* What is XSS-Me?

XSS-Me is the Exploit-Me tool used to test for reflected Cross-Site Scripting (XSS). It does **NOT** currently test for stored XSS

* How does XSS-Me work?

The tool works by submitting your HTML forms and substituting the form value with strings that are representative of an XSS attack. If the resulting HTML page sets a specific JavaScript value (document.vulnerable=true) then the tool marks the page as vulnerable to the given XSS string. 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 XSS-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](http://gplv3.fsf.org/).

* Does XSS-Me perform source code or network analysis?

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

* What is the target audience of XSS-Me?

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

* Will XSS-Me detect all Cross-Site Scripting (XSS) vulnerabilities?

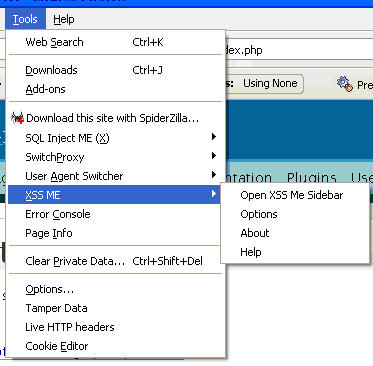
No. XSS-Me detects many types of reflected Cross-Site Scripting in form fields. Unfortunately, there are other methods of launching an XSS attack (e.g. stored cross-site scripting, user- supplied data from other sources such as cookies, links, or HTTP headers). Also, new attack vectors are being discovered all the time. The thoroughness of XSS-Me in testing for reflected XSS is directly related to the completeness of your list of attack strings. By default, the tool ships with a list of XSS attack strings derived from [RSnake’s XSS cheat sheet](http://ha.ckers.org/xss.html)as of September, 2007. RSnake’s list is considered the de-facto list of XSS attack strings, but it is possible that malicious people have attack vectors not known in the security industry. There are also many different types of encodings that may evade filtering tools; the current version of XSS-Me does not have built-in support for multiple encodings. Multiple encodings are planned in a future release. You can help keep XSS-Me up to date by submitting new attack strings.

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

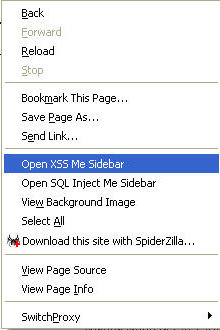
Absolutely No. Neither Security Compass, nor any third party, maintains data on testing results

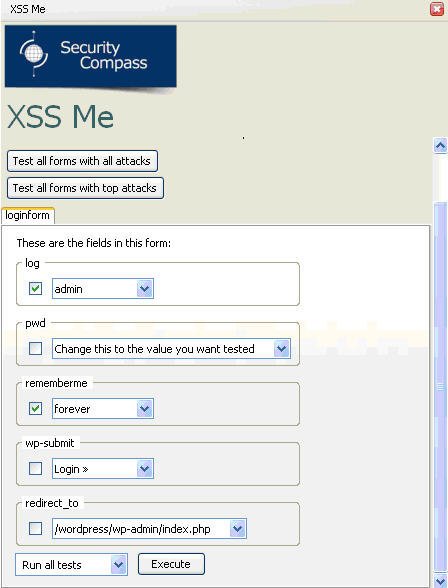
* How do I run XSS-Me?

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 -> XSS-Me -> Open XSS Me Sidebar.



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

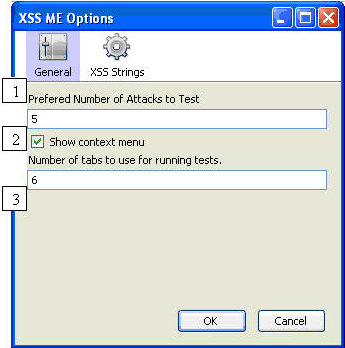


All the forms in your current web page will appear in a series of tabs in the sidebar, and each tab will have all the corresponding visible and hidden fields listed. 

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 “pwd” field in the above example). If you check the box next to a field name, then that field will be tested for Cross-Site Scripting (XSS). If the box is not checked, then the field will not be tested for XSS and the current value listed in the combo box will be submitted every time. XSS-Me works by testing each checked value one at a time. In the above example the tool would attempt to test the “log” field and then the “rememberme” field for XSS. The parameters for the submission would look something like: log=XSS\_ATTACK\_STRING&pwd=&rememberme=forever&wp-submit=Login >>&redirect\_to=/wordpress/wp-admin/index.php when the “log” field is being tested and log=admin&pwd=&rememberme=XSS\_ATTACK\_STRING&wp-submit=Login >>&redirect\_to=/wordpress/wp-admin/index.php when the “rememberme” field is being tested. The tool will substitute XSS\_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 comprehensive but generally allows you to test much quicker, depending on how many of the attacks you have specified to execute in the “top attacks” parameter (see what are the Options 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 XSS-Me?

