* What is SQL Inject Me?

SQL Inject Me is the Exploit-Me tool used to test for SQL Injection vulnerabilities.

* How does SQL Inject-Me work?

The tool work by submitting your HTML forms and substituting the form value with strings that are representative of an SQL Injection attack.

The tool works by sending database escape strings through the form fields. It then looks for database error messages that are output into the rendered HTML of the page.

The tool does not attempting to compromise the security of the given system. It looks for possible entry points for an attack against the system. There is no port scanning, packet sniffing, password hacking or firewall attacks done by the tool.

You can think of the work done by the tool as the same as the QA testers for the site manually entering all of these strings into the form fields.

* How much does SQL Inject Me cost/ is it open source/ what license is it under?

Exploit-Me tools are free of charge. They are all open source, under Gnu Public License (GPL) v.3.

* Does SQL Inject Me perform source code or network analysis?

No, it is only used for run-time application security testing.

* What is the target audience of SQL Inject Me?

SQL Inject-Me is aimed at developers, testers/ QA staff, and security auditors.

* Will SQL Inject Me detect all SQL injection vulnerabilities?

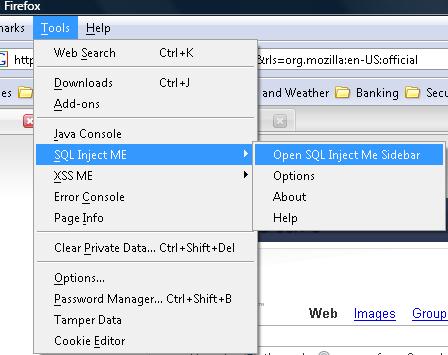
No. SQL Inject Me looks for unexpected responses from the server; as a result, its ability to detect SQL Injection is limited by the responses received from the sever. Testing for advanced attacks, such as blind SQL injection, may require additional manual testing (e.g. attempting to bypass authentication).

* Will Security Compass or any other third party have access to my results?

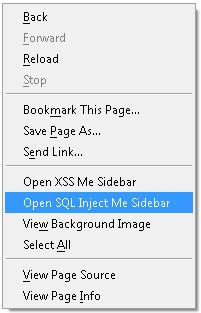
Absolutely Not. Neither Security Compass, nor any third party, maintains data on testing results.

* How do I run SQL Inject Me?

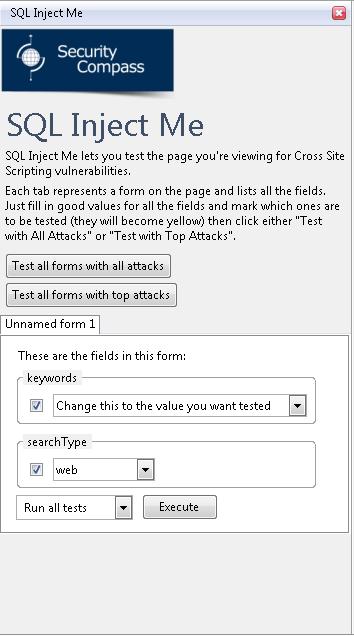
Download the XPI package and install it through Firefox. Once the tool is installed, restart Firefox. You can then start the Exploit-Me tools by using the top-level menu: Tools -> SQL Inject Me -> Open SQL Inject Me Sidebar.



You can also use the context menu by right-clicking on the page that you wish to test and selecting “Open SQL Inject Me Sidebar”.



All the forms in your current web page will appear in a series of tabs in the sidebar, and each tab will have all the corresponding visible and hidden fields listed.



The current value for each field will appear with a corresponding combo box. You can change the values directly in this combo box. The default value is the current value of that field, or if none is specified then you will see the string “Change this to the value you want tested” (as shown for the “keywords” field in the above example).

If you check the box next to a field name, then that field will be tested for SQL injection. If the box is not checked, then the field will not be tested for SQL Injection and the current value listed in the combo box will be submitted every time.

