sqlmap frequently asked questions (FAQ)

Bernardo Damele A. G. and Miroslav Stampar

July 14, 2012 (**DRAFT**)

Abstract

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

Contents

1 Frequently Asked Questions **1**

1.1 What is sqlmap? 1

1.2 How do I execute sqlmap? 1

1.3 Can I integrate sqlmap with a security tool I am developing? . . 2

1.4 Can I integrate sqlmap with a commercial closed source security tool my company is developing? 2

1.5 How can I report bugs or request new features? 2

1.6 Can I contribute occasionally to the development? 3

1.7 Can I actively contribute in the long-term development? 3

1.8 How can I support the development and show my appreciation? . 3

1.9 How can I follow closely the development? 4

1.10 Will you support other database management systems? 4

1.11 Can you hack a site for me? 4

1.12 When sqlmap will switch to Python 3? 4

1.13 How can I shorten the payloads injected by sqlmap? 4

1.14 What does `WARNING unknown charset '...'` mean? 5
1.15 How to use sqlmap with `mod_rewrite` enabled? 5
1.16 Why is sqlmap not able to get password hashes in some cases? . 5
1.17 What is `--text-only` switch? 5
1.18 I am getting `[CRITICAL] connection timed` while I am able to browse the site normally? 6
1.19 Is it possible to use `INSERT/UPDATE` SQL commands via `--sql-query`, `--sql-shell` and `--sql-file`? 6
1.20 sqlmap is not able to detect/exploit injection while other commercial tools are? 6
1.21 How can I dump only certain entries of a table based on my condition? 6
1.22 Where can I find old versions of sqlmap? 7

1 Frequently Asked Questions

1.1 What is sqlmap?

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

1.2 How do I execute sqlmap?

If you are running on a UNIX/Linux system type the following command from a terminal:

```
python sqlmap.py -h
```

You can also see the detailed help message typing:

```
python sqlmap.py -hh
```

If you are running on a Windows system type the following command from a terminal:

```
C:\Python27\python.exe sqlmap.py -h
```

Where `C:\Python27` is the path where you installed Python >= **2.6** and < **3.0**.

1.3 Can I integrate sqlmap with a security tool I am developing?

Yes. sqlmap is released under the terms of the GPLv2, which means that any derivative work must be distributed without further restrictions on the rights granted by the General Public License itself.

1.4 Can I integrate sqlmap with a commercial closed source security tool my company is developing?

We **might** consider to release you a copy under a commercial license - drop us an email and we will discuss it through.

1.5 How can I report bugs or request new features?

Bug reports are welcome! Please report all bugs on the issue tracker or, alternatively, to the mailing list.

Guidelines:

- Before you submit a bug report, search both open and closed issues to make sure the issue has not come up before. Also, check the user's manual for anything relevant.
- Make sure you can reproduce the bug with the latest development version of sqlmap.
- Your report should give detailed instructions for how to reproduce the problem. If sqlmap raises an unhandled exception, the traceback is needed. Details of the unexpected behaviour are welcome too. A small test case (just a few lines) is ideal.
- If you are making an enhancement request, lay out the rationale for the feature you are requesting. *Why would this feature be useful?*
- If you are not sure whether something is a bug, or want to discuss a potential new feature before putting in an enhancement request, the mailing list is a good place to bring it up.

1.6 Can I contribute occasionally to the development?

All code contributions are greatly appreciated. First off, clone the Git repository, read the user's manual carefully, go through the code yourself and drop us an email if you are having a hard time grasping its structure and meaning. We apologize for not commenting the code enough - you could take a chance to read it through and improve it.

Our preferred method of patch submission is via a Git pull request.

Each patch should make one logical change. Please follow the existing stylistic conventions: wrap code to 76 columns when possible. Avoid tabbing, use four blank spaces instead. Before you put time into a nontrivial patch, it is worth discussing it on the mailing list or privately by email.

Many people have contributed in different ways to the sqlmap development. **You** can be the next!

1.7 Can I actively contribute in the long-term development?

We are constantly seeking for people who can write some clean Python code, are up to do security research, know about web application security, database assessment and takeover, software refactoring and are motivated to join the development team.

If this sounds interesting to you, send us your pull requests - we are open to discuss granting of push access to the main repository if you prove professionalism, motivation and ability to write proper Python code.

1.8 How can I support the development and show my appreciation?

sqlmap is the result of numerous hours of passionated work from a small team of computer security enthusiasts. If you appreciated our work and you want to see sqlmap kept being developed, please consider making a donation to our efforts via PayPal to `dev@sqlmap.org`.

1.9 How can I follow closely the development?

We tend to keep our Twitter page, @sqlmap, up to date with the development. We certainly update it more often than the mailing list. Hence, if you are keen on keeping a closer look at the development you can:

- Watch the project on GitHub given you have a GitHub account.

