**Introduction:**

[Selenium](http://openqa.org/selenium) provides a record/playback tool for authoring tests without learning a test scripting language. Selenium provides a test domain specific language (DSL) to write tests in a number of popular programming languages, including Java, Ruby, Groovy, Python, PHP, and Perl.

|  |  |
| --- | --- |
| Selenium tests run **directly in a browser**, just like real users do. It runs in Internet Explorer, Mozilla and Firefox on Windows, Linux, and Macintosh, Safari on the Mac.  They have plans to target Safari on the iPhone in some months. The tool is free and available under Apache 2.0. |  |

The quickest way to learn Selenium is via a Firefox plugin called [Selenium IDE](http://selenium.openqa.org/index.html#Selenium_IDE). It is quite compelling   
for developing tests in and quickly trying out Selenium before choosing Selenium for your project.

There are two modes of operation for Selenium - [Core](http://selenium.openqa.org/index.html#Selenium_Core)and [Remote Control (RC)](http://selenium.openqa.org/index.html#Selenium_Remote_Control).  Remote Control   
mode also has a related capability called [Selenium Grid](http://selenium.openqa.org/index.html#Selenium_Grid) that allows you to throw hardware at tests   
to make it all faster.

**This tool is able to:**

* Simulate any action a human user may do either with the help of the keyboard or the mouse,   
  this go from entering a text to select values in select list.
* These workflow can be save and replay at any time and any speed.
* You may group a set of tests and form a test suite very easily.
* Export tests to Ruby, Python, Perl, Java .Net to run them in a non graphical environment ().

**Selenium is made of 3 components: (follow the links for downloads)**

* [Selenium Core](http://www.openqa.org/selenium-core/) : the core must be installed on your server where the web applications are running.
* [Selenium IDE](http://www.openqa.org/selenium-ide/) (FF) : is a Firefox/Mozilla extension able to record, execute tests and test suites
* [Selenium Remote Control](http://www.openqa.org/selenium-rc) (IE): is a server which let you execute tests targeting many different browser, Firefox, Internet Explorer, opera and different operating system GNU/Linux,Mac OS and  MS Windows in also many different languages Ruby, Python, Perl, Java .Net.

Also don't use Selenium for load testing web application, use Apache JMETER instead.

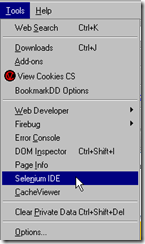
**Quick Start to writing Test with IDE**

**Features**:

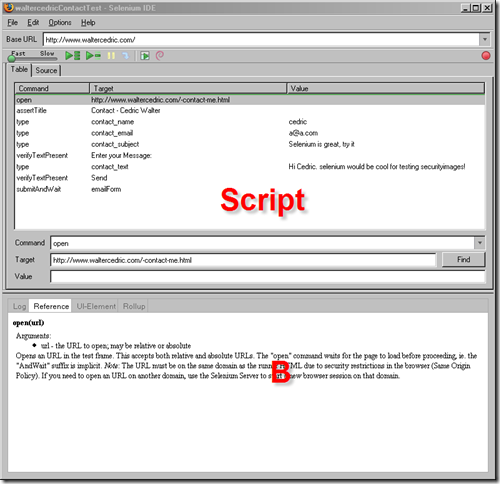
* Easy record and playback.
* Intelligent field selection will use IDs, names, or XPath as needed.
* Auto complete for all common Selenium commands.
* Walk through tests.
* Debug and set breakpoints.
* Save tests as HTML, Ruby scripts, or any other format.
* Support for Selenium *user-extensions.js* file.
* Option to automatically assert the title of every page.

To make it work with any other web applications, just install the Firefox plugins, and start it by going to the tools menu of Firefox.

**Sample Test Case 1:** Let’s say that we want to test the contact page of my site for proper operations...In Firefox, go to the menu **Tools**

[](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/selenium1.png)

This will open floating windows, which let you define a script step by step in the windows **Script**

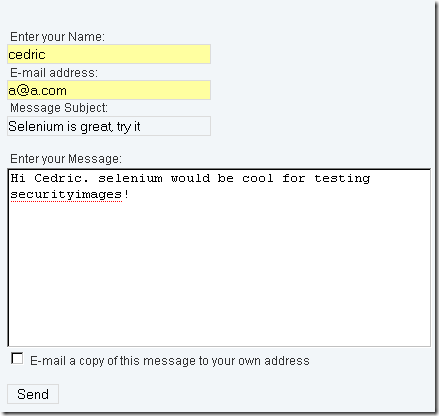
[](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/seleniumjoomlatest.png)

The base URL is my site ([http://www.waltercedric.com](http://www.waltercedric.com/)), the test case, open the contact page, check the   
title of the page, enter some values in form, check for button and texts and submit the form.

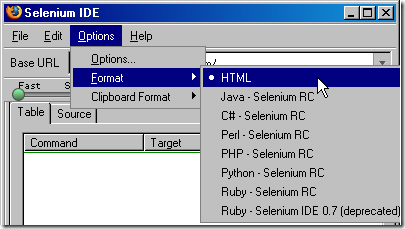
The menu [selenium2](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/selenium2.png)  let you run the test by clicking on [selenium3](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/selenium3.png)you can see the result, if everything is green then the test has succeed.

[](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/selenium4.png)

and you see every step of the test case in the browser windows:

[](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/selenium6.png)

Looks easy to develop a script to test a page, test can be saved on disk in different format, so can execute them in Selenium Core

[](http://www.waltercedric.com/images/stories/DevelopwebtestcasesusingSeleniumIDEinFir_12DA7/selenium5.png)

**VIDEO –**

**A small video presentation to record and execute a test with IDE – good demo** [http://wiki.openqa.org/download/attachments/400/**Selenium**+IDE.swf](http://wiki.openqa.org/download/attachments/400/Selenium+IDE.swf)?

