Air University Islamabad, Main Campus.

National Cyber Security Academy (NCSA)

Department of Cyber Security

Course name:

Introduction to Cyber Security

Semester Project

Submitted by:

Khadija Zia

Ayesha Mushtaq

Hasan Ahmed Mumtaz

OpenVAS Project

Contents

OpenVAS

OpenVAS (Open Vulnerability Assessment System) is an open-source tool for identifying security issues in networks. Part of the Greenbone Vulnerability Management (GVM) suite, it is widely used by cybersecurity professionals to assess vulnerabilities in systems and applications. It is a powerful tool for vulnerability scanning that identifies and evaluates potential weaknesses within a network.

1. Explanation of the Selected Tool:

1.1. Key Features:

OpenVAS offers a comprehensive vulnerability database that is regularly updated with information about known vulnerabilities, ensuring accurate and up-to-date assessments. It supports customizable scans, allowing users to target specific systems or services based on their needs. The tool generates detailed reports, highlighting detected vulnerabilities and their criticality, aiding in prioritization and remediation.

1.2. Purpose:

i. Identifying exploitable vulnerabilities in systems.
ii. Providing actionable insights to improve security posture.
iii. Supporting compliance with security standards and policies.

OpenVAS is widely utilized due to its robust features, open-source nature, and capability to uncover a broad spectrum of vulnerabilities. Key reasons include:

1. **Risk Mitigation:** By identifying vulnerabilities, organizations can implement controls to reduce the risk of exploitation.
2. **Comprehensive Scans:** It provides in-depth scans that detect issues across multiple layers of the network.
3. **Ease of Reporting:** Detailed reports help organizations prioritize remediation efforts effectively.
4. **Cost-Effectiveness:** As an open-source tool, it offers advanced vulnerability scanning capabilities without licensing costs.

1.3. Limitations:

While OpenVAS is a powerful tool, it has limitations such as being resource-intensive, often requiring significant processing power and memory. Scans may fail or timeout on large networks

due to misconfigurations or scale. Additionally, its initial setup can be complex for beginners, and successful scans depend on proper network configuration and access.

1.4. Advancements:

Recent advancements in OpenVAS have tackled several limitations, including an improved user interface for better usability and workflow management. The tool now features regular updates to its vulnerability database, ensuring coverage of the latest threats. Enhanced scalability optimizes performance on large networks and added API integration supports seamless connectivity with other security tools and frameworks.

2. Implementation of Risk Assessment:

Risk assessment was conducted using **OpenVAS on Kali Linux** to identify vulnerabilities in **Metasploitable2**, a purposely vulnerable virtual machine. Both systems were deployed in a controlled lab environment using VirtualBox to simulate a real-world network scenario.

2.1. Purpose of the Task:

The objectives of this task were:

i. To scan the Metasploitable2 system for known vulnerabilities using OpenVAS.
ii. To analyze detected vulnerabilities based on severity and risk level.
iii. To generate a vulnerability assessment report for learning and evaluation purposes.

2.2. Requirements:

i. Kali Linux (Attacker / Scanner Machine)
ii. OpenVAS (GVM)
iii. Metasploitable2 (Target Machine)
iv. VirtualBox (Virtualization Platform)

2.3. Tool and System Versions:

The following tools and systems were used during the vulnerability assessment:

- **Kali Linux:** Kali Linux 2025.x (64-bit)
- **OpenVAS (GVM):** Greenbone Vulnerability Management (GVM) version 22.x
- **Metasploitable2:** Metasploitable2 Linux (Rapid7 vulnerable VM)
- **VirtualBox:** Oracle VM VirtualBox version 7.2.4x

These versions ensured compatibility between the scanning tool and the target system and provided a stable laboratory environment for vulnerability assessment.

3. Setting Up the Environment:

Two virtual machines were configured on VirtualBox:

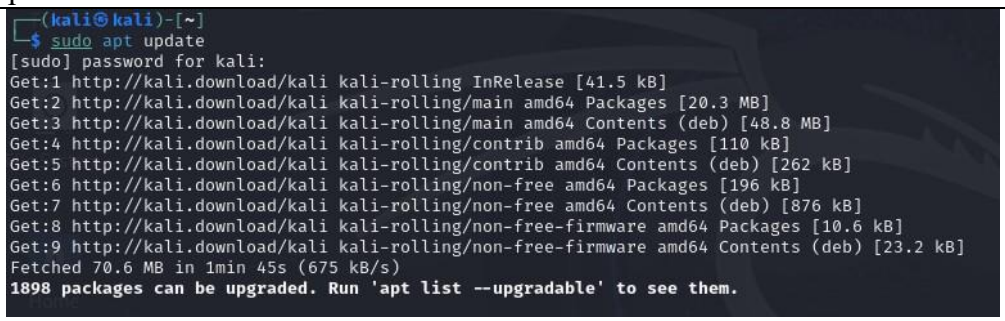
i. **Kali Linux VM** – Used to install and run OpenVAS.
ii. **Metasploitable2 VM** – Used as the vulnerable target system.

Both VMs were connected using a **Host-Only Adapter network**, enabling direct communication between them while isolating them from external networks. This configuration ensured a controlled and secure laboratory environment for vulnerability assessment.

