Network Vulnerability Assessment Report

Network Assessment:

Conduct a comprehensive vulnerability assessment using tools like Nessus, within a simulated network environment.

Domain:

It fall under the category **Vulnerability Assessment** – Using automated tools like **Nessus** to detect security flaws.

Submitted By:

Akshaya Dasari
Extion Infotech (Dec-Jan) 2024-2025

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to everyone who supported me during my cybersecurity internship. I am grateful to the entire cybersecurity team for their collaborative spirit and for providing a stimulating learning environment. I also extend my appreciation to "**Extion Infotech Internship Program**" for this incredible opportunity to gain practical experience.. This internship has been a great experience, and I am excited to apply the knowledge and skills I have acquired in future endeavors.

TABLE OF CONTENT

1. Introduction

1.1 Purpose of the Report

The primary purpose of this report is to document the findings of a N**etwork Vulnerability Assessment (NVA)** conducted using **Nessus** on a simulated network environment, specifically targeting a system running **Metasploit**. This assessment aims to identify critical security vulnerabilities, analyze their potential impact, and propose mitigation strategies to enhance network security.

By performing this assessment, the report serves the following objectives:

- Evaluate the security posture of the simulated network.
- Identify and categorize vulnerabilities based on severity.
- Recommend appropriate remediation measures to mitigate identified risks.

1.2 Scope of the Assessment

This assessment focuses on scanning a predefined network environment that simulates real-world security threats. The key aspects covered in this assessment include:

- **Target System**: A virtual machine running Metasploit, which contains known vulnerabilities for testing purposes.
- **Assessment Tools**: Nessus vulnerability scanner for detecting security flaws.
- **Vulnerability Identification**: Critical vulnerabilities will be identified and analyzed in detail.
- **Mitigation Strategies**: A structured plan for addressing and mitigating the detected vulnerabilities.
- **Final Report & Presentation**: A comprehensive report detailing findings and proposed remediation steps, summarizing key insights.

1.3 Tools Used

The following tools were utilized for conducting the vulnerability assessment:

1. **Nessus** – A powerful vulnerability scanning tool used to detect security flaws in the target network.
2. **Metasploit Framework** – A penetration testing platform that simulates real-world vulnerabilities.

3. **Linux** – Used for running network scans and analyzing security weaknesses.

2. Nessus

2.1 Introduction to Nessus

Nessus is a widely used **vulnerability assessment tool** developed by **Tenable, Inc.** It is designed to scan networks, systems, and applications to identify security weaknesses, misconfigurations, and compliance issues. Nessus is highly effective in detecting **known vulnerabilities**, including unpatched software, misconfigured firewalls, weak passwords, and potential entry points for cyber threats.

Key Features of Nessus:

- **Comprehensive Vulnerability Scanning** – Detects thousands of security flaws across multiple platforms.
- **Regularly Updated Database** – Ensures scanning accuracy with frequent updates on new vulnerabilities.
- **Customizable Scanning** – Allows users to perform targeted scans based on specific security concerns.
- **Detailed Reporting** – Provides in-depth reports with severity ratings and remediation suggestions.
- **User-Friendly Interface** – Simple GUI and command-line options for ease of use.

2.2 Nessus Installation and Configuration

1. Installing Nessus

Nessus can be installed on multiple operating systems, including **Windows, Linux, and macOS**. Below are the general installation steps for **Linux-based systems** like Kali Linux or Ubuntu:

Step 1: Download Nessus

- Visit the official **Tenable Nessus Download** page.
- Choose the appropriate version based on your operating system.

2. Configuring Nessus

Step 1: Create an Account

- On the Nessus web interface, sign up for a **Nessus Essentials (free)** or **Nessus Professional** license.

- Enter the activation code provided by Tenable.

Step 2: Plugin Updates

- Nessus will **download and update** its plugins automatically, ensuring the latest vulnerability checks are available.