The good news is that if you're using Firefox - you can pretty much bypass this limitation thanks to the wonderful Firefox extension called [Selenium IDE](http://openqa.org/selenium-ide/) (used to be called Selenium Recorder)

At this point, you can run the test in the standard Test Runner mode, by clicking for example on the Run button in the Test Runner Control Panel.  
  
Note however the URL of the test:  
chrome://selenium-ide/content/selenium/TestRunner.html?test=/content/PlayerTestSuite.html&userExtensionsURL=&baseURL=http://www.google.com/  
  
The URL doesn't use the http protocol, but instead uses the chrome protocol specific to Firefox. The fact that Selenium IDE is a Firefox extension give it access to the chrome protocol and allows it to get around the JavaScript XSS security limitation.  
  
At this point, you can also save the test by choosing File->Save Test or pressing Ctrl-S in the Selenium IDE. The test will be saved as an HTML file.  
  
Selenium IDE also allows you to test both http and https within the same application. Normally, without the Selenium IDE, you need to choose at the start of your test whether you want to test pages accessible via http, or pages accessible via https. Once you make this choice, you can't switch from http to https or from https to http within the same test, because of the dreaded JavaScript XSS security limitation. With Selenium IDE, however, you can open both kinds of pages within a single test.  
  
An important note: all this Selenium IDE hocus-pocus does lock you in on Firefox. As soon as you start testing pages that are not under your application root, or you start mixing http and https in your tests, you will not be able to run the same tests under IE, Camino, Safari or other browsers -- at least not until people come up with browser-specific extensions that will get around the XSS limitation.  
  
If you want your tests to be portable across browsers, you can still use Selenium IDE to create the tests. You just need to make sure you test pages that are within your application, and that use the same protocol throughout the test. After saving the tests as local HTML files, you need to copy them over to the Selenium installation that you have deployed on the server hosting the application under test. In any case, I urge you to start using Selenium IDE. It will give you a major productivity boost in writing Selenium tests.

**Selenium on the Server**

Aside from using the Selenium IDE to test WebPages, selenium itself can be installed on the server. This provides a central location for testers and developers to view and execute tests. "Server" here refers to the OpenACS instance where your application is running. If your OpenACS instance is in http://your\_openacs\_instance, then selenium should be installed in http://your\_openacs\_instance/selenium

**WARNING:** We do not recommend installing selenium on the server of a production instance. Only install selenium on a staging or test server. This is because

* When tests fail, they leave a mess of test data that you will not want on production

To install selenium on the server:

1. Download selenium from <http://www.openqa.org/selenium-core/download.action>
2. Choose to download the "Full Release"
3. After downloading, decompress the file.
4. You will see a folder selenium-x.x where "x.x" is the version of selenium you downloaded.
5. Go inside this folder and look for the "selenium" folder.
6. Copy this "selenium" folder to the openacsroot/www/ directory
7. Launch a browser and go to http://your\_openacs\_instance/selenium

At this point, selenium is installed. The page that you see when you visit /selenium is the default selenium page. It lists the tests that come with selenium.

To start using selenium on the server :

1. Customize the landing page /selenium/index.html
2. Create Test Suites for each package. A test suite is just a file in selenium/tests that lists a number of tests for a particular feature or section of an application. For instance, TestSuite-News.html would be a file that lists the tests for the News Package.
3. Upload tests to the /selenium/tests folder.

**Advantages:**

1. It is a Freeware
2. Simple, Easy to install, Easy to work
3. Selenium IDE is the only flavor of Selenium which allows you to record user action on browser window
4. Can also record user actions in most of the popular languages like Java, C#, Perl, Ruby
5. It will not record any operation that you do on your computer apart from the events on Firefox browser window
6. During recording if you right click on any element it will show all the selenium commands available.
7. The test cases can be created just by browsing through the application. Not even single line of coding required.
8. Test cases can be exported to multiple languages. For Java it means it gets imported as JUNIT test case.
9. Now they can be run as standalone JUNIT test cases on any machine.
10. JUNIT test cases can be further modified by developers as per need.

**Limitations:**

1. The IDE is limited to Firefox
2. Did not get satisfactory results for dynamic contents. If text is changing dynamically at run time, the Selenium was not able to recognize. This might require more analysis.
3. Pop-ups: If the window has a pop up, say ok cancel button, the tool was not able to handle it smoothly. More analysis required here.
4. We can’t run recorded script if it is converted to Java, C#, Ruby etc.
5. Not allowed to write manual scripts like conditions and Loops for Data Driven Testing

**Using Selenium with Internet Explorer (IE)**: Using Selenium with IE is not as straight forward as it is with FF, still possible. We need to create test cases manually in this case and then execute.

One way to create test cases is as mentioned above, run the application in FF with Selenium IDE and export the test cases in Java.

Secondly we can actually create the test cases manually. This is as not very tough, but certainly more work than using tool with FF. We would need to understand the tool in more depth and figure out Java commands which it uses internally and then create appropriate test cases.

**Feature Matrix:**

**Selenium IDE Selenium RC Selenium Core Selenium Core HTA**

Browser Support Firefox only Many All IE only

Requires Remote Installation No No Yes No

Supports HTTPS/ SSL Yes Yes Yes Yes

Supports Multiple Domain Yes Yes No Yes

Requires Java No Yes No No

Save Test/ Result to Disk No Yes No Yes

Language Support Many Many Selenese only Selenese only

**References & Helpful Links:**

<http://www.jroller.com/selenium/>

<http://seleniumhq.org/docs/>

<http://googletesting.blogspot.com/2009/06/my-selenium-tests-arent-stable.html> - Blog