**[压力测试之badboy和Jmeter的简单使用方法](https://www.cnblogs.com/liyuanhong/articles/5880730.html)**

**转载**

原文地址：[http://www.cnblogs.com/naaoveGIS/p/5476881.html](http://www.cnblogs.com/naaoveGIS/p/5476881.html" \t "_blank)

**1.背景**

所谓压力测试是指，通过确定一个系统的瓶颈或者不能接收的性能点，来获得系统能提供的最大的服务级别的测试。通俗地讲，压力测试是为了发现在什么条件下您的应用程序的性能会变得不可接受。通常进行压力测试均是测试人员的工作，但是研发人员或多或少也需要对压力测试有个基本的了解。这里我跟大家一起学习如何使用两个常用工具进行压力测试，即badboy和Jmeter。

**2.badboy简介及使用方法**

**2.1badboy简介**

Badboy是用C++开发的，被用于测试和开发复杂的动态应用。它提供了强大的屏幕录制和回放功能，同时也提供了丰富的图形结果分析功能。只要不用于商业目的就可以免费使用。因此这两工具的结合，就成为了绝配。我们可以用Badboy录制脚本，然后将录制的脚本导出为JMeter格式的脚本，最后将该脚本导入到JMeter，借助于JMeter强大的测试功能模拟大量的虚拟用户，进行复杂的性能测试。

其下载地址为：[http://www.badboy.com.au](http://www.badboy.com.au/)。下载完后需要进行安装，安装过程同一般的Windows 应用程序没有什么区别，安装完成后你可以在桌面和Windows开始菜单中看到相应的快捷方式。如果找不到，可以找一下[Badboy](http://www.badboy.com.au/)安装目录下的[Badboy](http://www.badboy.com.au/).exe文件，直接双击启动[Badboy](http://www.badboy.com.au/)。最后开到的启动界面如下：



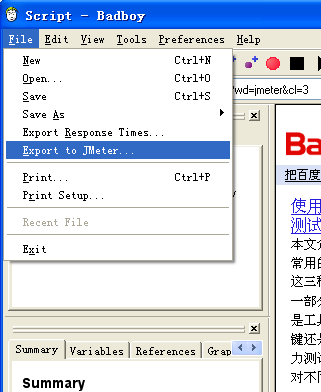
**2.2使用方法**

a.在地址栏（图中红色方框标注的部分）中输入你需要录制的Web应用的URL，这里我们以[http://www.baidu.com](http://www.baidu.com/)为例。

b.点击“开始录制”按钮（图中蓝色圆圈标注的部分）开始录制 。

c.开始录制后，你可以直接在[Badboy](http://www.badboy.com.au/)内嵌的浏览器（主界面的右侧）中对被测应用进行操作，所有的操作都会被记录在主界面左侧的编辑窗口中（图中黄色方框标注的部分）。在这个试验中，我们在baidu的搜索引擎中输入 [JMeter](http://jakarta.apache.org/jmeter/) 进行搜索。不过录制下来的脚本并不是一行行的代码，而是一个个Web对象——这有点像LoadRunner的VuGen中的Tree View视图。

d.录制完成后，点击工具栏中的“停止录制”按钮（图中紫色方框标注的部分），完成脚本的录制。



e.选择“File -> Export to [JMeter](http://jakarta.apache.org/jmeter/)”菜单，填写文件名“baidu.jmx”，将录制好脚本导出为[JMeter](http://jakarta.apache.org/jmeter/)脚本格式。也可以选择“File -> Save”菜单保存为[Badboy](http://www.badboy.com.au/)脚本。

f.启动[JMeter](http://jakarta.apache.org/jmeter/)并打开刚刚生成的测试脚本，就可以用JMeter进行测试了。

**3.Jmeter的简介和使用方法**

**3.1Jmeter的简介**

Apache JMeter，是Apache组织开发的基于Java的针对功能和性能的测试工具。最初JMeter是为Web/HTTP测试而设计的，但是后来它已经扩展到支持各种各样的测试模块。JMeter可以用于测试静态或者动态资源的性能（如静态文件、Java服务程序、Servlet、CGI脚本、Perl脚本、Java对象、数据库和查询、FTP服务器或者其他资源）。JMeter可以用于模拟对服务器、网络或对象加以巨大的负载，在不同压力类别下测试它们的强度，分析整体性能。另外，JMeter能够对应用程序做功能/回归测试，通过创建带有断言的脚本来验证你的程序返回了你期望的结果。为了最大限度的灵活性，JMeter允许使用正则表达式创建断言。同时它也提供了一个可替换的界面用来定制数据显示，测试的同步及测试的创建和执行。

