

**软件测试课程设计报告**

学 号： 20161001929

班级序号： 111162

学生姓名： 李东晖

指导教师： 张剑波

**中国地质大学信息工程学院软件工程系**

**2019年 4 月**

# 实习二 自动化测试工具

## 1. 概述

### 1.1 测试对象

百度搜索引擎的性能测试，

订票网站功能测试

### 1.2 测试环境

[操作系统](https://www.baidu.com/s?wd=%E6%93%8D%E4%BD%9C%E7%B3%BB%E7%BB%9F&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)：windows

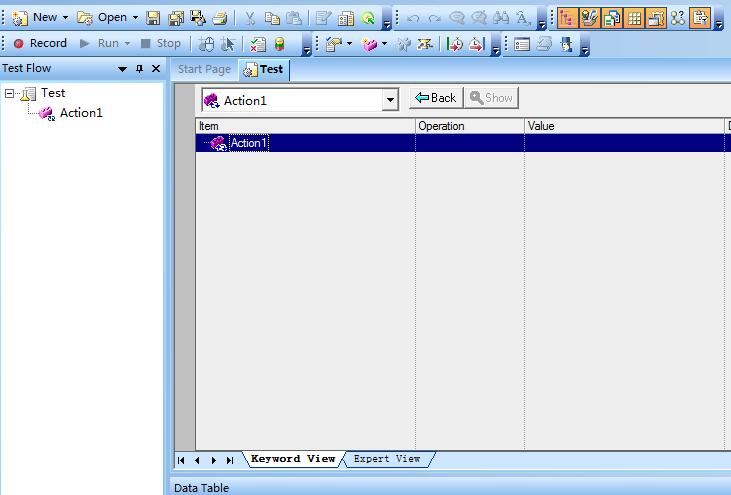
测试工具：QTP，Jmeter

## 2. 测试场景描述

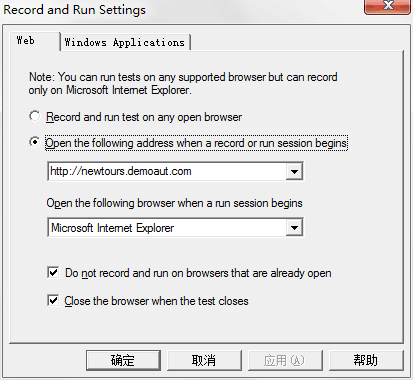
**表3-1 场景用例测试表（\*必填）**

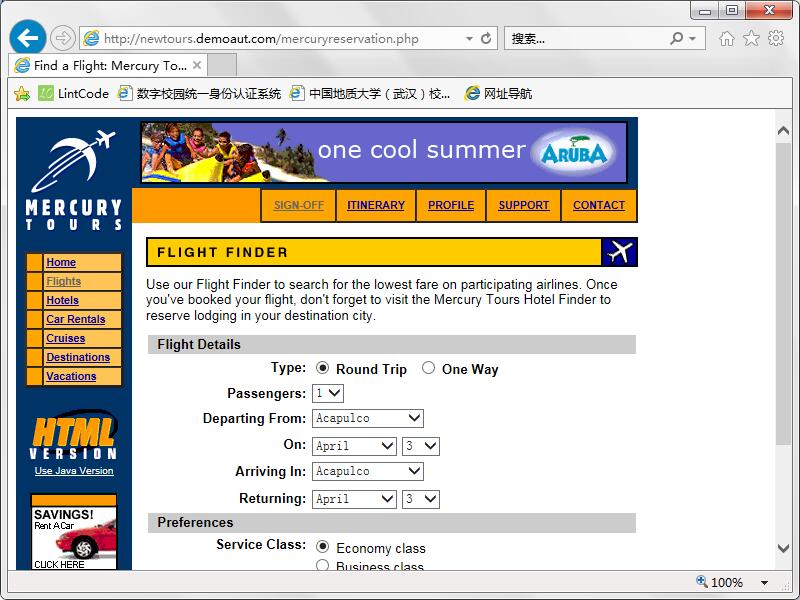
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **用例编号** | **U-1** | **测试优先级** | | **高** |
| **用例设计者** | **管理员** | **设计日期** | | **2019.4.3** |
| **对用需求编号** | **1** |  | |  |
| **测试人员** | **管理员** | **测试日期** | | **2019.4.3** |
| **备注** | **无** | | | |
| **用例描述\*** | **使用QTP进行购票网站测试** | | | |
| **用例的目的\*** | **测试购票网站功能** | | | |
| **前提条件\*** | **无** | | | |
| **场景描述\*** | **期望输出\*** | | **实际情况\*** | |
| **1** | | **1** | |
| **结论\*** | **成功** | | | |

