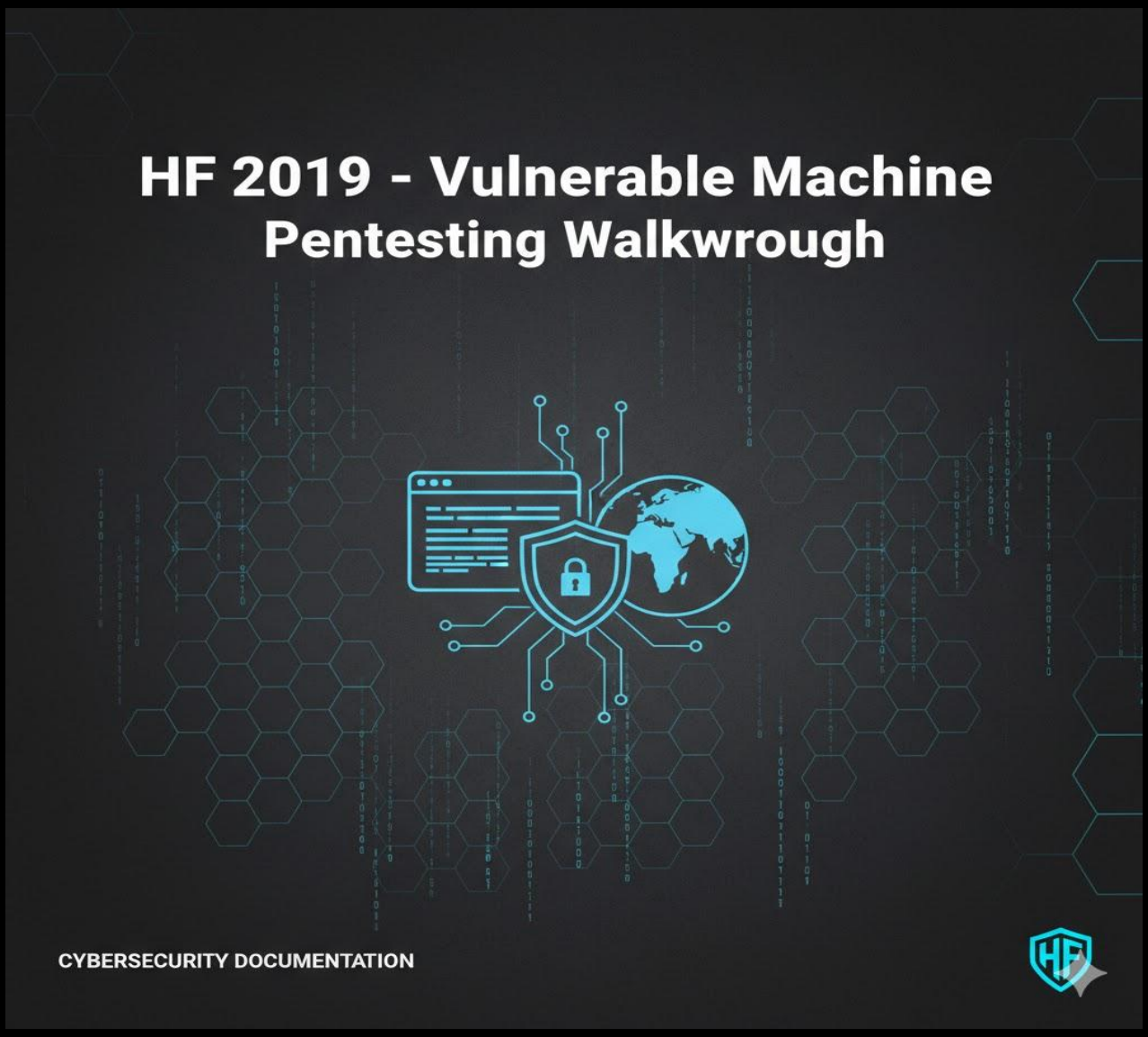
HF 2019 – Vulnerable Machine Pentesting Walkthrough

A practical step-by-step exploit workflow



Introduction

HF-2019 is a vulnerable machine designed for practicing real-world web application exploitation and privilege access techniques.
The machine primarily exposes:

- **HTTP (Port 80)** hosting a **WordPress installation**
- **SSH (Port 22)** for remote login
- Possible misconfigurations and exposed data that can be leveraged during enumeration

The main weaknesses typically include:

- Outdated WordPress components
- Publicly exposed user information
- Weak password or crackable password hash
- Poor server hardening

The exploitation path mainly revolves around:

1. Enumerating WordPress
2. Extracting usernames
3. Retrieving password hashes
4. Cracking them
5. Logging in through SSH

Walkthrough

---

1) Identify the Target IP

```
sudo netdiscover -r 192.168.56.0/24
```

Explanation:
Discover the HF-2019 machine's IP address inside the lab network.

---

2) Perform a Full Port Scan

```
nmap -p- -A -T4 <TARGET_IP> -oN full_scan.txt
```

Explanation:
Scan all ports to locate open services and gather service details. This reveals entry points—especially HTTP and SSH.

---

3) Run an Aggressive Scan

```
nmap -sC -sV -O <TARGET_IP> -oN aggressive_scan.txt
```

Explanation:
Use built-in scripts to detect versions, identify vulnerabilities, and fingerprint the OS.

---

4) Analyze Nmap Results

Explanation:
Check which services are running.
HF-2019 commonly reveals:

- **Port 80 → WordPress website**
- **Port 22 → SSH service**

This indicates the machine is likely vulnerable via web enumeration leading to credential compromise.

---

5) Confirm the HTTP Service

```
curl -I http://<TARGET_IP>
```

Explanation:
Verify that the web server is accessible and responding correctly.

---

6) Inspect the Webpage & Source Code

```
curl http://<TARGET_IP>/ | head -n 40
```

Explanation:
Check HTML patterns and directory hints to identify what CMS is running.

---

7) Identify WordPress

Indicators:

- `/wp-content/`
- `/wp-includes/`
- Meta tags containing *WordPress*

Explanation:
Confirming WordPress guides the next phase—WordPress-specific enumeration.

---

8) Technology Fingerprinting (Optional)

Tool: **Wappalyzer** Browser Extension
Explanation:
Visual confirmation of technologies like PHP, Apache, WordPress, themes, and plugins.

---

9) Enumerate WordPress Users

```
wpscan --url http://<TARGET_IP> --enumerate u --api-token YOUR_TOKEN -o wpscan_users.txt
```

Explanation:
WPScan extracts publicly exposed usernames, which are needed for authentication attacks or hash cracking.

---

10) Retrieve Password Hash

Explanation:
Depending on the machine, a vulnerable path or plugin may leak a user hash.
WPScan output or exposed backup files often reveal:

- WordPress user
- Hashed password (phpass)

This hash is used for cracking.

---

11) Crack the Password Using John

```
john --format=phpass --wordlist=/usr/share/wordlists/rockyou.txt hash.txt
```

Explanation:
John the Ripper attempts wordlist-based cracking. Weak or common passwords are often recovered quickly.

---

12) Gain SSH Access Using Cracked Credentials

```
ssh <username>@<TARGET_IP>
```

Explanation:
Use the cracked password for SSH login, granting full access to the HF-2019 machine.

Final Outcome

- WordPress enumerated
- Usernames extracted
- Password hash obtained
- Password cracked
- SSH login successful

You now have full access to the HF-2019 vulnerable machine.