

Greenbone Security Manager (GSM) and Metasploitable

Overview

This project demonstrates practical experience in setting up and performing vulnerability assessments in a controlled lab environment. The lab focused on deploying **Greenbone Security Manager (GSM)**, configuring a virtualized network environment, and scanning a deliberately vulnerable machine (**Metasploitable**) to identify and analyze security weaknesses.

This exercise replicates real-world vulnerability management workflows commonly used in enterprise security operations.

Objectives

- Install and configure **Greenbone Security Manager Trial** on VirtualBox.
 - Run and manage multiple virtual machines simultaneously within a secure, isolated network.
 - Perform **Nmap** and **OpenVAS/GSM vulnerability scans** against the Metasploitable target.
 - Analyze scan results and verify detected vulnerabilities.
-

Environment Setup

This lab required running the following virtual machines concurrently:

1. **Outside-Firewall** – Provided internet connectivity and DNS resolution.
 2. **Client2-DMZ** – Used as a management workstation to initiate scans.
 3. **Greenbone Security Manager (GSM)** – Hosted the OpenVAS vulnerability scanning engine.
 4. **Metasploitable** – A vulnerable Linux system serving as the target host.
-

Configuration Steps

1. Import and Configure GSM Virtual Machine in VirtualBox

- Downloaded the **GSM Trial OVA** file from Greenbone's official site.
- Opened VirtualBox → **File** → **Import Appliance**, selected the .ova file (GSM-TRIAL-21.04.5-VirtualBox.ova) → Clicked **Import**.
- After import completed, updated GSM VM settings as follows:

- **Display → Graphics Controller:** VMSVGA
- **System → Processor:** Reduced to 1 CPU core
- **System → Motherboard:** Set Base Memory to 3072 MB
- **Network → Adapter 1:**
 - Enabled Adapter
 - Attached to: Internal Network
 - Name: DMZ

This configuration ensured that GSM operated within the same VLAN as other DMZ systems.

2. Start GSM and Configure Networking

- Powered on the GSM virtual machine while **Outside-Firewall** was active.
- Logged in using default credentials:
 - Username: admin
 - Password: admin

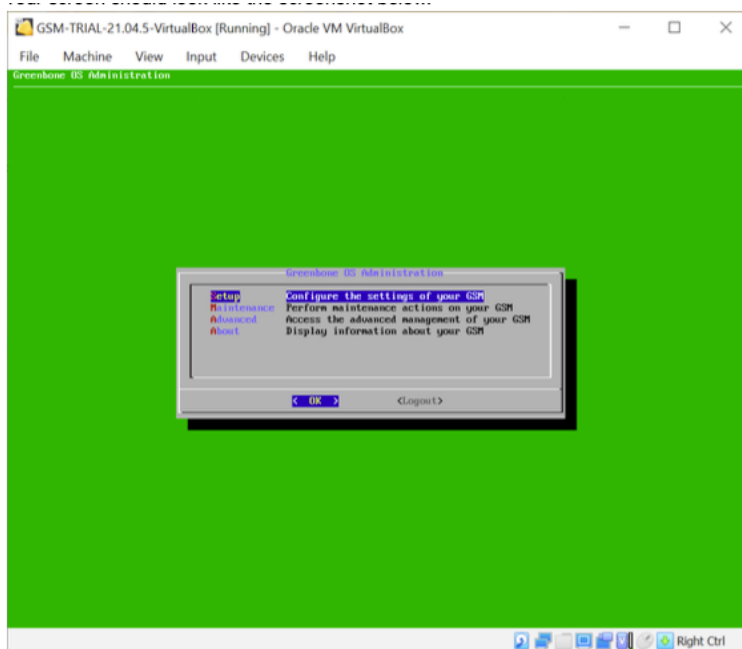


Figure 1/GSM

- In the GSM console, navigated to **Setup → Networking → Interfaces**.
- Enabled **IPv4** and **Static IP**, then entered:
 - IP Address: 192.168.4.80/24

