**Armageddon**

Difficulty: Easy

*Linux*

Nmap

First we start with our classic nmap scan

|  |
| --- |

We have ports 22 and 80 open on the machine. We see from the scan that the web page is running drupal 7 and has some other pages listed.

Time to move on to fuzzing the site.

Web page

Going to the web page, we are presented with the following logon.

|  |
| --- |

Trying default credentials provides nothing

Next I performed a FUZZ scan and got the following

|  |
| --- |

Initial thoughts on this - web.config looks interesting and so does cgi-bin.

Found nothing useful initially in those directories.

Found the version of drupal

|  |
| --- |

Looking up exploits online gives us some hits. I am going to try the following

<https://github.com/dreadlocked/Drupalgeddon2.git>

Metasploit also has a module for this too.

|  |
| --- |

Executing the script gives us a shell

|  |
| --- |

Shell

First thing was try to spawn a better shell, but this one will not allow certain characters and the user we are currently logged in as has very little permissions.

I did a grep search for passwords and found this

|  |
| --- |

CQHEy@9M\*m23gBVj

Looking into this more, I found this for mysql (I think)

|  |
| --- |

Going to attempt this.

I was able to log into the mysql database.

**Command**: *mysql -u drupaluser -p CQHEy@9M\*m23gBVj -e "SELECT \*" drupal*

I see the following entries in the “drupal” database

| mysql -u drupaluser -p CQHEy@9M\*m23gBVj drupal -e "SELECT \*" |
| --- |

Time to see if we can get some users and passwords out of this

**NOTE:** It took me some time to do this. Things I learned: If you are limited to a command line, you must execute all SQL code inside the command line and not within a mysql shell. I am including some more commands I used to show the step by step process of how I got users and passwords

| *mysql -u drupaluser -p -e "SHOW DATABASES;"* |
| --- |

Here, we enumerated what databases are on the system. Based on previous information, this user should have access only to the “drupal” database.

I posted a picture above of all the tables within the database, but the one most interesting to us is the “users” one

| *mysql -u drupaluser -p -e “use drupal;show tables;”* |
| --- |

I ended up getting a username and password

| *mysql -u drupaluser -p -e "use drupal;select name,pass from users;"* |
| --- |

Time to decrypt this password

$S$DgL2gjv6ZtxBo6CdqZEyJuBphBmrCqIV6W97.oOsUf1xAhaadURt

This once again took me some time. The hash was not easily identifiable, so I ended up plugging it into john in hopes of it being identified. I got a success!

| *john --wordlist=/opt/rockyou.txt hashpass.txt* |
| --- |

Username: brucetherealadmin

Password: booboo

User

Looking at our permissions shows the following

|  |
| --- |

Looking at the version of snap, we are on a non-vulnerable version to dirtysock (something I looked up)

| *Snap --version* |
| --- |

Following this GTFOBins site, I installed fpm and created a snap installation. Then I uploaded it to the target machine.

GTFOBins: <https://gtfobins.github.io/gtfobins/snap/>

Installing fpm: <https://fpm.readthedocs.io/en/latest/installing.html>

|  |
| --- |

I had to use *curl* since *wget* was not installed on the target machine.

The command worked and gave me the id of root. Now I just need to modify it to give me a full shell.

I decided to cheat and just get the root flag. I know we can get root by putting in an ssh key, or changing the password of root to something else with *passwd*. Either way, we had root and now the flag.

|  |
| --- |