3.1. Installation and Configuration of OpenVAS:

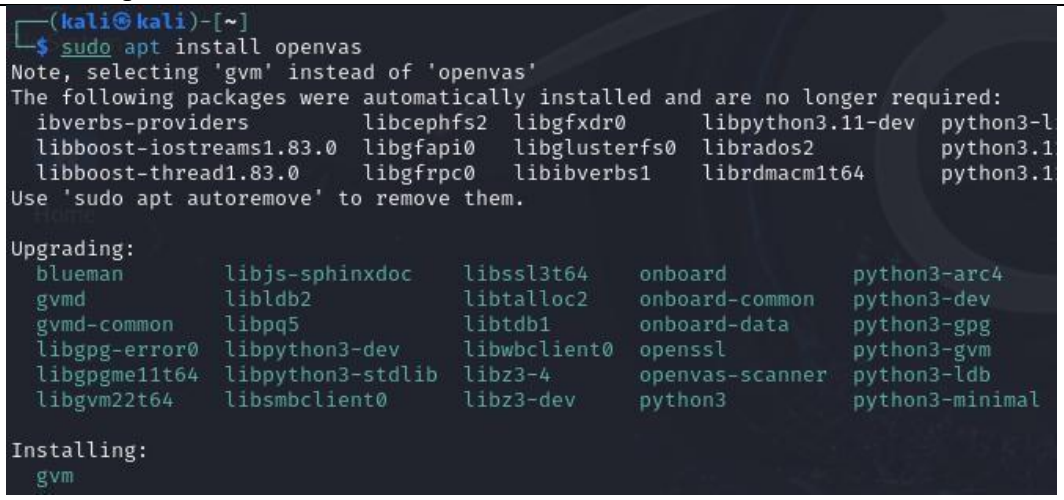
OpenVAS was installed on the Kali Linux virtual machine following system updates

```
sudo apt update
```

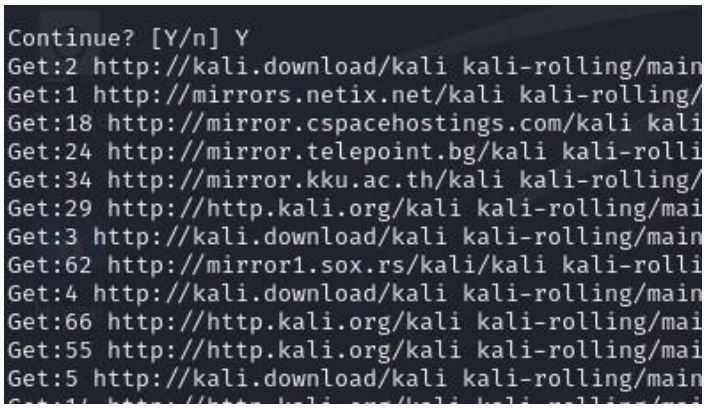


Install OpenVAS using this command

```
sudo apt install openvas
```



Most probably you will see the message asking you to select **gvm** instead of **openvas.** Options will appear write Y to continue.

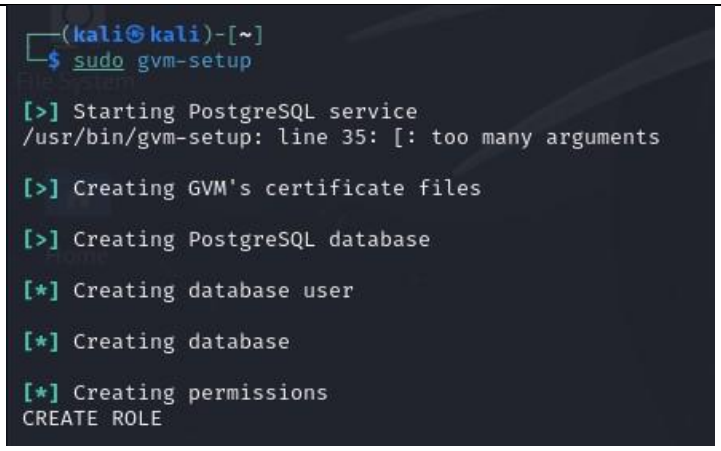


This will take some time. After this you will see an error message when you run gvm setup. Run setup using this command

```
sudo gvm-setup
```

This gvm command is used to run the setup.

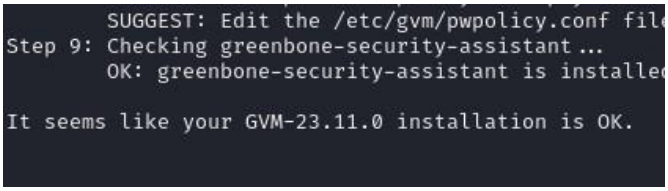
```
sudo gvm-setup
```



If everything went well, you would get this message. This process will take a lot of time. When all the desired files and configurations are installed, now it's time to check the setup of gvm. To do this, run this command:

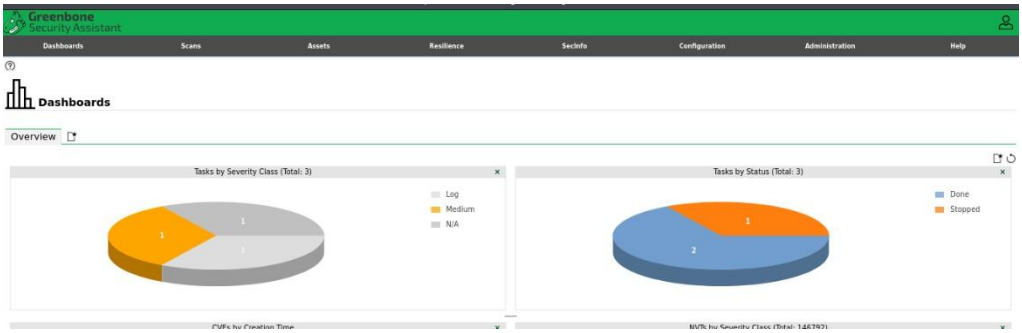
```
sudo gvm-check-setup
```

If everything went well, this message would display,



When you run the check setup command, check the portion where you get the local host address which is **127.0.0.1** and your password. Username is always **admin.** Copy the address and paste it on your browser, and this window will appear,

Enter the username and password, and Dashboard will appear.



