

文章编号:1003-5850(2012)04-0054-03

使用 Selenium 进行 Web 应用自动化测试的研究

黄华林

(广东女子职业技术学院, 广州 511450)

摘要:介绍了利用 Web 自动化测试工具 Selenium 进行 Web 自动化测试的情况,通过一个具体案例来阐述自动化测试的过程,并对测试结果进行了简要分析,说明了利用 Selenium 进行 Web 应用系统的自动化测试的好处,给应用工程人员提供了指导。

关键词:selenium, 自动化测试, Web 测试, 测试脚本, 录制与回放

中图分类号: TP311.56

文献标识码: A

A Study of Web Applications Automated Testing Using Selenium

HUANG Hua-lin

(Guangdong Women's Professional Technical College, Guangzhou 511450, China)

Abstract: In this paper described the use of selenium for web applications automated testing, and the automated testing process through a concrete case, Then we give a brief analysis of test results and present the benefits of web applications automated testing using selenium and thus provide guidance to the engineering staff.

Key words: selenium, automated testing, Web test, test scripts, record and replay

随着 Internet 的迅速发展,越来越多的应用系统由传统的 C/S 结构转向 B/S 结构,也出现了许多大型 Web 应用系统,如办公自动化、网络电子商务等。这些 Web 应用系统规模日趋庞大,复杂度也日益增加,对 Web 应用的开发提出了更高的要求,同时,也增加了 Web 应用测试的工作量和难度^[1]。传统的测试技术并不完全适用于 Web 应用的测试,而手工测试效率低且难以保证测试覆盖率,使用 Web 自动化测试工具是测试人员的不二选择。

软件自动化测试就是使用自动测试工具或手段,按照测试工程师的预定计划进行自动测试来验证各种软件测试的需求,包括测试活动的管理与实施,目的是减轻手工测试的工作量,提高软件的质量^[2]。由于 Web 自动化测试的强大需求,近年来出现了不少商业的或开源的 Web 测试工具,如 QTP、PesterCat、Selenium、Watin、Watij、Watir、WebInject、Canoo

WebTest、OperaDriver、Tellurium、Slimdog 等等^[3]。这些测试工具都有各自特点和各自的应用范围,其中 Selenium 是一个应用较广的开源 Web 自动化测试框架。

1 Selenium 介绍

Selenium 是 ThoughtWorks 公司人员开发的一套基于 WEB 应用的测试工具,它使用 JavaScript 和 Iframes 在浏览器中嵌入自动化测试引擎。通过编写模仿用户操作的测试脚本,测试人员可以从终端用户的角度来对 Web 应用程序进行黑盒测试。Selenium 测试可以直接在浏览器内运行,它支持多种浏览器,以及这些浏览器的多个版本^[4]。

Selenium 是一套开源的自动化测试框架,包含 Selenium IDE、Selenium Remote Control (RC) 和 Selenium Grid 等几个工具,可以用于单元测试、回扫

* 收稿日期:2011-12-13,修回日期:2012-02-27

** 黄华林,男,1981年生,讲师,硕士,研究方向:计算机技术应用、软件测试。

测试、冒烟测试、集成测试、验收测试等。

1.1 Selenium IDE

Selenium IDE 是用于 Selenium 测试的一个集成测试工具,被嵌套在 Firefox 浏览器中,作为 Firefox 浏览器的一个组件来使用。通过 Selenium IDE,测试人员可以录制在 Web 界面上的一切操作,并进行编辑,调试和快速回放,就像用户在操作一样。

1.2 Selenium RC

Selenium RC 是 Selenium 中的最主要的测试工具,可以使用 Java、Javascript、Ruby、PHP、Python、Perl 或 C# 等语言编写脚本,它通过 Selenium RC 服务器作为代理服务器去访问 Web 应用从而达到测试的目的。Selenium RC 的使用包括服务器和脚本两部分。服务器是一个已经打包好的 JAVA 程序,在运行测试脚本时需先启动。脚本则需要测试人员通过 Selenium IDE 来录制或利用 Selenium 提供的接口与类来编写了。Selenium RC 不需要依附 Firefox 浏览器,可以在很多浏览器上进行测试。