其下载地址为: [http://jakarta.apache.org](http://jakarta.apache.org/)。双击D:/JMeter/bin下的jmeter.bat或ApacheJMeter.jar即可开启运行。

**3.2主要组件介绍**

a.测试计划(Test Plan)

   是使用JMeter进行测试的起点，它是其它JMeter测试元件的容器。

b.线程组(Thread Group)

   代表一定数量的并发用户，它可以用来模拟并发用户发送请求。

c.取样器(sampler)

   定义实际的请求内容，被线程组包含，我们主要用HTTP请求。

d.监听器(Listener)

负责收集测试结果，同时也被告知了结果显示的方式。我们常用的包括：聚合报告、察看结果树、用表格查看结果，都支持将结果数据写入文件。其他的添加上去看看就行。



e.逻辑控制器(Logic Controller)

可以自定义JMeter发送请求的行为逻辑，它与Sampler结合使用可以模拟复杂的请求序列。

其分为循环控制器和事务控制器。

f.断言(Assertions)

可以用来判断请求响应的结果是否如用户所期望的。它可以用来隔离问题域，即在确保功能正确的前提下执行压力测试。这个限制对于有效的测试是非常有用的。

g.配置元件(Config Element)

维护Sampler需要的配置信息，并根据实际的需要会修改请求的内容。我们主要在参数化中用到CSV Data Set Config。

h.前置处理器(Pre Processors)和后置处理器(Post Processors)

负责在生成请求之前和之后完成工作。前置处理器常常用来修改请求的设置，后置处理器则常常用来处理响应的数据。我们主要在动态关联中用到后置处理器的正则表达式提取器。

i.定时器(Timer)

负责定义请求之间的延迟间隔。

**3.3两种使用方法**

**3.3.1直接导入脚本的方法**

在前面我们使用badboy进行了脚本录制，这里可以直接将其导入到Jmeter中。具体补步骤如下：

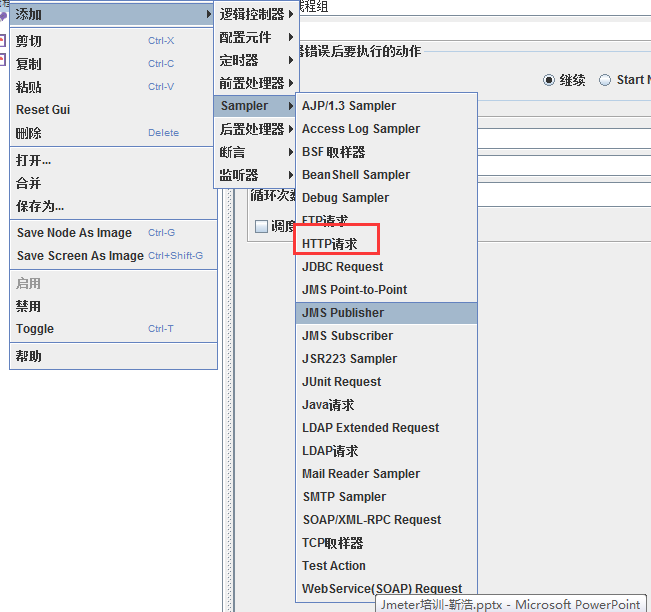
a.打开JMeter会有一个默认的测试计划，点击文件－打开，选中录制的脚本文件如：WebXSample\_addUser.jmx，打开脚本进行测试。