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

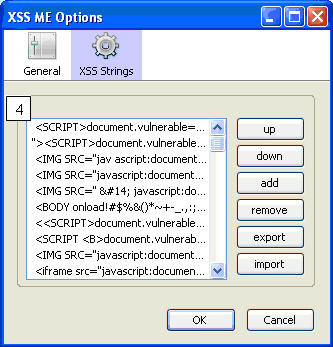


Preferred Number of Attacks to Test This specifies the number of attacks that should be tested when you select the “Test All Forms with Top Attacks” or “Run Top X Attacks” options in the XSS-Me sidebar. If you enter “5″ for this value, then the first 5 values listed in the “XSS-Strings” table will be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

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

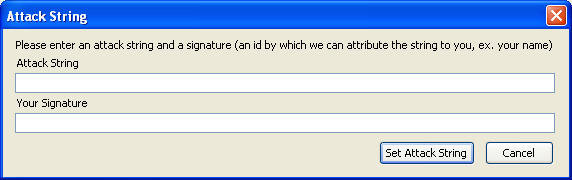
Number of Tabs to Use for Running Tests This specifies how many concurrent tabs can be opened to run the XSS tests. More concurrent tabs may mean quicker overall testing, but will also incur greater memory impact. Opening too many concurrent tabs may cause Firefox to crash.

XSS Strings XSS-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 options.



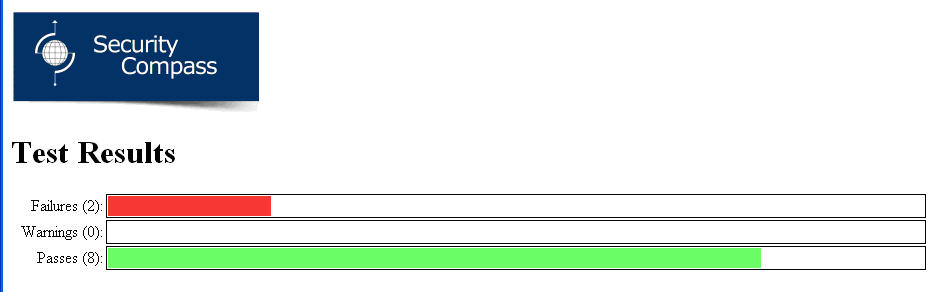
* How do I add my own signatures to the files?

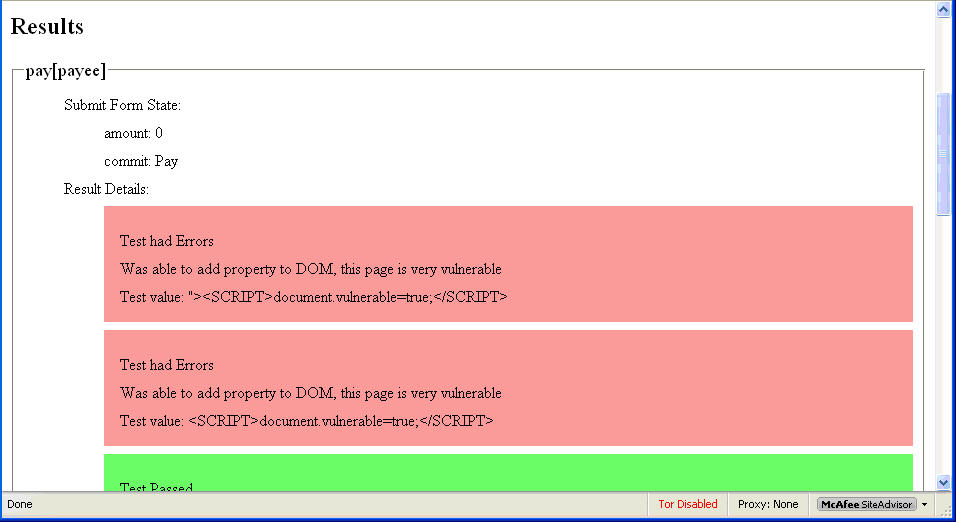
Use the XSS-Strings option in the Tools->XSS-Me->Options menu. Enter the attack in the “Attack String” text field. Note that your attack string should run “document.vulnerable=true” in the resulting JavaScript for the tool to work properly. E.g.<script>document.vulnerable=true</script> is a valid XSS-Me 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 XSS-Me results?

XSS-Me has three result types: Failures The number of tests that certainly resulted in detection of reflected XSS Warnings Number of tests that may have resulted in detection of cross-site scripting (i.e. they did not change the DOM object in Firefox as specified in the JavaScript, but they may result in a successful attack on a different browser Pass Number of tests that did not result in detection of reflected XSS. 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 JavaScript (e.g. alert(“XSS”)) to manually verify.





* Why does my form or field have no name on the XSS-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 XSS-Me sidebar.

* I deleted the default attack strings but I want to get them back.

Don’t worry, XSS-Me has a list of attack strings embedded inside. Type ‘about:config’ in your url bar. Then type ‘extensions.xssme’ in the filter text box. Attack strings are in “extenions.\*.attacks”. 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->XSS Me->Options you will see the original strings.

* Why am I getting a message about a bug 420025? What is that?

Bug 420025 is an issue that XSS Me sometimes experiences. The precise cause of it is currently unknown though a reproducible test case exists. We’ve discussed the issue on Mozilla’s bugzilla and have tried various suggested workarounds to no avail. We currently consider the root cause of this somewhere in Firefox (possibly in the DOM code).