Before starting your task, go to the configuration tab and check whether configurations are present or not. If yes then so worries, but if not then enter these three commands one by one

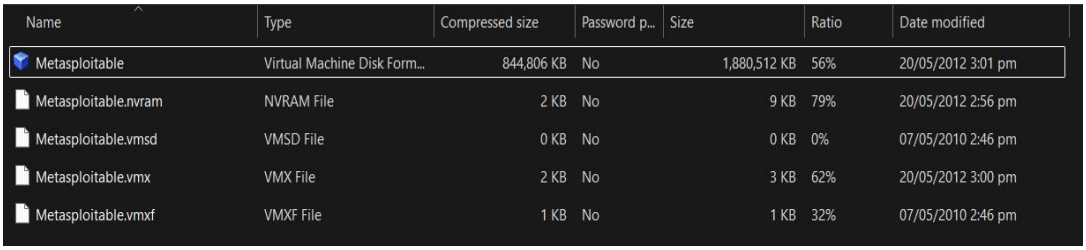
greenbone-feed-sync --type GVMD_DATA
greenbone-feed-sync --type SCAP
greenbone-feed-sync --type CERT

After a long wait, you will see configurations in configurations portion.

Summary of OpenVAS Setup on Kali Linux:
OpenVAS was installed on the Kali Linux machine by updating the system and installing the required packages. After installation, the GVM setup process was completed, and all necessary services were verified using the built-in setup check command. The OpenVAS web interface was accessed through the local host address provided during setup.

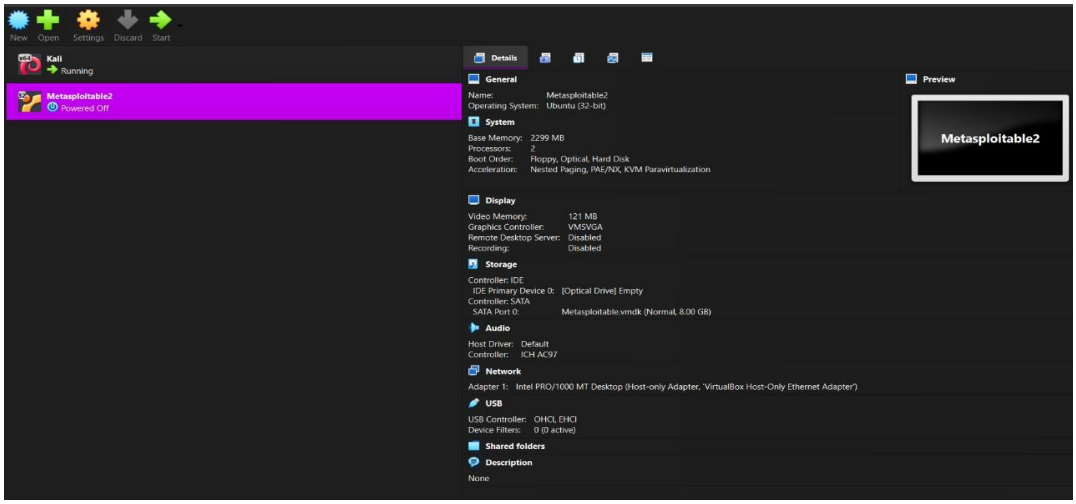
3.2. Metasploitable2(Target) Setup:

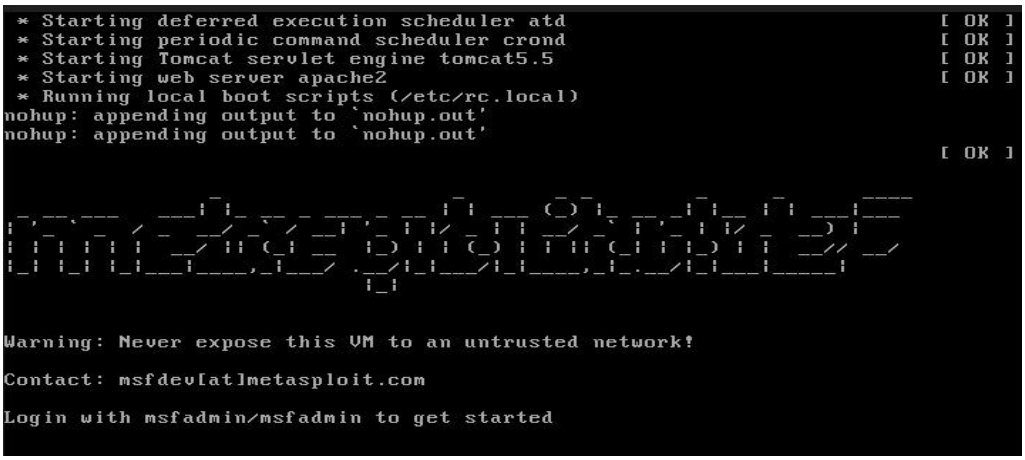
Metasploitable2 was installed using Rapid7.



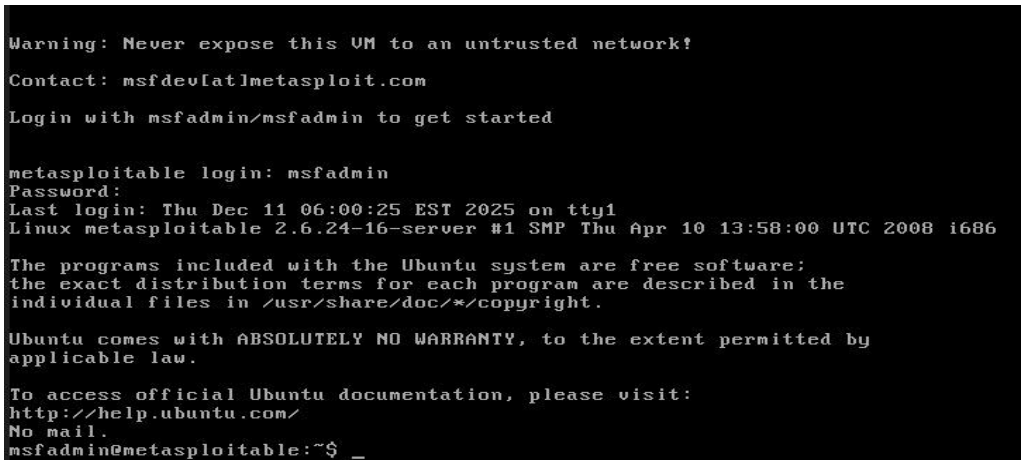
Name	Type	Compressed size	Password p...	Size	Ratio	Date modified
Metasploitable	Virtual Machine Disk Form...	844,806 KB	No	1,880,512 KB	56%	20/05/2012 3:01 pm
Metasploitable.nvram	NVRAM File	2 KB	No	9 KB	79%	20/05/2012 2:56 pm
Metasploitable.vmsd	VMSD File	0 KB	No	0 KB	0%	07/05/2010 2:46 pm
Metasploitable.vmx	VMX File	2 KB	No	3 KB	62%	20/05/2012 3:00 pm
Metasploitable.vmxf	VMXF File	1 KB	No	1 KB	32%	07/05/2010 2:46 pm