1.3 Selenium Grid

Selenium Grid 是基于 Selenium RC 的自动化测试辅助工具,它可以让 Selenium RC 的自动化测试脚本在多个服务器上并行执行,大大节省运行测试的时间。

2 使用 Selenium 进行 Web 测试

以 21CN 邮箱的登陆页面作为测试目标,研究 Selenium 的自动化测试应用。测试目标页面地址为 <http://mail.qq.com>,界面如图 1 所示。

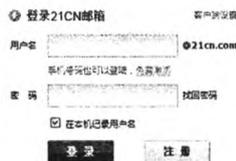


图 1 21CN 邮箱登陆界面图

2.1 使用 Selenium IDE 录制脚本

Selenium IDE 是一个 Firefox 浏览器插件,须事先下载并安装到 Firefox 浏览器中。在 Firefox 浏览器中打开待测网页 <http://mail.21cn.com>,执行 Firefox 浏览器的 [工具]-[Selenium IDE] 菜单命令,调出 Selenium IDE。在 Selenium IDE 的 Base URL 中同样填上待测网页地址 <http://mail.21cn.com>,接着在 Selenium IDE 中新建一个名为“T21cnLogin”的测试用例,然后点击 Selenium IDE 界面的录制按钮即进入

脚本录制。

在 Firefox 浏览器的 <http://mail.21cn.com> 网页中进行正常的邮箱登陆操作,填写正确的用户名和密码,点击登陆按钮,进入邮箱。这个操作过程自然被 Selenium IDE 录制下来了。停止 Selenium IDE 的脚本录制,可看到 Selenium IDE 做到了 UI 元素的定位,包括 ID 为“UserName”的文本框、ID 为“passwd”的文本框和 ID 为“Submit”的按钮,并且捕获了文本框的输入、按钮的点击的交互操作,如图 2 所示。

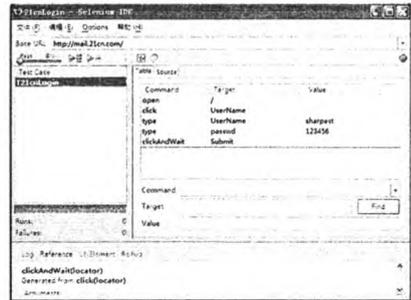


图 2 Selenium IDE 脚本录制界面图

2.2 转换脚本代码

Selenium IDE 可以将录制的脚本转换成 C#、Java、PHP 等语言。在 Selenium IDE 中执行 [Options]-[Format]-[Java(JUnit)] 命令即可轻松将脚本转换成 Java 语言的脚本,转换后的代码如下。但是,在实际使用 Selenium RC 执行这些脚本前,是需要对这些脚本稍作修改的。

```
package com.example.tests;
import com.thoughtworks.selenium.*;
import java.util.regex.Pattern;
public class T21cnLogin extends SeleneTestCase {
    public void setUp() throws Exception {
        setUp (" http://change-this-to-the-site-you-are-testing/", " * chrome");
    }
    public void testT21cnLogin() throws Exception {
        selenium.open("/");
        selenium.click("UserName");
        selenium.type("UserName", "sharpest");
        selenium.type("passwd", "123456");
        selenium.click("Submit");
        selenium.waitForPageToLoad("30000");
    }
}
```

2.3 使用 Selenium RC 进行测试

使用 Selenium RC 进行 Web 测试时,是客户端利

用各种编程语言,通过网络向 Selenium Server 发送指令,Selenium Server 接收到测试指令后,启动浏览器并向其发出 JavaScript 调用实现对 Html 页面的全面追踪,并通过网络把执行结果返回给调用者^[5]。