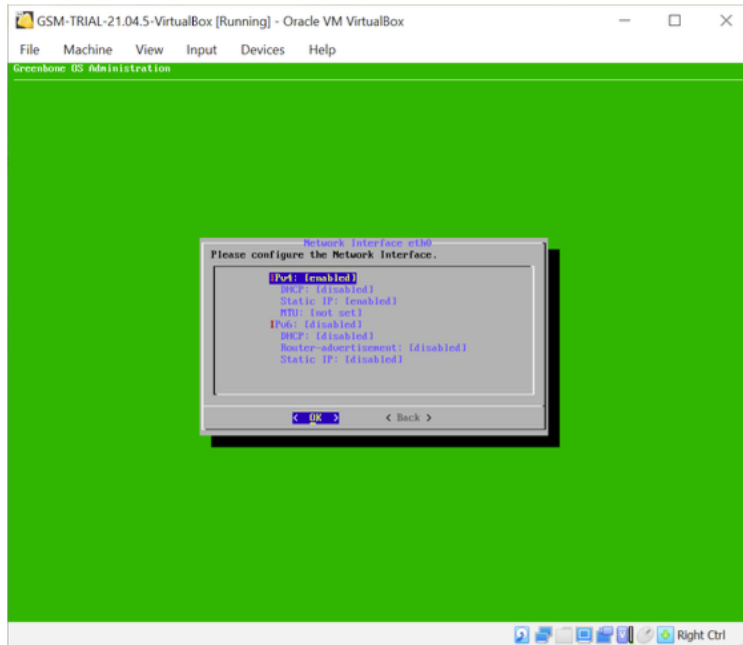


Figure 2/Networking

- Configured **DNS** under **Networking** → **DNS**:
 - DNS Servers: 8.8.8.8, 1.1.1.1, 192.168.4.1

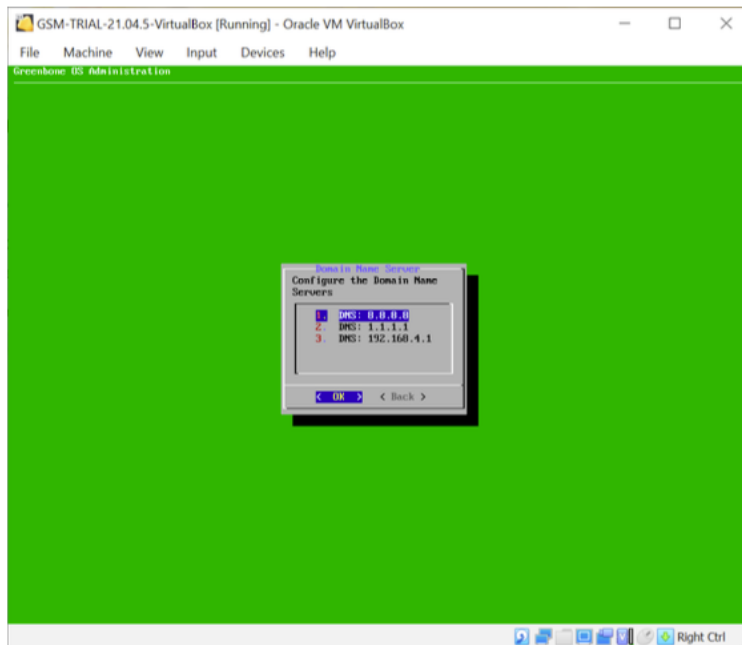


Figure 3/Dns

- Set the **Global Gateway** to:
 - 192.168.4.1 (Outside-Firewall)

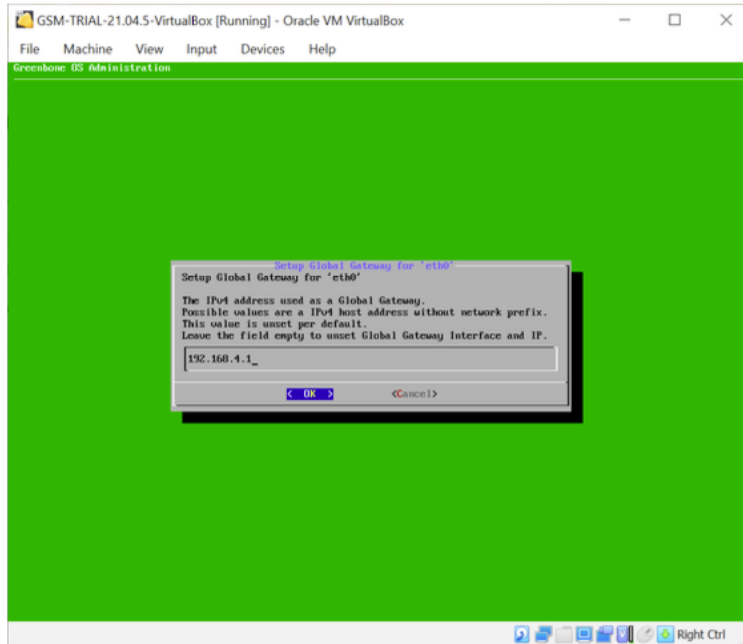


Figure 4/Gateway

- Navigated to **User and Password Management** → **Manage Web Users**, created a new global admin account for GSM web access.
- Noted the web interface URL:
 - <https://192.168.4.80>

