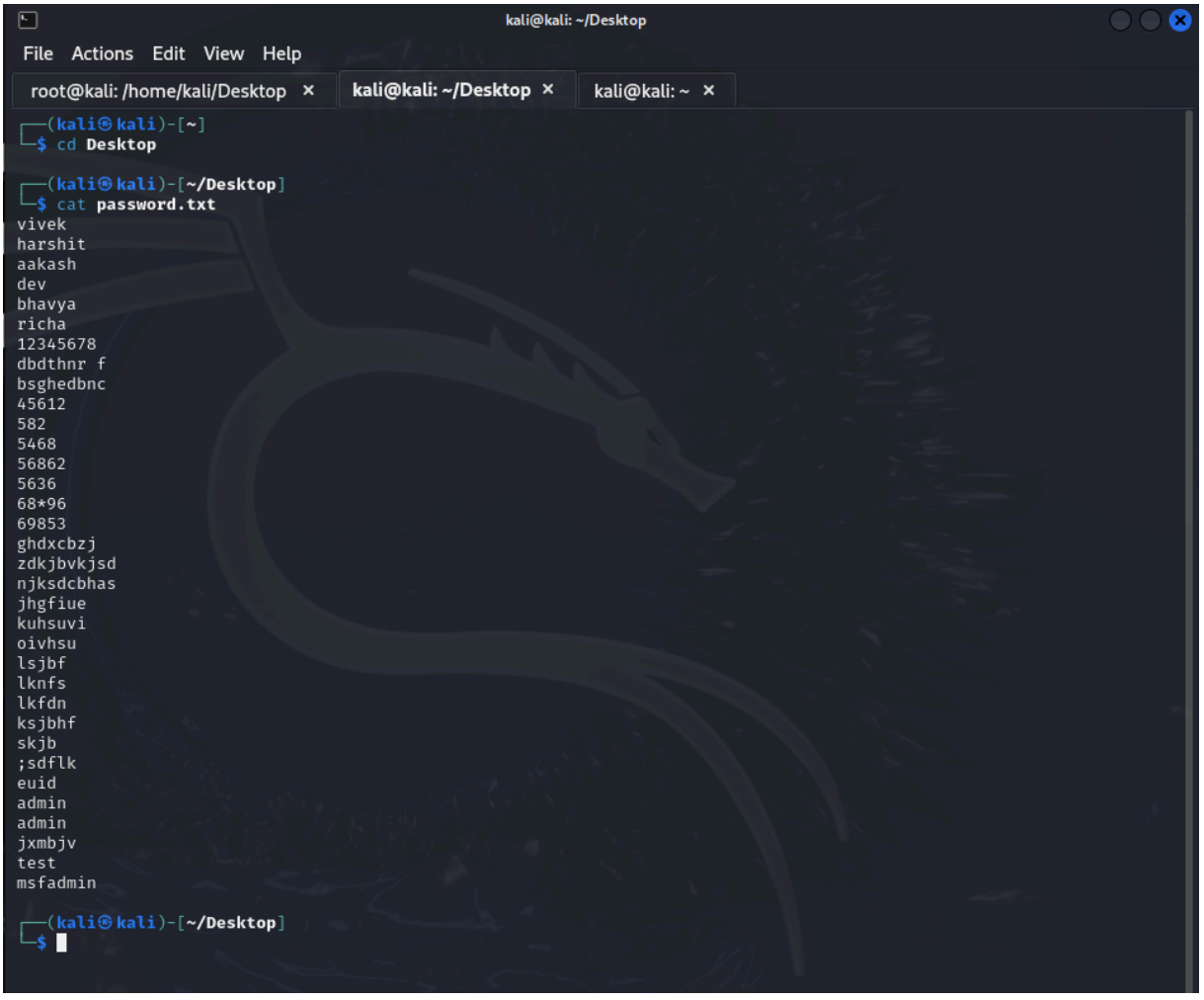


Step 7: We created our own wordlist on Desktop

> user.txt

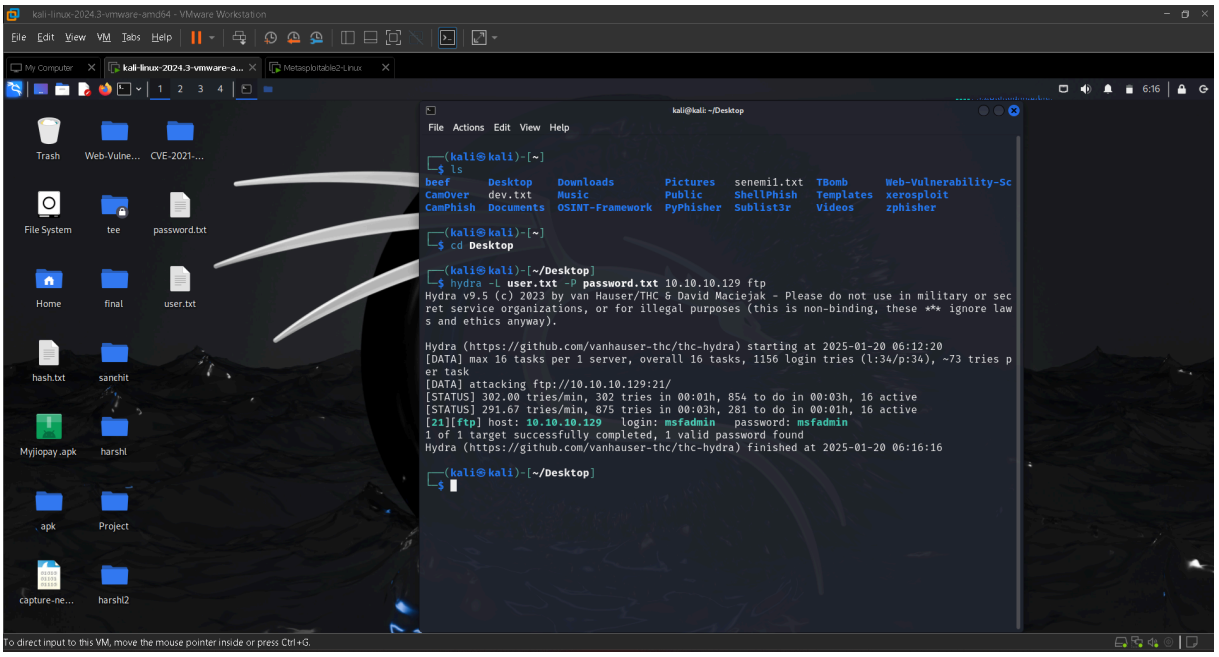
```
┌──(kali㉿kali)-[~/Desktop]
└─$ cat user.txt
vivek
harshit
aakash
dev
bhavya
richa
12345678
dbdthnr f
bsghedbnc
45612
582
5468
56862
5636
68*96
69853
ghdxcbzj
zdkjbvkjsd
njksdcbhas
jhgfiue
kuhsuvi
oivhsu
lsjbf
lknfs
lkfdn
ksjbhf
skjb
;sdflk
euid
admin
admin
jxmbjv
test
msfadmin

┌──(kali㉿kali)-[~/Desktop]
└─$
```

