Sécurité Web 1

Olivier LASNE

2021-01-18

Sécurité Web

Pour ce TP nous utiliserons la machine **OWASP Broken Web Apps** que vous avez déjà. Et l'iso **"From SLQi to Shell"** que vous pouvez télécharger ici :

https://pentesterlab.com/exercises/from_sqli_to_shell/iso

Rappel SQL

SQL est un langage de requêtes de base de données.

Vous pouvez vous connecter en SSH à votre VM OWASP Broken Web Apps en SSH. `root`/`owaspbwa`.

`ssh root@192.168.56.101` (remplacer avec l'IP de la machine)

On peut y lancer MySQL avec la commande suivante : `mysql -u root -powaspbwa`

Cela lance un shell MySQL. On obtient de l'aide avec la command `help` ou `\h`.

```
mysql> help

For information about MySQL products and services, visit:
   http://www.mysql.com/
For developer information, including the MySQL Reference Manual, visit:
   http://dev.mysql.com/
To buy MySQL Enterprise support, training, or other products, visit:
   https://shop.mysql.com/

List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?         (\?) Synonym for `help'.
clear     (\c) Clear the current input statement.
connect   (\r) Reconnect to the server. Optional arguments are db and
   host.
delimiter (\d) Set statement delimiter.
edit      (\e) Edit command with $EDITOR.
ego       (\G) Send command to mysql server, display result vertically.
exit      (\q) Exit mysql. Same as quit.
go        (\g) Send command to mysql server.
help      (\h) Display this help.
nopager   (\n) Disable pager, print to stdout.
notee     (\t) Don't write into outfile.
pager     (\P) Set PAGER [to_pager]. Print the query results via PAGER.
print     (\p) Print current command.
prompt    (\R) Change your mysql prompt.
quit      (\q) Quit mysql.
rehash    (\#) Rebuild completion hash.
source    (\.) Execute an SQL script file. Takes a file name as an
   argument.
```

```
status     (\s) Get status information from the server.
system     (\!) Execute a system shell command.
tee        (\T) Set outfile [to_outfile]. Append everything into given
   outfile.
use        (\u) Use another database. Takes database name as argument.
charset    (\C) Switch to another charset. Might be needed for
   processing binlog with multi-byte charsets.
warnings   (\W) Show warnings after every statement.
nowarning  (\w) Don't show warnings after every statement.

For server side help, type 'help contents'
```

On peut voir les bases de données avec la commande `show databases;`

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| .svn               |
| bricks             |
| bwapp              |
| citizens           |
| cryptomg           |
| dvwa               |
| gallery2           |
| getboo             |
| ghost              |
| gtd-php            |
| hex                |
| isp                |
| joomla             |
| mutillidae         |
| mysql              |
| nowasp             |
| orangehrm          |
| personalblog       |
| peruggia           |
| phpbb              |
| phpmyadmin         |
| proxy              |
| rentnet            |
| sqlol              |
| tikiwiki           |
| vicnum             |
| wackopicko         |
| wavsepdb           |
| webcal             |
| webgoat_coins      |
| wordpress          |
| wraithlogin        |
```

```
| yazd                |
+---------------------+
34 rows in set (0.00 sec)
```

On selectionne une base avec la commande **use** :

```
mysql> use peruggia;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
```

On peut ensuite lister les tables avec la commande **show tables;** :

```
mysql> show tables;
+---------------------+
| Tables_in_peruggia |
+---------------------+
| picdata             |
| users               |
+---------------------+
2 rows in set (0.00 sec)
```

/ ! Les commandes `show` et `help` sont des commandes du SHELL MySQL. Il ne s'agit pas de requêtes SQL valides.

On peut selectionner l'ensemble des champs d'un tables avec la requête `SELECT * FROM nom_de_la_table`.
Le caractère * signifie *tout les champs*.

```
mysql> SELECT * FROM users;
+----+----------+----------------------------------+
| ID | username | password                         |
+----+----------+----------------------------------+
|  1 | admin    | 21232f297a57a5a743894a0e4a801fc3 |
|  2 | user     | ee11cbb19052e40b07aac0ca060c23ee |
+----+----------+----------------------------------+
2 rows in set (0.00 sec)
```

On peut selectionner un seulement certains champs, en les listants séparés par des virgules.

```
mysql> SELECT ID, username FROM users;
+----+----------+
| ID | username |
+----+----------+
|  1 | admin    |
|  2 | user     |
+----+----------+
2 rows in set (0.00 sec)
```

Note : il n'est pas nécessaire de mettre `SELECT` et `FROM` en majuscule. Néanmoins il s'agit de la convention prise dans la plupart des cas de façon à distinguer les *champs* des *opérateurs*.

On peut utiliser le mot-clé `WHERE` pour filtrer les éléments sélectionnés.

```
mysql> SELECT password FROM users WHERE username = 'admin';
+----------------------------------+
| password                         |
+----------------------------------+
| 21232f297a57a5a743894a0e4a801fc3 |
+----------------------------------+
1 row in set (0.00 sec)
```

Exercice : Selectionner le nom de l'utilisateur avec l'ID 2.

Exercice 2 : Dans la base de données *sqlol*. Faire une requête qui trouve si l'utilisateur avec l'*id* 2 est admin. Le résultat de la requête doit donner un *0* ou un *1*.

On peut utiliser l'opérateur **`AND`** pour préciser plusieurs conditions.

```
mysql> SELECT * FROM users WHERE id=1 AND username='admin';
+----+----------+----------------------------------+
| ID | username | password                         |
+----+----------+----------------------------------+
|  1 | admin    | 21232f297a57a5a743894a0e4a801fc3 |
+----+----------+----------------------------------+
1 row in set (0.00 sec)
```

Exercice : Dans la table *accounts* de la base de données *nowasp*. Faire une requête qui authentifie un utilisateur c'est à dire :

- Renvoie des données si le nom d'utilisateur et le mot de passe sont bons (et correspondent au même utilisateur).
- Ne renvoie pas de données, si le couple utilisateur mot de passe n'est pas valide.

Pour cela, faire une requête *WHERE* et *AND* qui vérifie à la fois le nom d'utilisateur et le mot de passe.