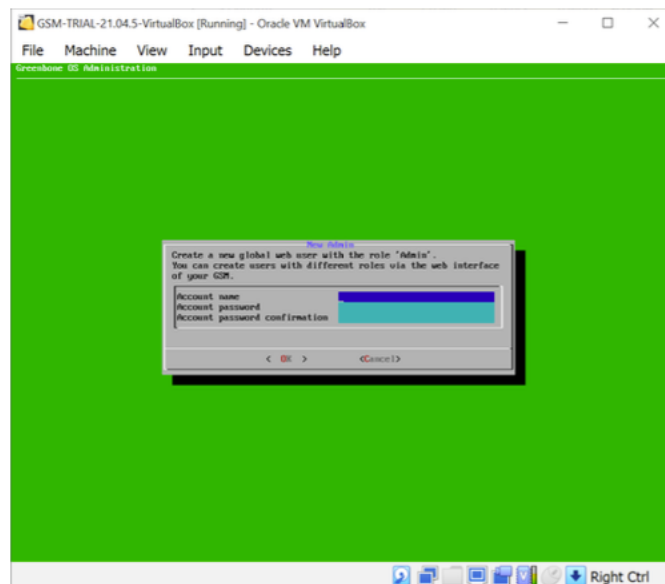


Figure 5/Users

3. Import and Configure Metasploitable Virtual Machine

- Downloaded **Metasploitable 2** from SourceForge (<https://sourceforge.net/projects/metasploitable/>).
 - Extracted the ZIP file to reveal Metasploitable.vmdk.
 - In VirtualBox, created a new VM named “Metasploitable”:
 - Type: Linux
 - Version: Ubuntu (32-bit)
 - Selected **Use an existing virtual hard disk file** → Browsed to Metasploitable.vmdk.
 - Configured the network:
 - **Adapter 1** → **Attached to:** Internal Network
 - **Name:** DMZ
-

4. Configure Static IP on Metasploitable

- Started the Metasploitable VM.
- Logged in with:
 - Username: msfadmin
 - Password: msfadmin
- Assigned a static IP:
- `sudo ifconfig eth0 192.168.4.85 netmask 255.255.255.0 up`
- `sudo route add default gw 192.168.4.1`
- Verified connectivity:
- `ping -c 4 192.168.4.1`

Successful replies confirmed connection to the Outside-Firewall.

5. Verify Connectivity Between Client2-DMZ, GSM, and Metasploitable

- Started **Client2-DMZ** virtual machine.
 - Verified connectivity to all systems in the DMZ VLAN:
 - `ping -c 4 192.168.4.85 # Metasploitable`
 - `ping -c 4 192.168.4.80 # GSM`
 - `ping -c 4 192.168.4.1 # Firewall Gateway`
 - All responses were successful, confirming proper DMZ network configuration.
-

6. Conduct Nmap Reconnaissance Scan

- Opened Terminal on **Client2-DMZ** and executed:
- `nmap -sS -sV -A -T4 192.168.4.85 -oN ~/nmap-metasploitable.txt`
 - -sS: Stealth SYN scan

- -sV: Service version detection
 - -A: OS and service script detection
 - -T4: Aggressive timing for faster results
- Saved the results as nmap-metasploitable.txt.
- The output revealed multiple open services such as:
 - FTP (21/tcp)
 - SSH (22/tcp)
 - Telnet (23/tcp)
 - HTTP (80/tcp)
 - MySQL (3306/tcp)
 - SMB (445/tcp)

```

4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 0.661/1.863/2.877/0.811 ms
sam@Client1: $ nmap 192.168.4.85
Starting Nmap 7.80 ( https://nmap.org ) at 2024-03-19 13:24 EDT
Nmap scan report for 192.168.4.85
Host is up (0.0028s latency).
Not shown: 977 closed ports
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
5900/tcp  open  vnc
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 0.14 seconds
sam@Client1: $

```

Figure 6/Nmap

7. Perform GSM Vulnerability Scan Against Metasploitable

- Opened Firefox on **Client2-DMZ** → navigated to <https://192.168.4.80>.
- Accepted the browser's SSL certificate warning (self-signed certificate).

- Logged into GSM web interface using the admin credentials.
- Navigated to **Scans** → **Tasks** → **Task Wizard**.