> password.txt



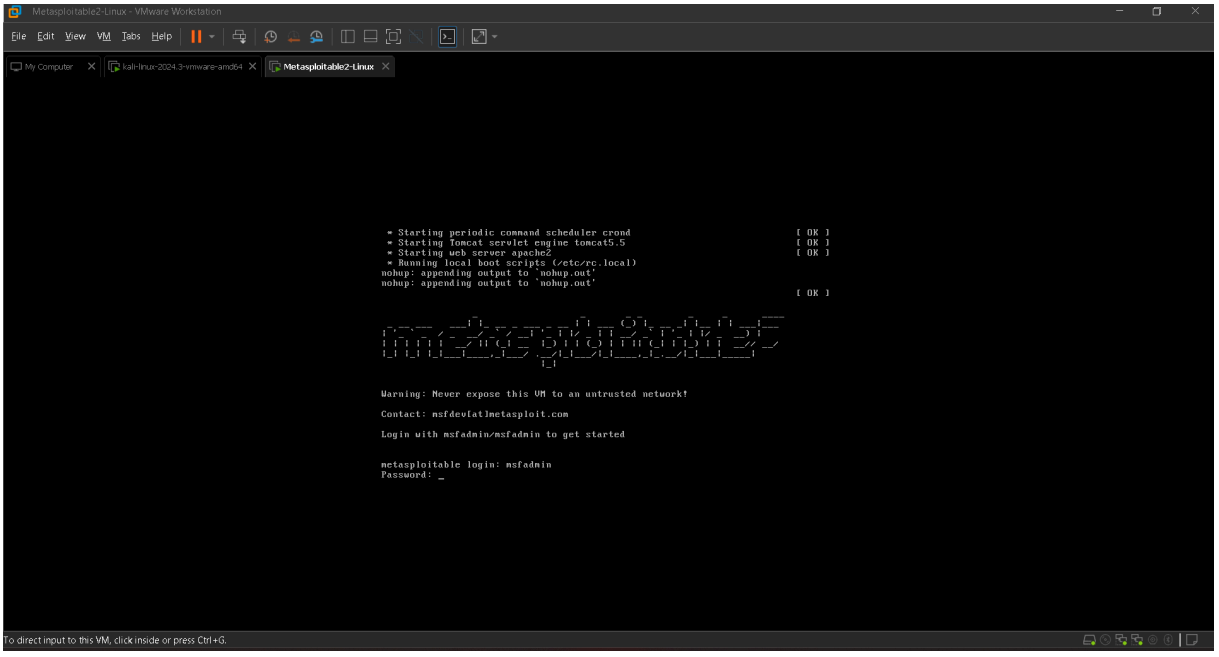
Step 8:We crack the user name and password using our own lists using command

> hydra -L user.txt -P password.txt 10.10.10.129 ftp



Step 9: After all that we see that successful password will be shown, which means we successfully cracked the password of the server.

Step 10: Use the cracked password for login into gds server and we see that we successfully login into gds server .To check you are login into gds server or not use command.



3.3 Identification

1.Using Kali Linux and Hydra for Attack Testing and Alert Generation.

2.Brute force attacks are an essential part of penetration testing, allowing security professionals to assess the strength of a system's passwords.

3. One popular tool is Hydra, an open-source login cracker that supports over 50 protocols.

Bibliography

[1] https://youtu.be/fQ0kMthJFMw?si=JJFDzXSZLLjH8uFH

[2] https://youtu.be/DElzWHWDG9Q?si=cqic8uHNg7m8KGKk [3]

https://www.techtarget.com/searchnetworking/definition/network-analyzer [4]

https://youtu.be/CSd7uXSZmA8?si=HH-fxeuPbZuDRV23 [5]

https://chat.openai.com/

[6] https://youtu.be/S9FdzDXgniA?si=gQM4IIiCsocbUMoc [7]

https://youtu.be/P7d-pPCQlyo?si=WVs3NSMawYZJQIV0 [8]

https://youtu.be/Hb214nXShkQ?si=6mwLqgmp3jsR2-E4