打开qtp，点击New进行项目的创建

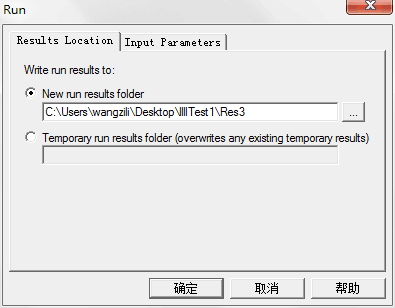


点击Record，进入下界面，点击Open the following address，点击确定进行脚本的录制

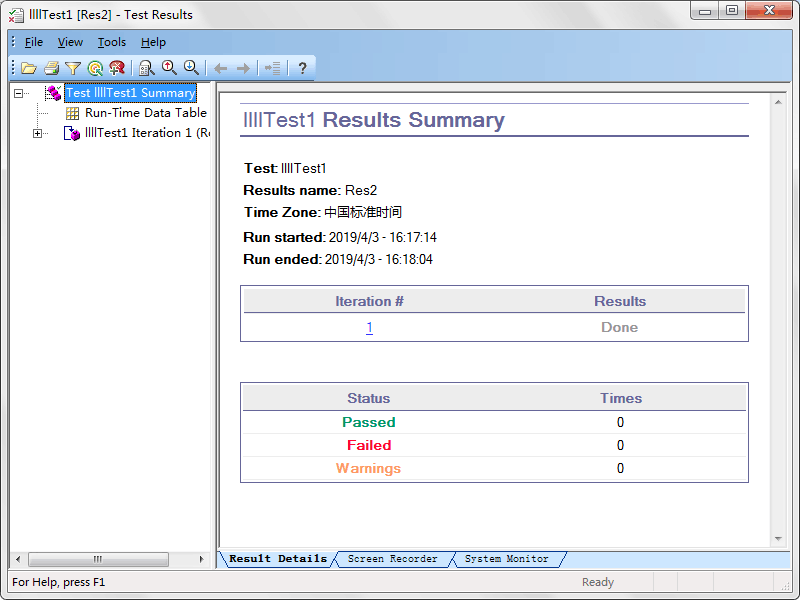




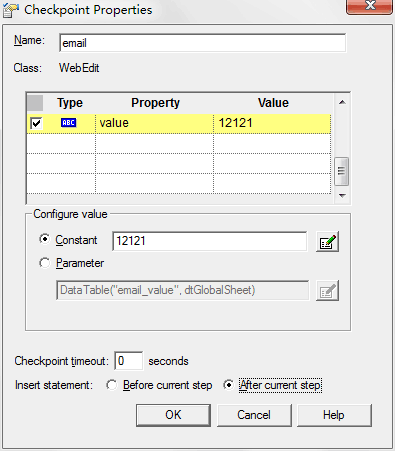