Selenium 客户端一般使用单元测试技术实现,通过判断返回的结果与预期是否一致来判定程序是否运行正确。

2.3.1 Selenium RC 测试环境

① Selenium 提供了 Selenium 服务器 selenium-server.jar,进行测试前必须先启动 Selenium 服务器,可以通过命令行 java -jar selenium-server.jar 来启动 Selenium 服务器。而 Selenium RC 实际上就是 Selenium 提供的 selenium-java-client-driver.jar。

② 打开 Java IDE(如 Eclipse),新建一个项目,在项目的 build path 里面加上 junit.jar 和 selenium-java-client-driver.jar。

③ 把 Selenium IDE 录制并转换好的 java 脚本文件,导入到项目中,根据 Selenium 提供的接口与类对脚本文件稍作修改,然后就能在 Java IDE 中执行测试脚本和查看测试报告了。

2.3.2 测试脚本

本测试脚本是把 Selenium IDE 录制并转换好的 java 脚本文件稍作修改而来,整个测试脚本代码如下:

```

package com.example.tests;
import com.thoughtworks.selenium.*;
import org.junit.After;
import org.junit.Before;
import org.junit.Test;
import java.util.regex.Pattern;
public class T21cnLogin extends SeleneTestCase {
    @Before
    public void setUp() throws Exception {
        selenium = new DefaultSelenium (" localhost",
4444, " * Firefox", "http://mail.21cn.com/");
        selenium.start();
    }
    @Test
    public void testT21cnLogin() throws Exception {
        selenium.open("/");
        selenium.click("UserName");
        selenium.type("UserName", "sharpest");
        selenium.type("passwd", "123456");
        selenium.click("Submit");
        selenium.waitForPageToLoad("30000");
        selenium.selectWindow("欢迎使用 21CN 个人邮箱");

```

```

Assert.assertTrue(selenium.isTextPresent("您好"));
}
@After
public void tearDown() throws Exception {
    selenium.stop();
}
}

```

2.3.3 测试结果及分析

执行以上测试脚本时,Selenium RC 启动浏览器打开被测网页 http://mail.21cn.com,自动在用户名文本框和密码文本框中分别填入“sharpest”和“123456”,并自动点击“登陆”按钮,等待新开网页窗口的载入,然后自动查找标签为“欢迎使用 21CN 个人邮箱”的网页窗口,在该窗口中查找字符串“您好”。由于成功登入 21CN 邮箱页面是有“您好.....”的欢迎信息,字符串匹配成功,断言成立,测试通过。如果修改上述测试脚本中的测试用例,自动输入的是错误密码“666666”,再去执行测试脚本,得到的是一个失败测试。

此外,TestNG 测试框架是可以并发执行测试用例的,Selenium Grid 结合 TestNG 测试框架可以实现并行自动化测试,能极大地提高测试效率。

3 结束语

Selenium 是一套开源的 Web 应用测试工具,具有脚本录制功能,而且可很方便地把脚本转换成 C#、JAVA、PHP 等常见的编程语言,生成这些语言开发环境下的测试用例。通过将该工具与持续集成工具相结合,测试人员就可以高效地进行 Web 自动化测试,提高 Web 应用程序的质量。Selenium 提供了一套低成本高效率的 Web 测试方案,灵活运用 Selenium 将给相关企业带来效益上的巨大提升。

参考文献:

- [1] 许 蕾,徐宝文,陈振强. Web 测试综述[J]. 计算机科学,2003,30(3):100-104.
- [2] 程 泓,朱文兴. Web 应用程序自动化测试工具[J]. 莆田学院学报,2010,17(2):52-56.
- [3] 陆 璐,王柏勇. 软件自动化测试技术[M]. 北京:清华大学出版社,2006.
- [4] 温素剑. 零成本实现 Web 自动化测试——基于 Selenium 和 Bromine [M]. 北京:电子工业出版社,2011.
- [5] 《开源技术选型手册》编委会. 开源技术选型手册 [M]. 北京:电子工业出版社,2008.