Metasploitable2 was set up on Virtual Box.



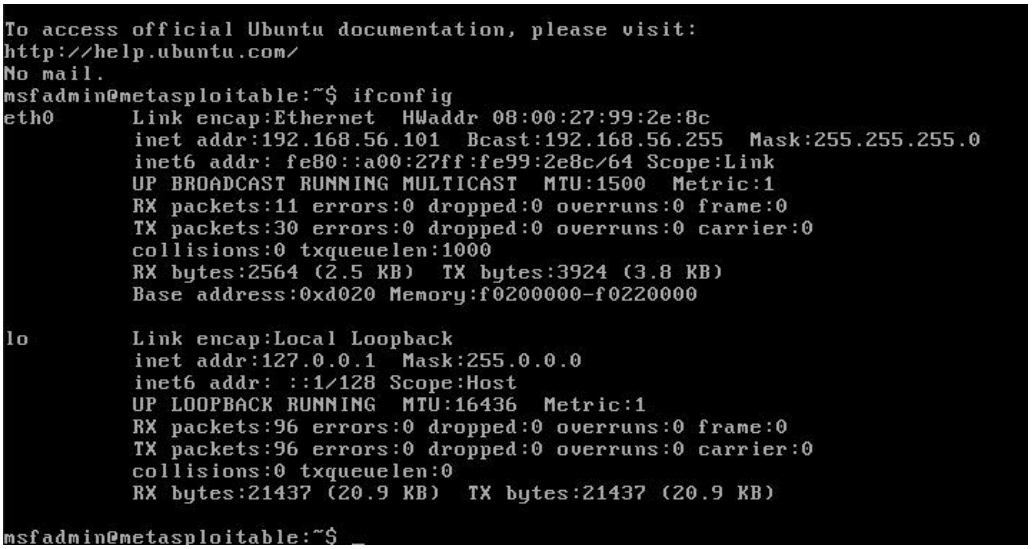


Metasploitable2 was logged in after entering the "**msfadmin**", the admin ID.



Now checking for the IP of Metasploitable2 using "ifconfig" command.

ifconfig



The IP address of the Metasploitable2 machine was identified (192.168.56.101) using network configuration command.
This IP address was then added manually as a target in OpenVAS under the **Targets** configuration section.

3.3. Network Configuration Verification:

Before performing connectivity tests and scans, it was verified that both VMs were on the **same host-only network**:

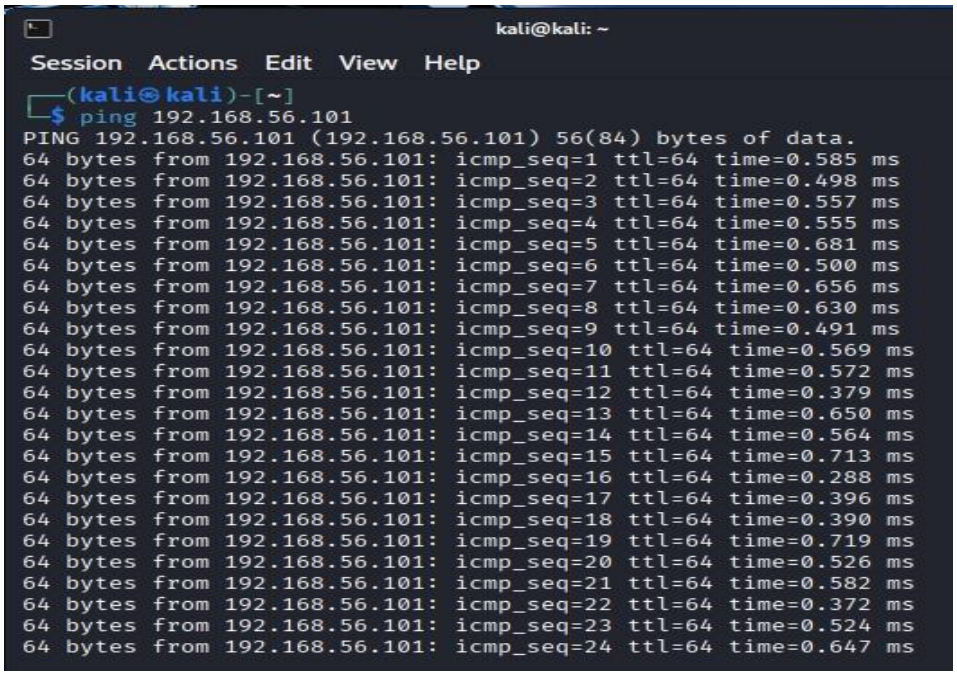
- VirtualBox network settings were checked to confirm both machines used the **same Host-Only Adapter**.
- IP addresses were verified using ifconfig (Metasploitable2) and ip a (Kali Linux) to ensure both systems were on the **same subnet**.

This configuration allowed proper communication between Kali Linux and Metasploitable2, making them ready for vulnerability scanning using OpenVAS

3.4. Connectivity Verification (Reachability Test):

Ping Test:
The ping command was used from the Kali Linux machine to verify basic network connectivity with the Metasploitable2 target system.