Step 3: Configure Scan Policies

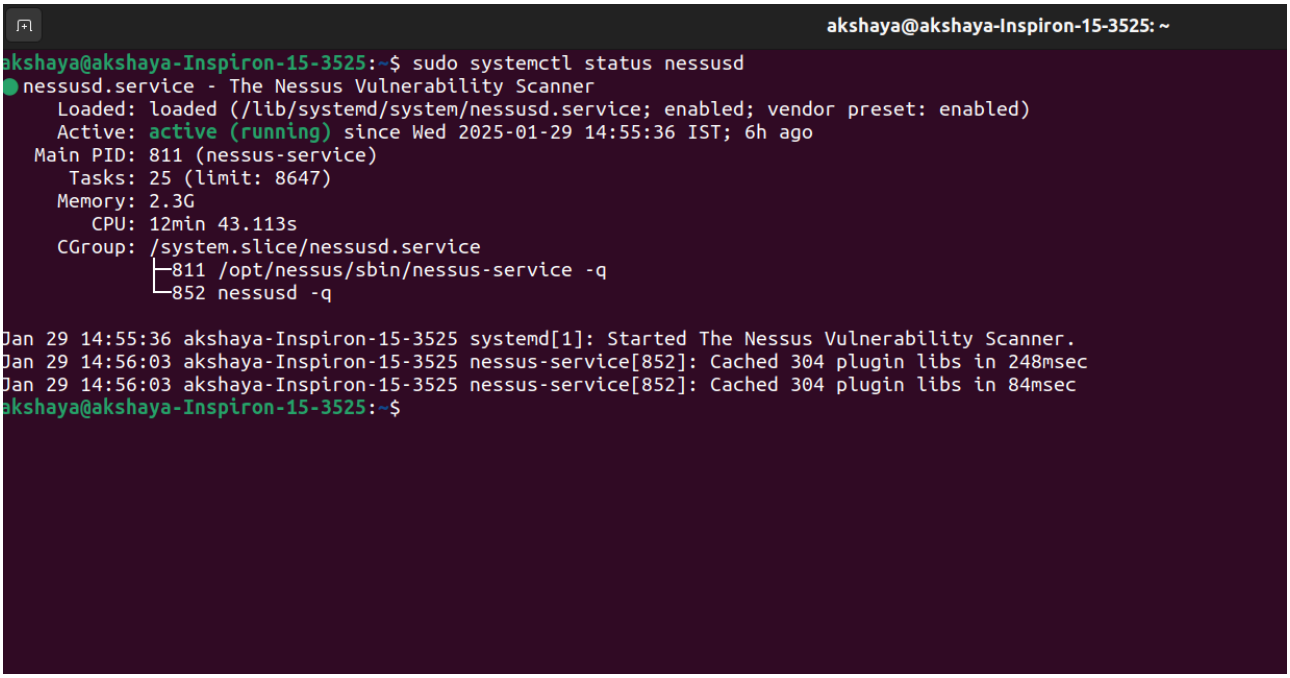
- Define scan templates such as **Basic Network Scan, Advanced Scan, or Web Application Scan** based on requirements.

Step 4: Add Target Systems

- Specify the **IP address or hostname** of the target systems to be scanned.
- Configure scanning options like **port range, scan depth, and authentication credentials** if needed.

Step 5: Run a Scan

- Click on **"New Scan"**, select the appropriate scan type, and start the assessment.
- After completion, analyze the **severity levels (Critical, High, Medium, Low)** and take necessary remediation actions.



3. Metasploit

3.1 Introduction to Metasploit

Metasploit is a powerful penetration testing framework used for exploiting, testing, and securing networks and systems. Developed by **Rapid7**, it is widely utilized by ethical hackers and security professionals for conducting **vulnerability assessments, penetration tests**, and exploit development.

Key Features of Metasploit:

- **Exploit Framework** – Provides pre-built exploits to test security vulnerabilities.
- **Payloads and Post-Exploitation Modules** – Offers various payloads for system control after exploitation.
- **Auxiliary Modules** – Supports scanning, fuzzing, and reconnaissance tasks.
- **Meterpreter** – An advanced payload that allows remote system control without detection.
- **Integration with Tools** – Works alongside Nmap, Nessus, and Wireshark for better attack simulation.

3.2 Metasploit Installation and Configuration

Step 1: Update System

```
sudo apt update && sudo apt upgrade -y
```

Step 2: Install Dependencies

Metasploit requires several dependencies. Install them using:

```
sudo apt install -y git curl wget gnupg2 build-essential
libssl-dev libreadline-dev zlib1g-dev libsqlite3-dev
```

Step 3: Clone the Metasploit GitHub Repository

```
git clone https://github.com/rapid7/metasploit-framework.git
cd metasploit-framework
```

Step 4: Install RVM (Ruby Version Manager)

Metasploit requires a specific Ruby version. Install **RVM** and Ruby:

```
\curl -sSL https://get.rvm.io | bash -s stable
source ~/.rvm/scripts/rvm
rvm install 3.2.2  # (Use the latest compatible Ruby version)
rvm use 3.2.2 --default
```

Step 5: Install Bundler and Required Gems

```
gem install bundler
bundle install
```

Step 6: Initialize the Database

Metasploit uses PostgreSQL. Start and set up the database:

```
sudo apt install -y postgresql postgresql-contrib
sudo systemctl start postgresql
sudo systemctl enable postgresql
msfdb init
```