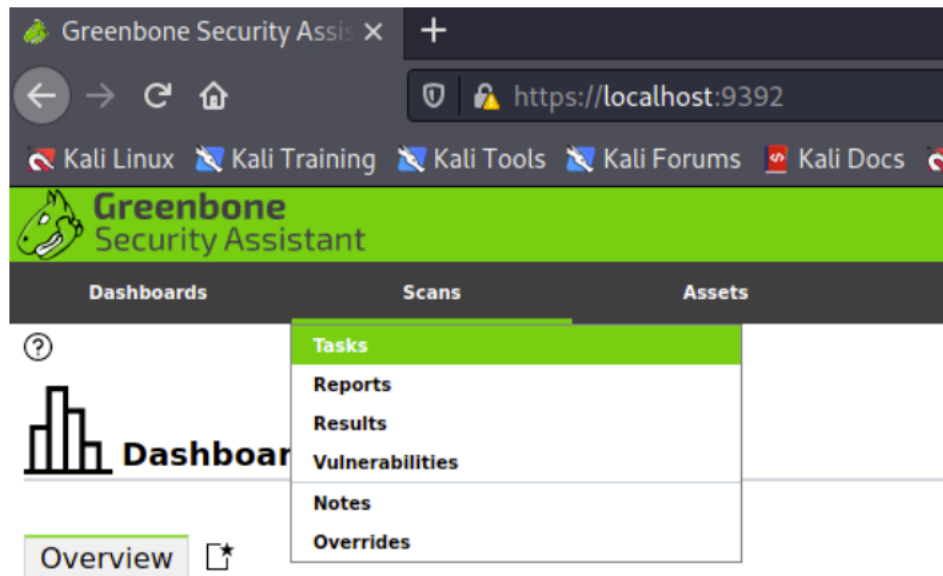


Figure 7/Tasks

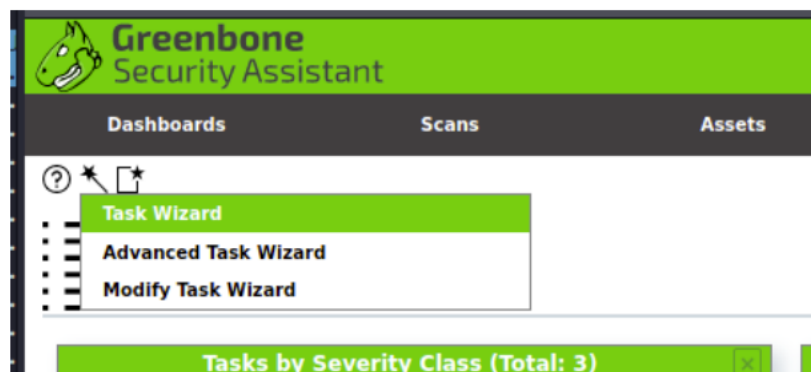


Figure 8/Task Wizard

- Entered the target IP: 192.168.4.85 and selected **Full and Fast** scan configuration.

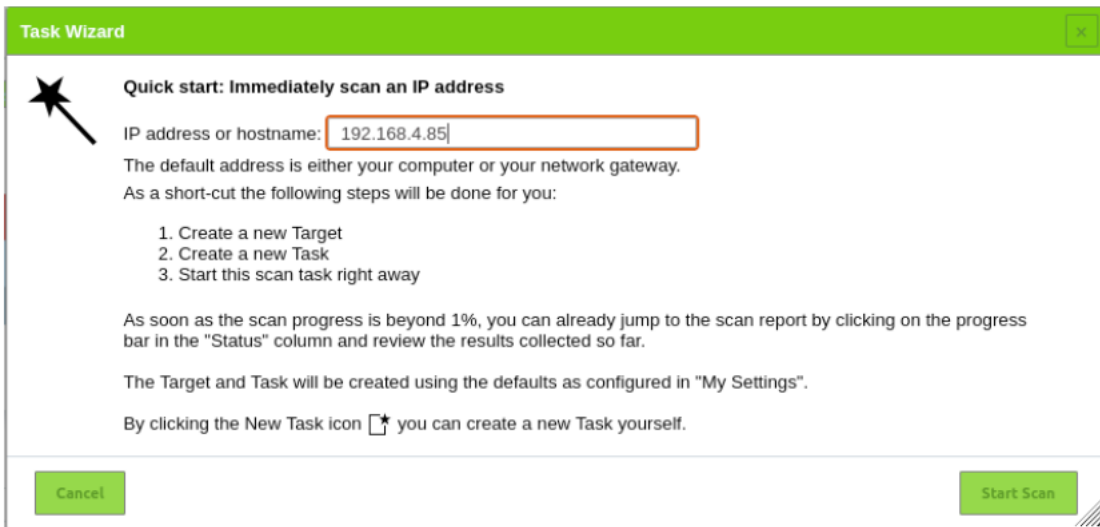


Figure 9/Scan

- Clicked **Start Scan** to begin the vulnerability assessment.