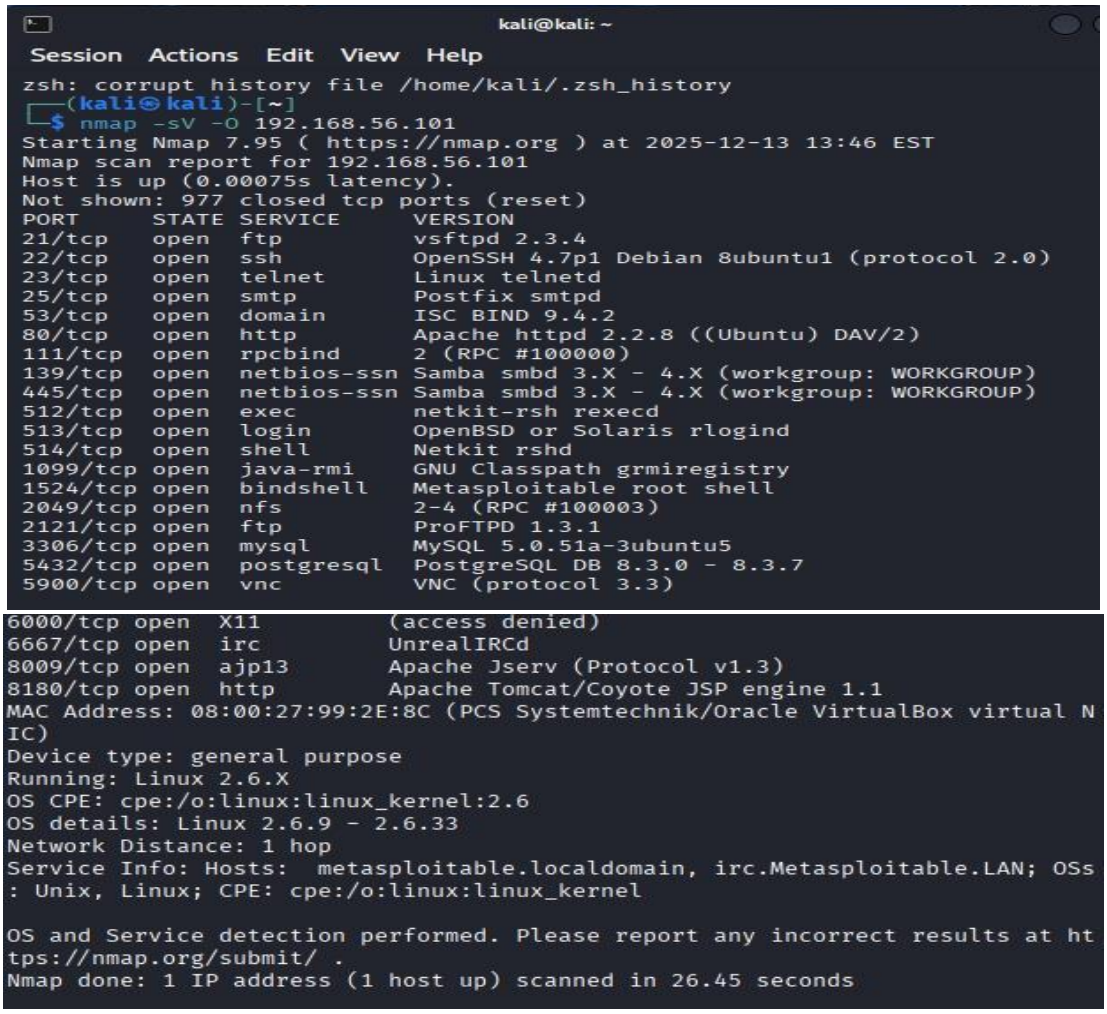
```
ping 192.168.56.101
```



Result:
The Metasploitable2 machine responded successfully with ICMP replies and acceptable response times. This confirmed that the target system was online and reachable from the Kali Linux machine.

Nmap Scan:
The nmap tool was used to identify open ports and active services running on the Metasploitable2 system.

```
nmap -sV -O 192.168.56.101
```



Result:
The scan revealed multiple open ports and running services, including vulnerable services such as FTP, Telnet, HTTP, and SMB. The presence of these open ports confirmed that the Metasploitable2 system was accessible and suitable for vulnerability assessment using OpenVAS.

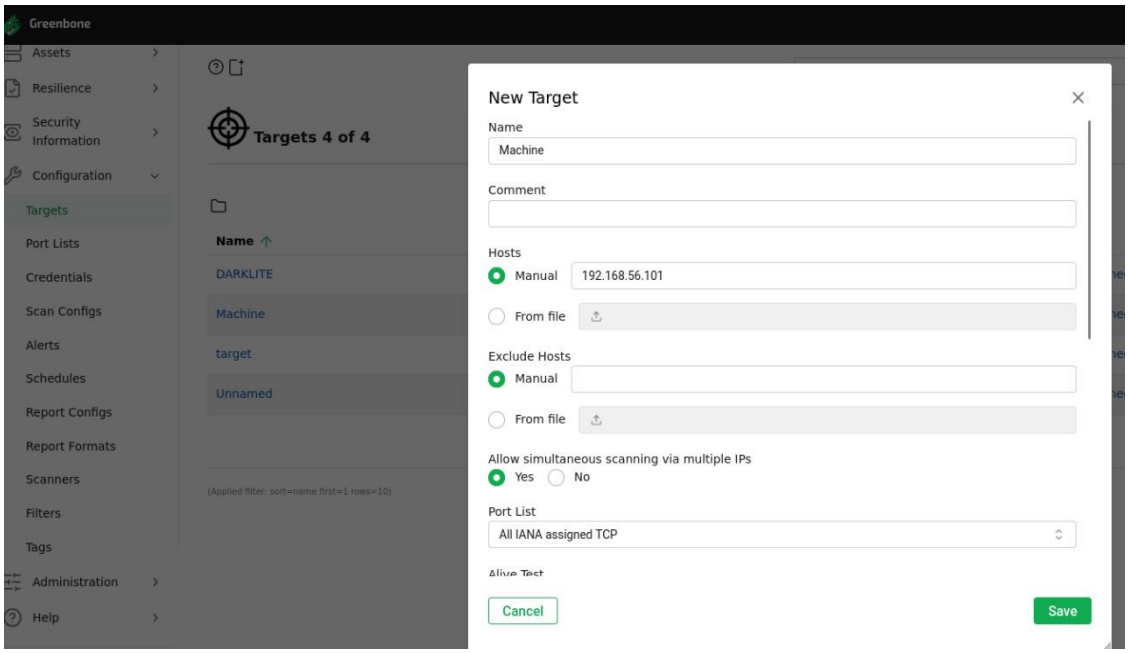
3.5. Target Configuration:

To configure the target, the **Configuration → Targets** option was selected, and a new target was created.

The identified IP address (192.168.56.101) of the Metasploitable2 machine was entered in the target field and the target was names as “Machine”.

Default port ranges and scan settings were used to ensure a comprehensive vulnerability assessment.

Once saved, the target became available for selection during scan task creation. This configuration enabled OpenVAS to correctly identify and assess vulnerabilities present on the Metasploitable2 system.



3.6. Assessment of Risk:

The objective of the assessment was to identify vulnerabilities in Metasploitable2 and evaluate associated risks. The assessment included:

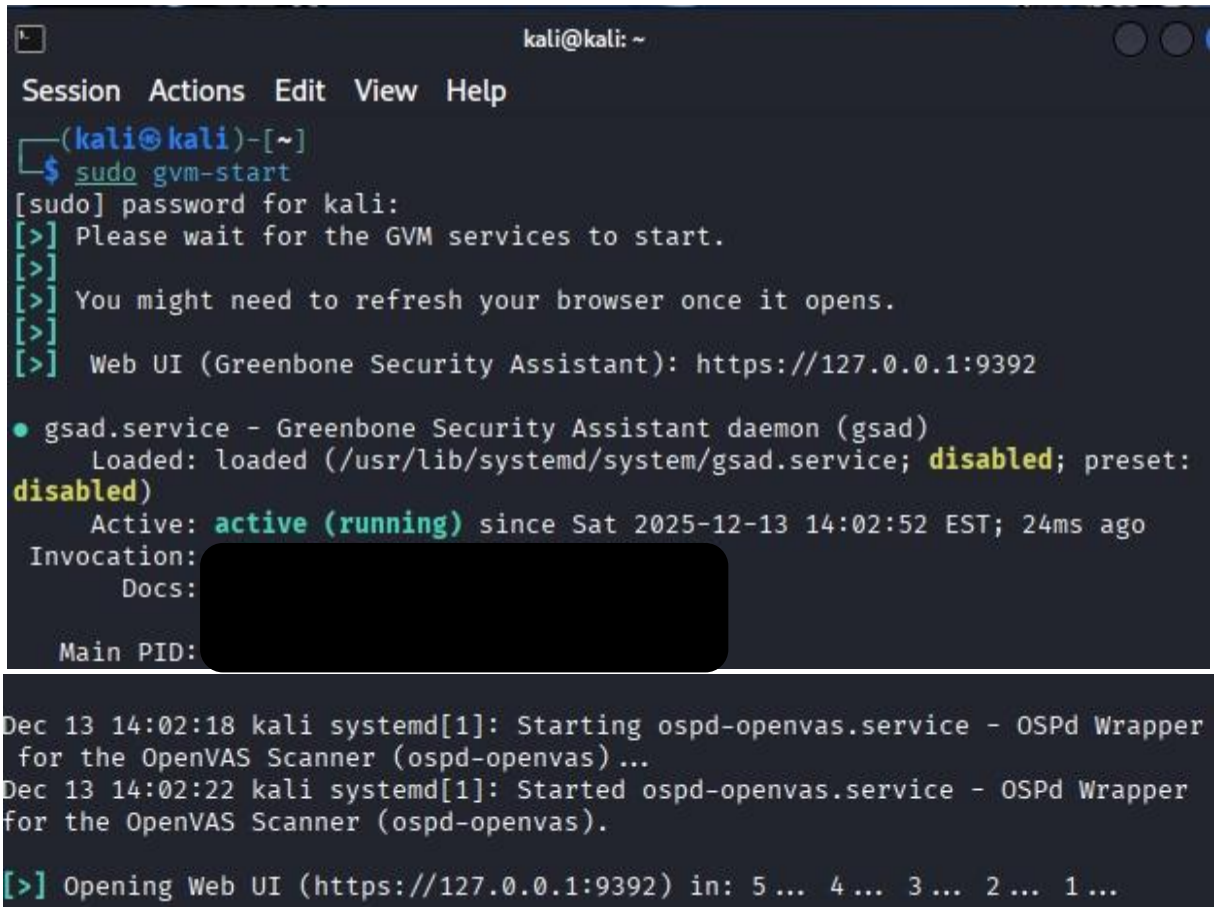
- Scanning all open services on Metasploitable2 using OpenVAS.
- Detecting outdated, misconfigured, or vulnerable services (FTP, Telnet, Samba, HTTP).
- Categorizing vulnerabilities based on severity (Critical, High, Medium, Low).
- Generating actionable insights for risk mitigation and security improvements.

This step established the **risk context** and justified the need for a vulnerability assessment.

