CyberSci 2021 Regional Challenges

January 23, 2020 11:00am EST

Writeup

Welcome Challenge (100 pts)

Read the rules.

Bastion 1 (50 pts)

Given a pastebin containing logs of the host machine the attackers are trying to connect to, find out what the correct username and password is to connect to the host machine.

Once inside, find out what the highest IP address for the subnet of the host machine is.

Solution

A first look at the pastebin, we can see that it is very large so it will require some formating. There are numerous ways of doing this, I used VSCode because it has a fast regex matcher integrated.

A hint was also given saying that it's possible that users enterred their passwords in the username fields by mistake. Knowing this just fortifies the idea that we should check for failed login attempts and the usernames used.

Here are the regular expressions I used in vscode. These could be used with grep's invert matching or awk.

This will remove everything before the first comma.

```
Dec 10 \d+:\d+:\d+ LabSZ sshd\[\d+\]:
```

This will remove all the words litterally

```
Failed password for invalid user
```

This will remove everything after the used username

```
from \d+.\d+.\d+.\d+ port \d+ \w+
```

Then manipulate the remaining entries to remove the eol characters. I used this and just deleted all of the instances:

```
\n
```

This will leave all of the failed attempts' usernames. We can do some additional formating, such as removing all dupplicates:

```
^(.*)(\r?\n\1)+$
```

Finally this leaves us with a file of about 80 lines which is a lot easier to read. Some additional formating could be done, but unnecessary as a first lookthrough we will find the right password:

```
#my#secr_t#p_ssw_rd#
```

We know this is the right password, because if we look back at the original pastebin, we can look for usernames that were successfully logged in, we find 2 of them:

```
Dec 10 09:32:20 LabSZ sshd[24680]: Accepted password for rock from 119.137.62.142 port 49116 ssh2

Dec 10 14:32:20 LabSZ sshd[24680]: Accepted password for snowdumb from 119.137.62.142 port 49116 ssh2
```

Trying out the password with the username snowdumb successfully logged us in.

Recon 1 (150 pts)

Prompt: We took a cursory glance at this machine but other than the website didn't see much of value. Can you figure out if there's anything here?

Your target: A very long file or document name.

Solution

Initial nmap returns the following:

```
# Nmap 7.91 scan initiated Sat Jan 23 12:46:30 2021 as: nmap -sC -sV -oA nmap 10.0.1.8
Nmap scan report for 10.0.1.8
Host is up (0.010s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 54:bb:c8:28:da:9c:df:af:99:1b:c7:aa:bd:ce:39:5f (RSA)
|   256 09:b8:95:3d:30:4d:6e:46:e0:90:ab:18:e2:a6:0f:bf (ECDSA)
|_  256 13:d6:21:92:04:d0:4d:bf:c6:b5:72:b4:c1:c8:a8:8b (ED25519)
443/tcp open  ssl/http Apache httpd 2.4.41 ((Ubuntu))
|_http-generator: Hugo 0.74.1
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: AxxHatz Konsulting
| ssl-cert: Subject: commonName=AxxhatzWeb/organizationName=Axxhatz/stateOrProvinceName=Some-State/countryName=AU
| Not valid before: 2021-01-05T23:34:48
|_Not valid after:  2022-01-05T23:34:48
| tls-alpn:
|_  http/1.1
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 Sat Jan 23 12:46:57 2021 -- 1 IP address (1 host up) scanned in 26.67 seconds
```

Here SSL is running on port 443. Note: We were told not to touch the open ssh ports.

Looking at the website, there are a couple of paragraphs and links with some filler text, nothing too important. (Unfortunately, I don't have a copy of the webpage to show)

The source code however, contained a little hint to what needed to be done:

```
<? showForKeyedUsers("Send Finbonacci 3 Times to udp ports starting at 89") />
```

The Fibonacci numbers are 89 144 233 when starting at 89.

Searching online, we found port knocking.

We thus used the knockd utility to knock on all three ports with their respective fibonacci numbers.

After doing that, running

```
nmap -sS -p- <ip>
```

Gives us the new open port through which we can find the flag.

Vaccine Database 1 (300 pts)

An initial nmap scan yields the following:

```
# Nmap 7.91 scan initiated Sat Jan 23 13:23:52 2021 as: nmap -sC -sV -p- -oA nmap/ 10.0.1.6
Nmap scan report for 10.0.1.6
Host is up (0.021s latency).
Not shown: 65532 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 9a:69:ee:7a:52:c5:1a:e7:e7:24:f9:97:98:45:cb:d7 (RSA)
|   256 9b:10:d5:cf:bc:85:ee:24:d0:c2:46:da:51:96:1d:56 (ECDSA)
|_  256 ef:fc:5f:96:86:91:ec:00:f6:d1:b8:84:bc:06:1e:40 (ED25519)
53720/tcp open  ftp     vsftpd 3.0.3
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.0.0.5
|      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
62789/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
Service Info: OSs: Linux, Unix; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Jan 23 13:24:51 2021 -- 1 IP address (1 host up) scanned in 58.89 seconds
```

Here FTP is open on port 53720, with anonymous authentication enabled, as well as an Apache Webserver on port 62789.

Running the following with username "anonymous" and password anon will get us in.

```
$ ftp 10.0.1.6 53720
```

```
$ pass
$ ls -laR
```

After running a passive mode full listing we will get the following results:

```
.
..
...
```

Going into the "..." directory we find a file called phpmyadmin_hash_backup.txt which contains a username and password for a phpMyAdmin service, probably running on the webserver previously found.

```
phpmyadmin:*A7DCD8A49BB131BF563D832B5F086681B76CDE45
```

The hash is 40 chars long, which initially could help us identify it as a SHA1 hash and analyzing the hash returns a similar answer.

Running

```
hashcat -m 100 hash.txt /usr/share/wordlists/rockyou.txt
```

Yields no results. This was the same case for other wordlists. Finally, using online databases, we could find the associated password: `TheCure!!!`

We now had the password and the username for the phpMyAdmin service.

Navigating to `http://10.0.1.6:62789/phpmyadmin/` we can input the two and successfully login.

The flag is the name of the database.