SQL Inject Me works by testing each checked value one at a time. In the above example, the tool would attempt to test the “keywords” field and then the “searchType” field for SQL Injection. The parameters for the submission would look something like:

keywords=SQLInjection\_ATTACK\_STRING&searchType=web

when the “keywords” field is being tested and

keywords=&searchType= SQLInjection\_ATTACK\_STRING

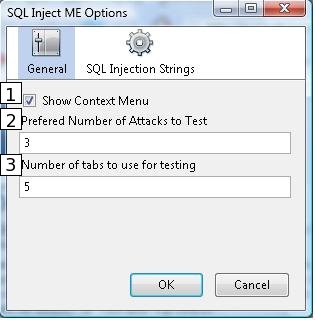
when the “searchType” field is being tested.

The tool will substitute SQLInjection\_ATTACK\_STRING with the list of strings specified in the options. This is called “fuzzing” in application testing terminology. You can choose to fuzz all the attack strings by selecting the “Run all tests” option and pressing execute, or you can choose to fuzz a few of them by selecting the “Run top X attacks” option and pressing execute. Running all tests with the default list of attack strings can be very time consuming if the server responses are not instant or if there are several fields to be tested. Running the top attacks is usually not as thorough but generally allows you to test much quicker, depending on how many attacks you specify to be “top attacks” (see “What are the Options” for SQL Inject Me below).

There are also options at the top of the side bar to “Test all forms with all attacks” and “Test all forms with top attacks”. This will automatically test every field in every form with either all attack strings or the Top X attacks. If you select this option then the checkboxes next to field names will be ignored.

* What are the Options for SQL Inject Me?

There are currently five options in SQL Inject Me that you can access through the top-level menu Tools->SQL Inject Me->Options.



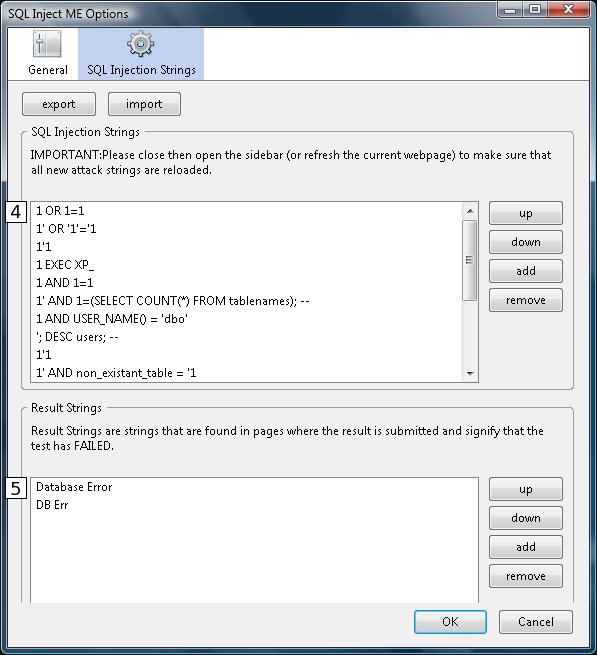
Show Context Menu  
Toggle whether or not the open “SQL Inject Me sidebar” option should be shown in the context menu

Preferred Number of Attacks to Test  
This specifies the number of attacks that should be tested when you select the “Test All Forms with Top Attacks” or “Run Top X Attacks” options in the SQL Inject Me sidebar. If you enter “5″ for this value, then the first 5 values listed in the “SQL Injection Strings” table will be tested.

Number of Tabs to Use for Running Tests  
This specifies how many concurrent tabs can be opened to run the SQL injection tests. More concurrent tabs may mean quicker overall testing, but will also incur greater memory impact. Opening too many concurrent tabs may cause Firefox to crash.

SQL Injection Strings  
SQL Inject Me will enter these strings as the values in the fields that you specify for testing. The tool starts testing from the first string to the last; if you select the “Test All Forms with Top Attacks” or “Run Top X Attacks” options then only the first X attacks will be tested (where “X” is specified in option #1 above). In order to change the order of a particular string in the list, use the “Up” and “Down” buttons. You can also add or remove individual strings by clicking on them and pressing the “Add” and “Remove” buttons. Finally, you can export the entire list or import another list using the export and import buttons located above the list of strings.

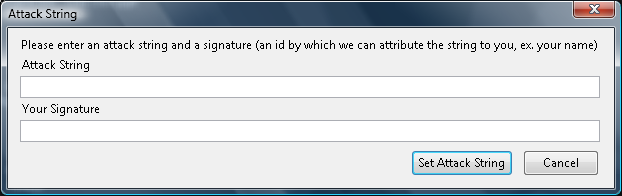
Result Strings  
SQL Inject Me looks for the presence of these strings in the HTTP response returned from the server. If any of these strings are found then the attack string is listed as a potential SQL injection.