录制成功后点击stop结束录制，点击run进行回放，



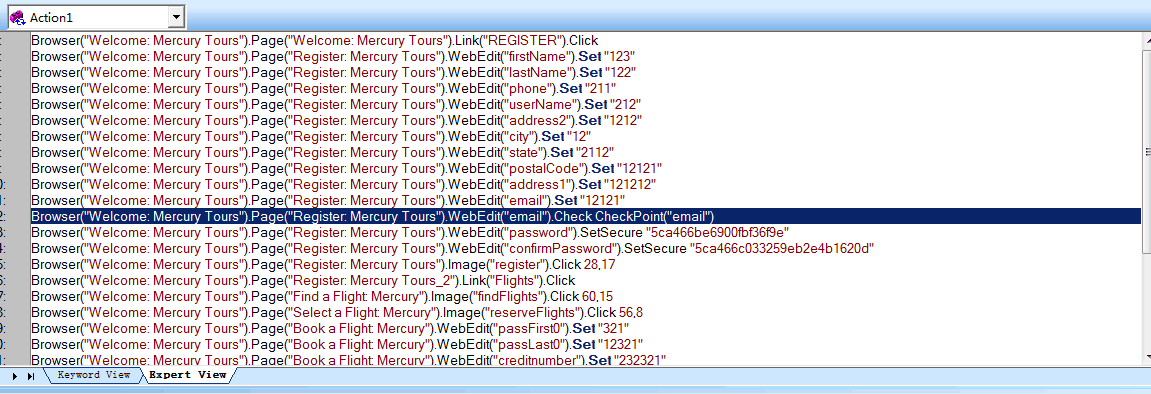
看到结果页面：



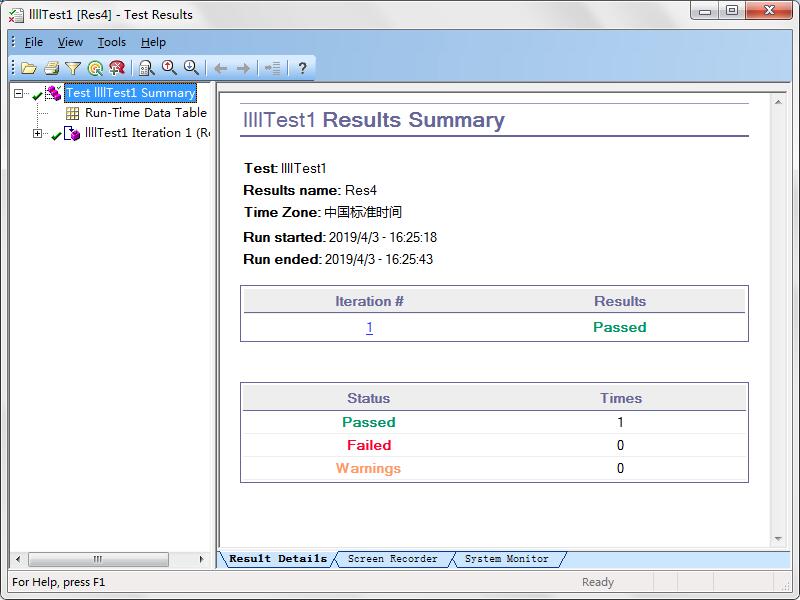
创建检查点：在对应想要插入检查点的位置右键添加检查点，出现下面界面，选择value进行插入检查点。