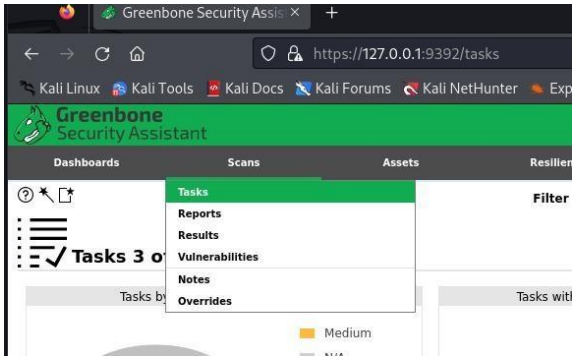
3.7. Vulnerability Scanning Process:

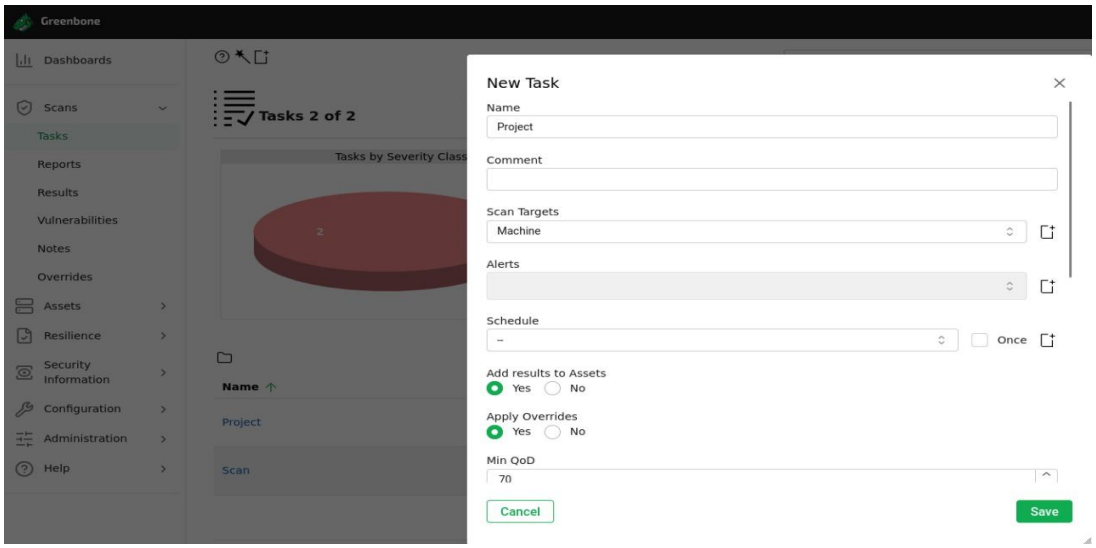
Starting the vulnerability process after running both machines on Virtual Box.
On Linux, using the command to start OpenVAS.

```
sudo gvm-start
```

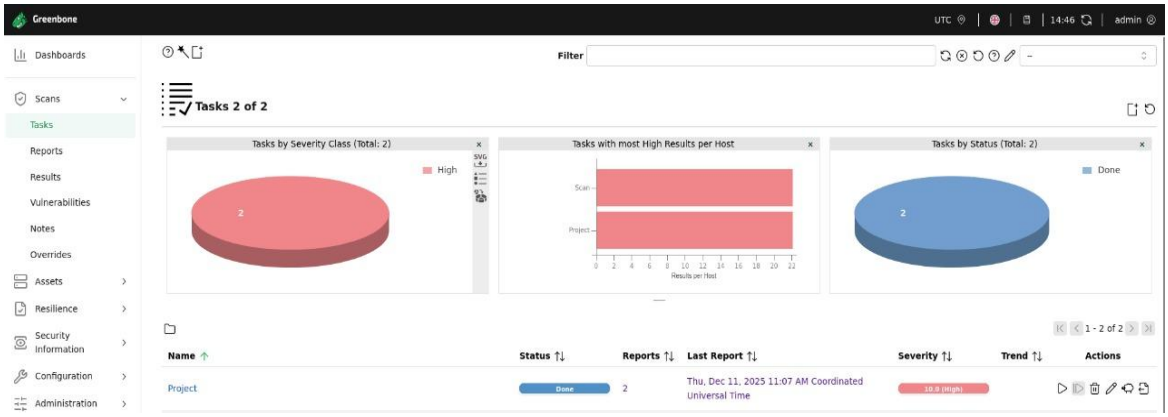


The command resulted in opening of Webpage of Greenbone Vulnerability Assessment.
Access OpenVAS web interface.
Create a new scan task using **Wizard**





In Scans, we had named the scanning procedure as "**Project**" and put the target the Metasploitable2(Machine).
Then save the Project and start the scanning.



The scan ran successfully and gave the required output.