Step 7: Run Metasploit

Start Metasploit using:

```
./msfconsole
```

4. Implementation

Step 1: Start Nessus Service

Ensure that **Nessus** is running. Open a terminal and start the service:

Step 2: Access Nessus Web Interface

1. Open a browser and go to **(https://localhost:8834)**
2. Log in with your Nessus credentials.

Step 3: Create a New Scan

1. Click on **"Scans" → "New Scan"**
2. Select **"Basic Network Scan"**
3. Fill in the details:
    - **Name:** (e.g., "Scan for 192.168.1.113")
    - **Description:** (Optional)
    - **Targets:** `192.168.1.113`
    - **Schedule:** (Run once or set a recurring schedule)
4. Click **"Save"**

Step 4: Run the Scan

1. Go to **"My Scans"**, locate your created scan.
2. Click **"Start"** to begin scanning the target **192.168.1.113**.

Step 5: Monitor Scan Progress

- Nessus will analyze the target for **open ports, vulnerabilities, and misconfigurations**.

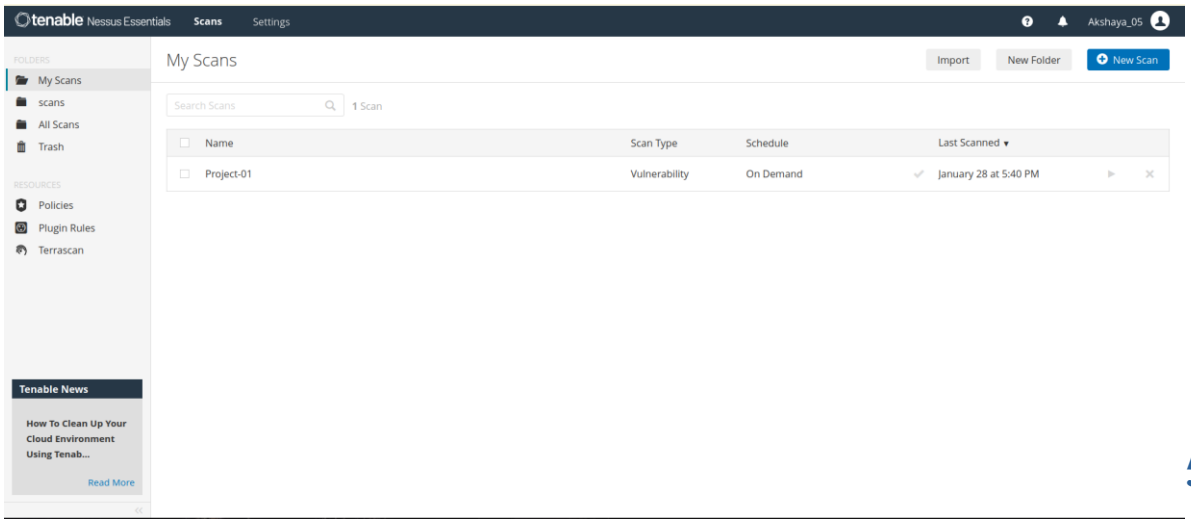
- The scan may take a few minutes, depending on the network and system configuration.

Step 6: Review and Export Scan Results

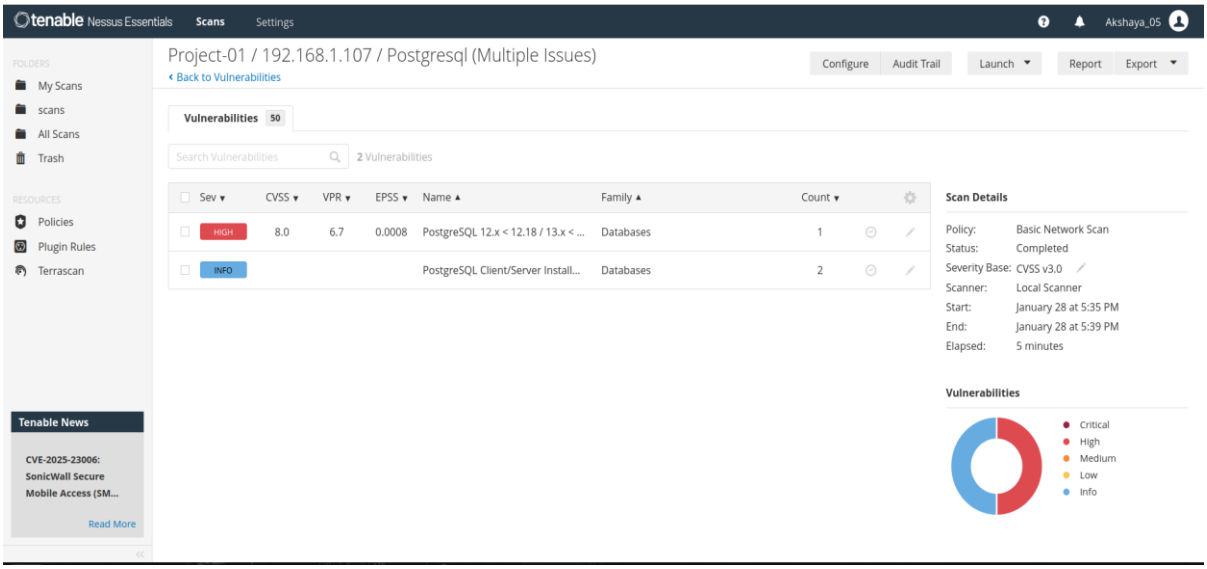
1. Once completed, click on the scan report to view:
    - **Open Ports & Services**
    - **Detected Vulnerabilities**
    - **Risk Levels (Low, Medium, High, Critical)**
2. To export results, click **"Export"**, and select:
    - **Formats:** HTML, PDF, CSV, or Nessus file (.nessus)