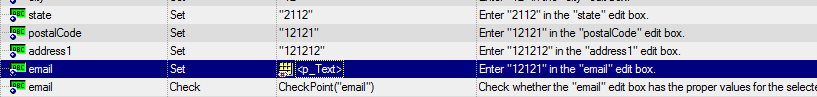
插入成功后会看到在对应的语句下出现了新的语句。

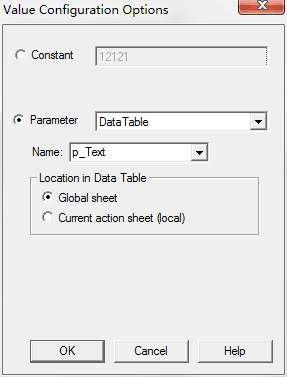


点击run再次运行脚本，成功后结果显示Passed为1，证明检查点插入成功

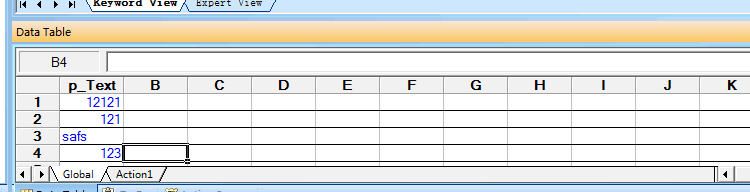


参数化外挂，更改email行的值，改为Data Table

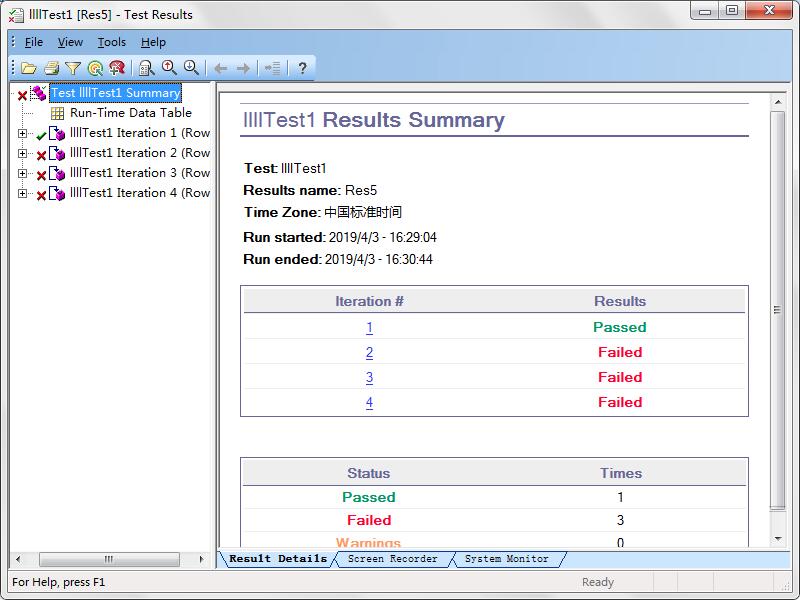




在下面的表格中输入想要的判断的值



再运行可以看到成功一个错误三个。



## 3. 压力测试

### 3.1 压力测试详细步骤

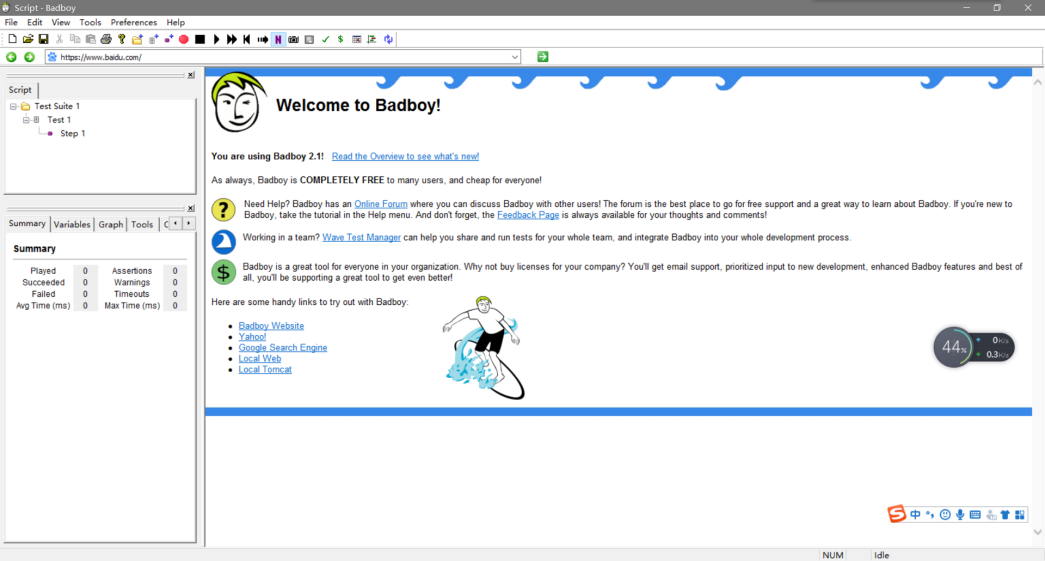
我是使用jmeter和badboy完成的压力测试，通过badboy进行脚本的录制然后用jmeter使用多个线程组进行百度的搜索测试。