b.在线程组上添加监听器－聚合报告（用于分析测试结果）后，点击运行－启动，开始测试，测试完毕后在聚合报告中就可以看到测试结果。一个简单的测试计划就完成了。

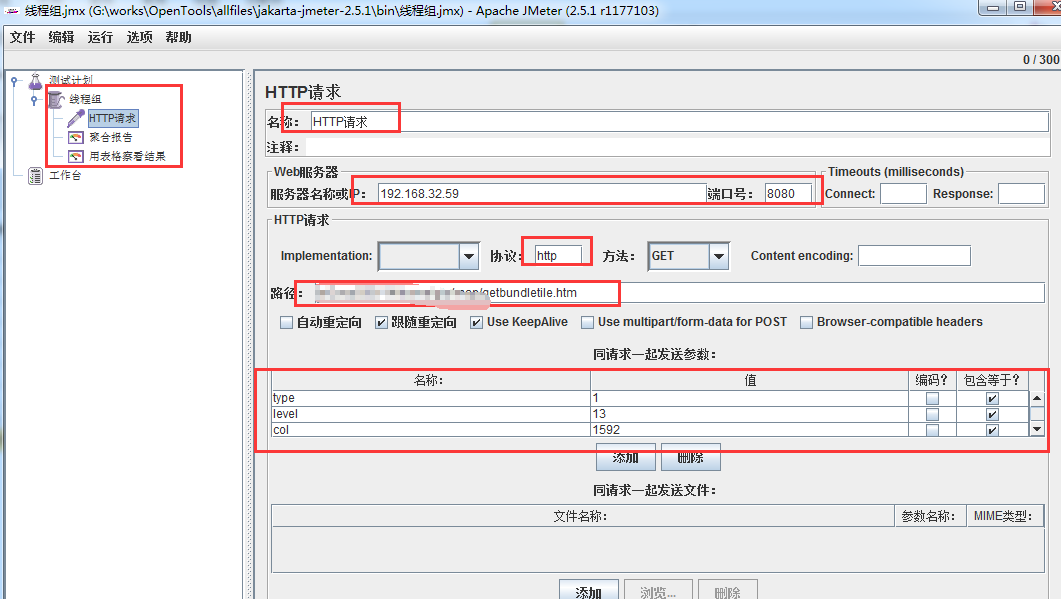


**3.3.2手动编写测试样例的方法**

a.在测试计划中先添加一个线程组，然后在该线程组中加入http请求模块。可以在sampler中看到，Jmeter还能测试数据库连接查询压力，FTP连接压力等。



b.编写测试请求。



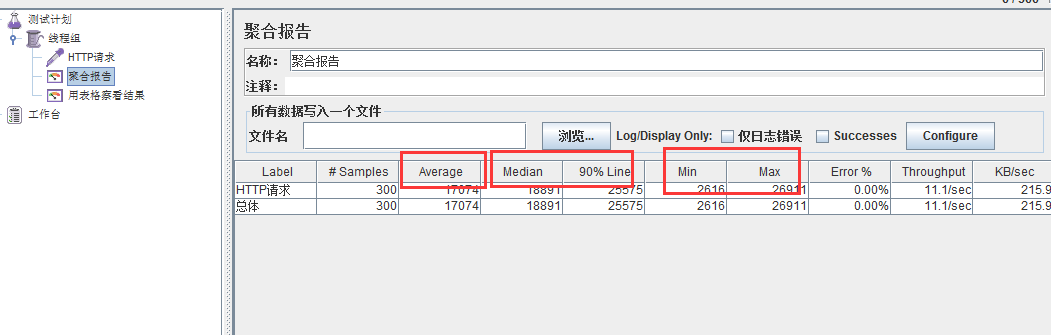
c.添加监听等。

d.设置线程并发量。



f.点击运行，查看运行结果，进行分析。





即300个并发量中，平均时间是17074毫秒，中间段发出的的请求耗时18891毫秒，末尾百分之90处发出的请求耗时25575毫秒。整个测试过程中，最小耗时为2616毫秒，最大耗时为26911毫秒。