The scan transitioned through stages:

- *Requested* → *Queued* → *Running* → *Done*
and took approximately 30 minutes to complete.

8. Analyze GSM Scan Results

- After completion, clicked **Done** → **Report** → **Results**.
- The scan identified several high-severity vulnerabilities, including:
 - Outdated services (e.g., Apache, Samba, MySQL).
 - Known CVEs associated with remote code execution and privilege escalation.
 - Weak SSH and Telnet configurations.
- Exported the results as an HTML report and saved screenshots of:
 - The completed scan summary.
 - The high-severity findings list.

Client2-dmz [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Firefox Web Browser Mar 19 14:52

pfSense-outside.home.ar x YouTube x Settings x Greenbone Enterprise Appliance x

https://192.168.4.80/report/84aeddcf-f675-439a-8b26-863ee3c3db88

Greenbone Enterprise Appliance

Dashboards Scans Assets Resilience SecInfo Configuration Administration Help

Filter

Report Tue, Mar 19, 2024 5:29 PM UTC Done ID: 84aeddcf-f675-439a-8b26-863ee3c3db88 Created: Tue, Mar 19, 2024 5:29 PM UTC Modified: Tue, Mar 19, 2024 6:07 PM UTC Owner: sam

Information Results (69 of 595) Hosts (1 of 1) Ports (19 of 23) Applications (15 of 15) Operating Systems (1 of 1) CVEs (35 of 35) Closed CVEs (0 of 0) TLS Certificates (2 of 2) Error Messages (0 of 0) User Tags (0)

1 - 69 of 69

Vulnerability	Severity	QoD	Host IP	Name	Location	Created
Possible Backdoor: Ingreslock	10.0 (High)	99 %	192.168.4.85		1524/tcp	Tue, Mar 19, 2024 5:56 PM UTC
rlogin Passwordless Login	10.0 (High)	80 %	192.168.4.85		513/tcp	Tue, Mar 19, 2024 5:44 PM UTC
Twiki XSS and Command Execution Vulnerabilities	10.0 (High)	80 %	192.168.4.85		80/tcp	Tue, Mar 19, 2024 5:50 PM UTC
The rexec service is running	10.0 (High)	80 %	192.168.4.85		512/tcp	Tue, Mar 19, 2024 5:48 PM UTC
Distributed Ruby (dRuby/DRb) Multiple Remote Code Execution Vulnerabilities	10.0 (High)	99 %	192.168.4.85		8787/tcp	Tue, Mar 19, 2024 5:54 PM UTC
Operating System (OS) End of Life (EOL) Detection	10.0 (High)	80 %	192.168.4.85		general/tcp	Tue, Mar 19, 2024 5:46 PM UTC
Apache Tomcat AJP RCE Vulnerability (Ghostcat)	9.8 (High)	99 %	192.168.4.85		8009/tcp	Tue, Mar 19, 2024 6:00 PM UTC
MySQL / MariaDB Default Credentials (MySQL Protocol)	9.8 (High)	95 %	192.168.4.85		3306/tcp	Tue, Mar 19, 2024 5:53 PM UTC
vsftpd Compromised Source Packages Backdoor Vulnerability	9.8 (High)	99 %	192.168.4.85		21/tcp	Tue, Mar 19, 2024 5:55 PM UTC
vsftpd Compromised Source Packages Backdoor Vulnerability	9.8 (High)	99 %	192.168.4.85		6200/tcp	Tue, Mar 19, 2024 5:55 PM UTC
DistCC RCE Vulnerability (CVE-2004-2687)	9.3 (High)	99 %	192.168.4.85		3632/tcp	Tue, Mar 19, 2024 5:54 PM UTC
PostgreSQL Default Credentials (PostgreSQL Protocol)	9.0 (High)	99 %	192.168.4.85		5432/tcp	Tue, Mar 19, 2024 5:53 PM UTC
VNC Brute Force Login	9.0 (High)	95 %	192.168.4.85		5900/tcp	Tue, Mar 19, 2024 5:50 PM UTC
SSH Brute Force Logins With Default Credentials Reporting	7.8 (High)	95 %	192.168.4.85		22/tcp	Tue, Mar 19, 2024 5:56 PM UTC

Greenbone Enterprise Appliance Copyright © 2009-2023 by Greenbone AG

Figure 10/Results