1. 脚本录制：
2. Badboy安装：

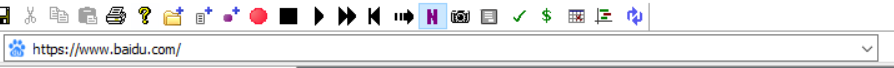
下载badboy安装软件，双击安装



安装完成后进入badboy界面

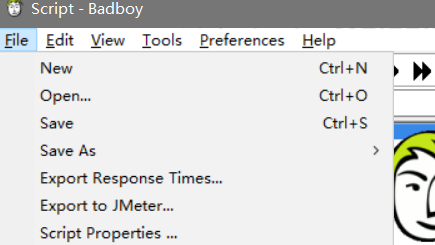


在下方输入网址，点击红色按钮进行录制（一开始打开便是录制状态）



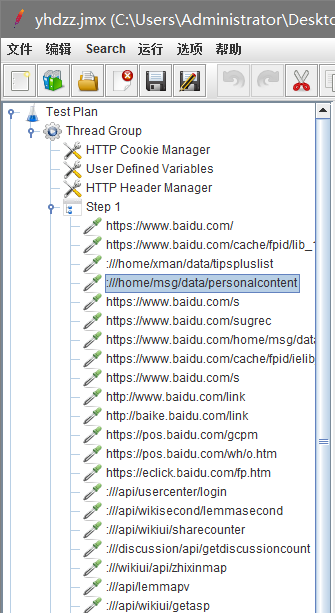


录制完成后再次点击红色按钮结束录制，点击Export to Jmeter将录制的脚本存为jmx文件（jmeter能识别的文件）



1. 打开jmeter导入脚本

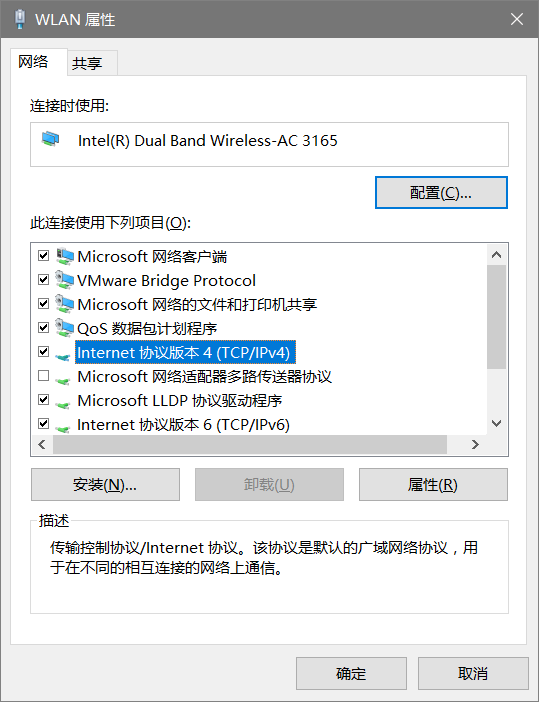
文件打开对应存储的jmx文件文件如下所示,该文件便是录制好的脚本



1. Ip欺骗

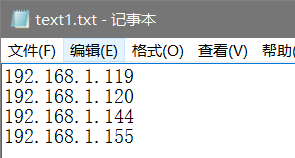
从网络打开属性，点击WLAN，打开属性中的Internet协议版本4（TCP/IPv4）,将自动获取ip地址改为使用下面的ip地址，自己添加ip地址，从高级中添加多个ip地址。



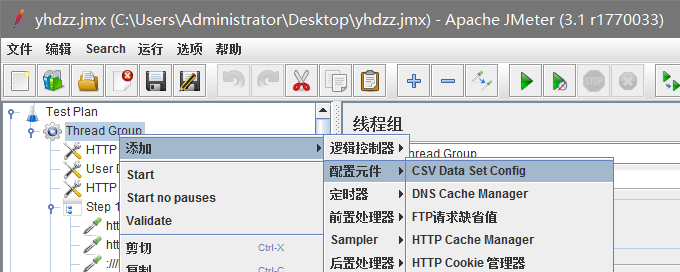




将所有添加的ip地址打包成一个txt文件



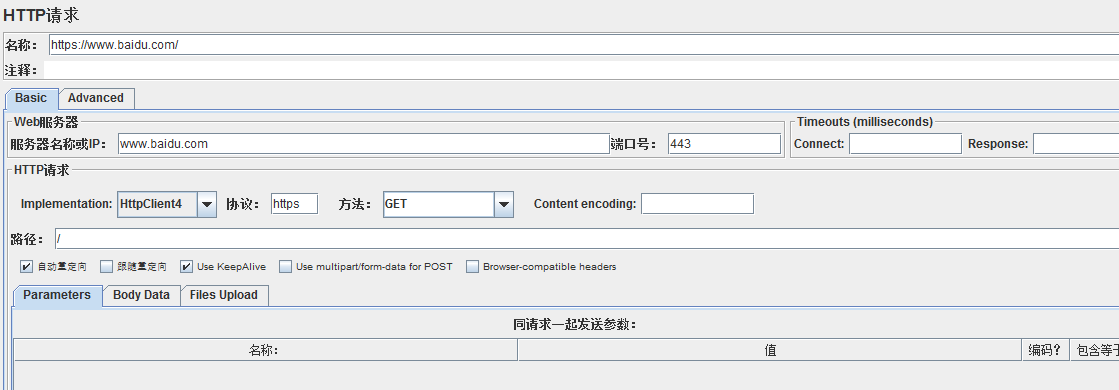
打开jmeter，在线程组->添加->配置元件->CSV Data Set Config