Step 7: Analyze and Mitigate Vulnerabilities

- Review the critical vulnerabilities and **apply patches** or **security fixes** as needed.
- If necessary, rerun the scan to verify fixes.



5.



Vulnerability Analysis and Findings

5.1 Summary of Detected Vulnerabilities

The Nessus scan detected **50 vulnerabilities** on the target system. Among these, we focus on the most critical and high-impact vulnerabilities that pose significant risks.

5.2 Detailed Analysis of Critical Vulnerabilities

1. **Vulnerability 1: PostgreSQL SQL Injection (High Severity)**
    - **Description:** The installed version of PostgreSQL (12.x < 12.18, 13.x < 13.14, 14.x < 14.11, 15.x < 15.6) is vulnerable to a SQL injection attack due to a late privilege drop in `REFRESH MATERIALIZED VIEW CONCURRENTLY`. This allows attackers to execute arbitrary SQL functions with elevated privileges.
    - **Severity:** High
    - **Impact:** An attacker could potentially exploit this to execute unauthorized SQL commands, compromise sensitive data, or escalate privileges.
    - **Mitigation:** Upgrade to PostgreSQL **12.18, 13.14, 14.11, or 15.6** or later.
    - **Reference:** Nessus Report
2. **Vulnerability 2: Apache mod_status Information Disclosure (Medium Severity)**
    - **Description:** The Apache `mod_status` module exposes detailed server activity and performance metrics via `/server-status`, potentially leaking sensitive information to unauthorized users.
    - **Severity:** Medium
    - **Impact:** Attackers can view system information such as active connections, CPU load, and request details, which can aid in further attacks.
    - **Mitigation:** Disable `mod_status` in Apache configuration or restrict access to specific trusted hosts.
    - **Reference:** OWASP Guide
3. **SSL Certificate Cannot Be Trusted**

- **Description:**The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken.
- **Severity:**Medium
- **Impact:**

  - First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.

  - Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's 'notBefore' dates, or after one of the certificate's 'notAfter' dates.

  - Third, the certificate chain may contain a signature that either didn't match the certificate's information or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.
- **Mitigation:** Purchase or generate a proper SSL certificate for this service.
- **Reference:**

  https://www.itu.int/rec/T-REC-X.509/en

  https://en.wikipedia.org/wiki/X.509

6. Conclusion and recommendations

6.1 Summary and Findings

The Nessus vulnerability scan on the target system (IP: 192.168.1.113) revealed **50 vulnerabilities**, with varying severity levels ranging from low to high.

Among these, critical vulnerabilities were identified, particularly affecting PostgreSQL, Apache services and SSL certificate.

6.2 Best Practices for Network Security

To maintain a strong network security posture, the following best practices should be followed:

1. **Firewall and Security Policies**
    - Configure firewalls to allow only necessary services and block unauthorized access.
    - Enforce strict inbound and outbound traffic rules.
2. **Secure Configuration Management**
    - Disable unnecessary services and ports.
    - Harden server configurations by following industry benchmarks (e.g., CIS benchmarks).
3. **User Awareness and Training**
    - Conduct regular security training for employees.
    - Educate users on phishing, social engineering, and safe browsing practices.
4. **Logging and Monitoring**
    - Enable centralized logging for network and application activities.
    - Regularly review logs for unusual activities or signs of compromise.
5. **Incident Response and Recovery Plan**
    - Develop a structured incident response plan.
    - Conduct periodic security drills to test response readiness.