- Subscribe to the Atom feed in your feed reader of choice.
- Follow us on Twitter, @sqlmap.
- Watch demos on YouTube: #1 and #2.
- Subscribe to the mailing list.
- Alternatively, you can subscribe to the RSS feed.
- You can also browse the posts' archive online.

1.10 Will you support other database management systems?

We already support the major and some minor databases. We do have plans to extend support for some of them and support also new ones: Informix and Ingres at some point in time.

1.11 Can you hack a site for me?

No.

1.12 When sqlmap will switch to Python 3?

Currently there is no pressure on Python projects to switch to the new version of Python interpreter, as the process of switching, especially on larger projects can be cumbersome (due to the few backward incompatibilities). The switch will take place eventually, but currently it is a very low priority task.

1.13 How can I shorten the payloads injected by sqlmap?

You can provide sqlmap with the following two switches:

```
--no-cast           Turn off payload casting mechanism
--no-unescape       Turn off string unescaping mechanism
```

However, on the other hand you might lose the benefits provided by these switches in the default configuration.

1.14 What does `WARNING unknown charset '...'` mean?

sqlmap needs to properly decode page content to be able to properly detect and deal with internationalized characters. In some cases web developers are doing mistakes when declaring used web page charset (e.g. `iso_8859` instead of standardized name `iso-8859`), which can cause problems. As a failsafe mechanism we have incorporated heuristic detection engine chardet, so in most cases sqlmap will deal with this kind of problems automatically. Nevertheless, you are strongly advised to report us back those typographic *mistakes* so we could handle them manually inside the code.

Question(s): #1 #2 #3

1.15 How to use sqlmap with `mod_rewrite` enabled?

Append an asterisk, `*`, to the place where sqlmap should check for injections in URI itself. For example, `./sqlmap.py -u "http://target.tld/id1/1*/id2/2"`, sqlmap will inject its payloads at that place marked with `*` character. This feature also applies to POST data. Multiple injection points are supported and will be assessed sequentially.

Question(s): #1 #2 #3

1.16 Why is sqlmap not able to get password hashes in some cases?

The session user most probably does not have enough permissions for querying on a system table containing password hashes.

Question(s): #1

1.17 What is `--text-only` switch?

Switch `--text-only` is used for removing non-textual data (tags, javascripts, styles, etc.) from the retrieved page content to further improve SQL injection detection capabilities.

Question(s): #1

1.18 I am getting `[CRITICAL] connection timed` while I am able to browse the site normally?

There are few IDSes that filter out all sqlmap requests based on its default `User-Agent` HTTP header (e.g. `User-agent: sqlmap/1.0-dev`). To prevent

this kind of situations you are advised to use switch `--random-agent`. If you are getting those kind of messages for all targets then you most probably need to properly set up your proxy settings (switches `--proxy` and/or `--ignore-proxy`).

Question(s): #1

1.19 Is it possible to use `INSERT/UPDATE` SQL commands via `--sql-query`, `--sql-shell` and `--sql-file`?

It is possible to run those statements as well as any other statement on the target database given that stacked queries SQL injection is supported by the vulnerable application or you are connecting directly to the database with `-d` switch and the session user has such privileges (or a privilege escalation vector has been injected upfront).

Question(s): #1

1.20 sqlmap is not able to detect/exploit injection while other commercial tools are?

In most of those kind of cases blatant error message detection is used by commercial tools leading to *false positive* claims. You have to be aware that a DBMS error message does not mean that the affected web application is vulnerable to SQL injection attacks. sqlmap goes several steps further and never claims an injection point without making through tests if it can be exploited on the first place.

Question(s): #1

1.21 How can I dump only certain entries of a table based on my condition?

sqlmap is very granular in terms of dumping entries from a table. The relevant switches are:

```
--dump               Dump DBMS database table entries
-D DB                DBMS database to enumerate
-T TBL               DBMS database table to enumerate
-C COL               DBMS database table column to enumerate
--start=LIMITSTART   First query output entry to retrieve
--stop=LIMITSTOP     Last query output entry to retrieve
--first=FIRSTCHAR    First query output word character to retrieve
--last=LASTCHAR      Last query output word character to retrieve
```

However, in some cases you might want to dump all entries given a custom `WHERE` condition. For such cases, we recommend using one of the following switches:

```
--sql-query=QUERY   SQL statement to be executed
--sql-shell         Prompt for an interactive SQL shell
--sql-file=SQLFILE  Execute SQL statements from given file(s)
```

For example:

```
--sql-query "SELECT user, password FROM users WHERE privilege='admin'"
```

Question(s): #1

1.22 Where can I find old versions of sqlmap?

From the Downloads page on GitHub.

Question(s): #1