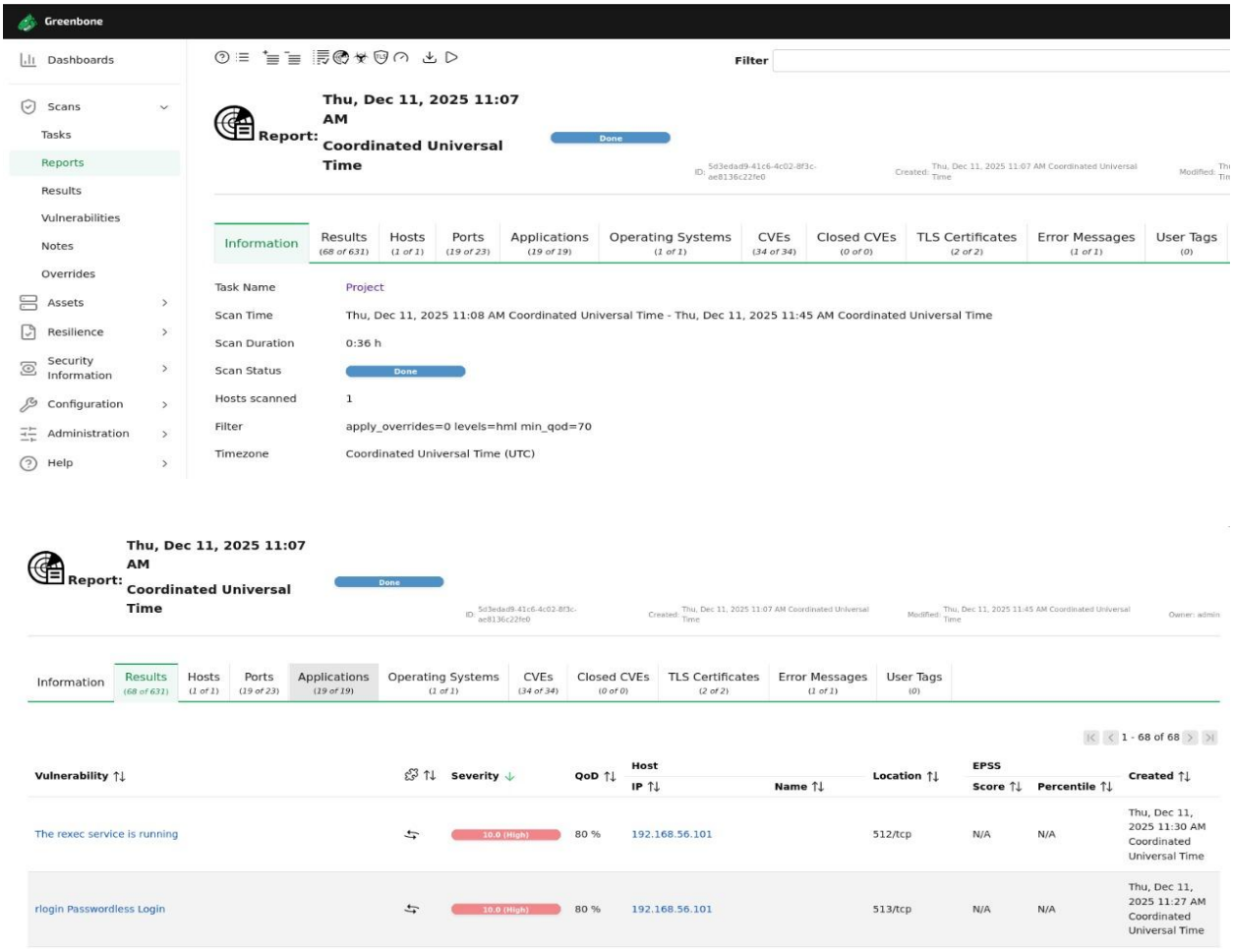
Summary:

The vulnerability assessment was performed using the following steps:

- The OpenVAS web interface was accessed from the Kali Linux browser.
- A new scan task was created using the Wizard option.
- The IP address of the Metasploitable2 machine was entered as the scan target.
- The scan was initiated, and OpenVAS performed a comprehensive vulnerability assessment.
- The scan status was monitored until completion.

3.8. Results and Findings:

After completion of the scan, the results revealed multiple vulnerabilities in the Metasploitable2 system.



These included:

- Outdated and insecure services such as FTP, Telnet, and Samba.
- Multiple open ports exposing vulnerable services.
- Critical and high-severity vulnerabilities associated with known CVEs.

The results demonstrated that Metasploitable2 is highly vulnerable and unsuitable for use in a production environment.

3.9. Report Generation:

OpenVAS generated a detailed vulnerability report categorizing issues by severity. The report was exported in **PDF format** for analysis and documentation. This report provided valuable insights into the vulnerabilities present and their potential impact.

4. Challenges Faced and Solutions:

During the installation and use of OpenVAS on Kali Linux, several challenges were encountered:

4.1. Installation Issues:

- **Problem:** Initial attempts to install OpenVAS failed due to missing dependencies and misconfigured repositories.
- **Solution:** Manually added and updated repositories, followed by troubleshooting missing packages.

4.2. Timeout Errors:

- **Problem:** After successfully running one scan, subsequent scans resulted in timeout errors.
- **Solution:** Reinstalled OpenVAS and manually adjusted the network configuration to ensure proper connectivity.

4.3. Configuration Complexity:

- **Problem:** The tool required extensive configuration, including setting up services and defining proper network parameters.
- **Solution:** Consulted documentation and forums to manually configure the network and services.

4.4. Resource Limitations:

- **Problem:** Scans were resource-intensive, slowing down the system and causing delays.
- **Solution:** Optimized scan settings by reducing the number of simultaneous checks and disabling non-critical features.

4.5. Lack of Documentation:

- **Problem:** The lack of detailed, step-by-step documentation for resolving specific issues added to the complexity.
- **Solution:** Relied on community forums and trial-and-error methods to resolve issues.

4.6. PDF Export Challenge:

- **Problem:** Exporting the vulnerability assessment report as a PDF from the virtualized Kali Linux environment to Windows was initially difficult. The PDF was generated in Kali but not automatically accessible in Windows.
- **Solution:** The report was copied from the Kali Desktop to a VirtualBox shared folder (/media/sf_linux), making it accessible on the Windows host system. This ensured the report could be submitted and reviewed without issues.

5. Contribution Table:

S#	Names	Contribution
1	Khadija Zia	Tool Research, setup, scanning & report
2	Ayesha Mushtaq	Tool Research, OpenVAS theory & scanning
3	Hasan Ahmed Mumtaz	Network setup & scanning support

6. Conclusion:

This project successfully demonstrated the use of OpenVAS on Kali Linux to detect vulnerabilities in the Metasploitable2 system. The assessment highlighted numerous critical security weaknesses, emphasizing the importance of vulnerability scanning in identifying and mitigating security risks. OpenVAS proved to be an effective and reliable tool for vulnerability assessment within a controlled laboratory environment.

7. References:

[1] Greenbone Networks, *Greenbone Vulnerability Management Documentation*. [Online]. Available: https://greenbone.github.io/docs/

[2] Kali Linux, *Official Kali Linux Documentation*. [Online]. Available: https://www.kali.org/docs/

[3] Rapid7, *Metasploitable2 Documentation*. [Online]. Available: https://docs.rapid7.com/metasploit/metasploitable-2/

[4] Oracle Corporation, *Oracle VM VirtualBox User Manual*. [Online]. Available: https://www.virtualbox.org/manual/

.....