* How do I add my own signatures to the files?

Use the SQL Injection Strings tab in the Tools->SQL Inject Me->Options menu. Click on the “Add” button, and the “Attack String” menu will pop up. Enter the attack in the “Attack String” text field. Note that your attack string should run “document.vulnerable=true” in the resulting JavaScript for the tool to work properly. E.g. <script> document.vulnerable=true</script> is a valid SQL Inject Me string.

The “Your signature” field allows you to specify your name to associate to the attack string. This feature was added to allow people to take credit for their attack string contributions.



* How do I interpret the SQL Inject Me results?

SQL Inject Me has three result types:

Failures

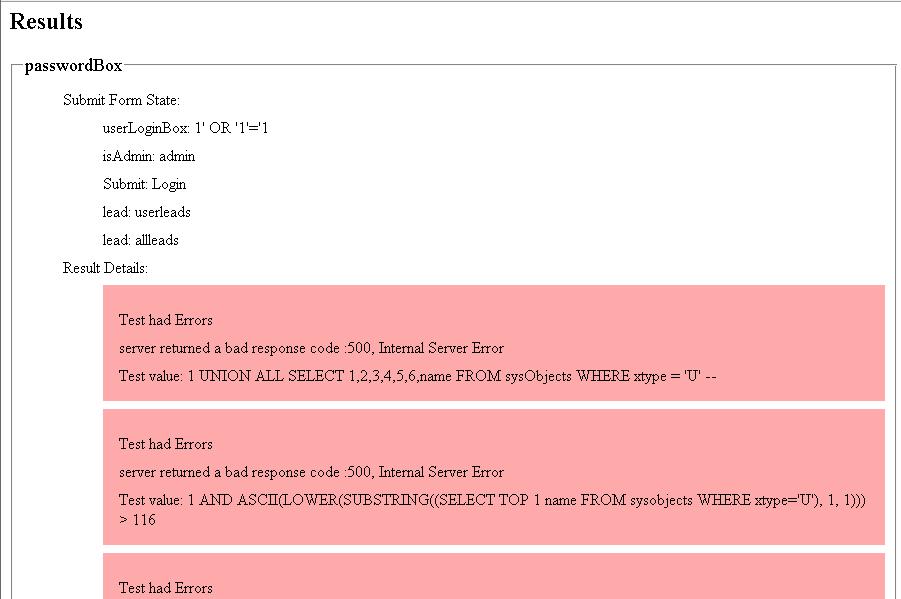
The number of tests that resulted in high likelihood of SQL injection vulnerabilities (e.g. Result string from the user-supplied list is detected)  
Warnings  
Number of tests that resulted in some likelihood of SQL injection vulnerabilities (e.g. there was a difference in the server response between the submission of a normal value and an SQL attack string value)  
Pass  
Number of tests that did not result in any detection of SQL injection

Each result is specified in the detailed section below. Test results are grouped by field name. Failures are listed first, followed by warnings, and then passes.



For each field the following details are given:

Form state  
Values of all other parameters during submission of the form  
Result details  
Individual failures, warnings and passes including the test value that lead to that individual result. This information is important in determining how a particular field may be vulnerable; you can take any of the test values that resulted in a failure and write your own injection string to manually verify.



* Why does my form or field have no name on the SQL Inject Me Sidebar?

In some cases a web page may create a form without specifying a corresponding name, or a form field without specifying a field name. In those cases, there is no name given in the SQL Inject Me sidebar.

* I deleted the default attack and/or error strings but I want to get them back?

Don’t worry, SQL Inject-Me has a list of attack and error strings embedded inside. Type ‘about:config’ in your url bar. Then extensions.sqlime in the filter text box. Attack strings are in “extenions.\*.attacks” and error strings are in “extensions.\*.errorstrings”. Right click on the row with the preference you want to restore and click on “reset”. On some platforms you may have to restart Firefox for it to register the changes. Now when you go to Tools->SQL Inject Me->Options you will see the original strings.