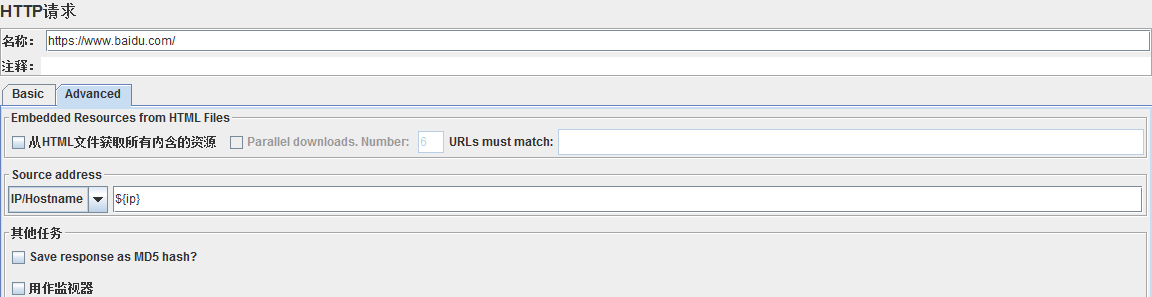


在Filename处填写ip文件的地址，在Variable Names处自定义变量ip，

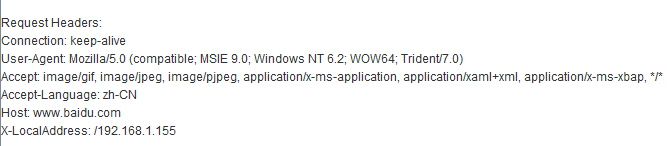


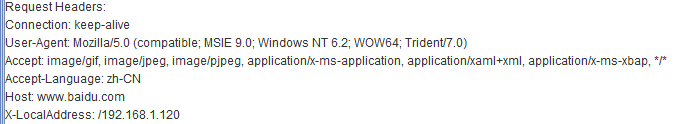
将所有http请求中的Implemtation调成HTTPClient4，在Source address中选择IP/Hostname，在期中填写${ip}({}中的是上面设置的变量)

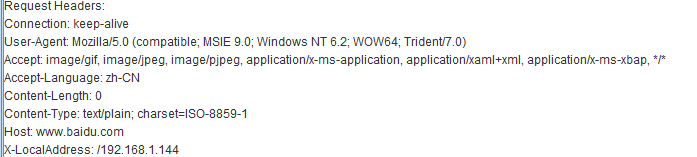




运行，在察看结果树中可以看到有多个ip

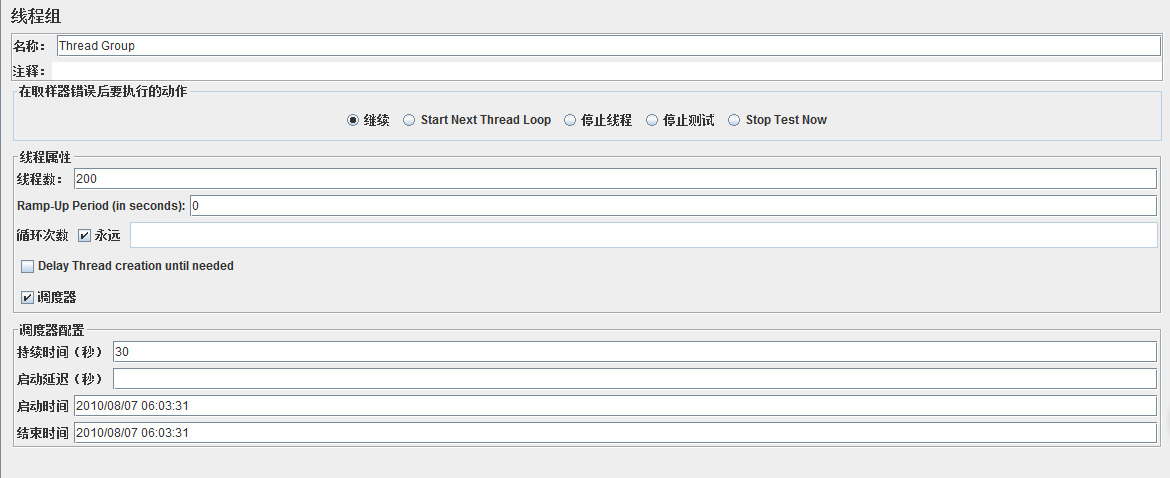






1. 进行压力测试

在线程组中设置200线程数运行30s

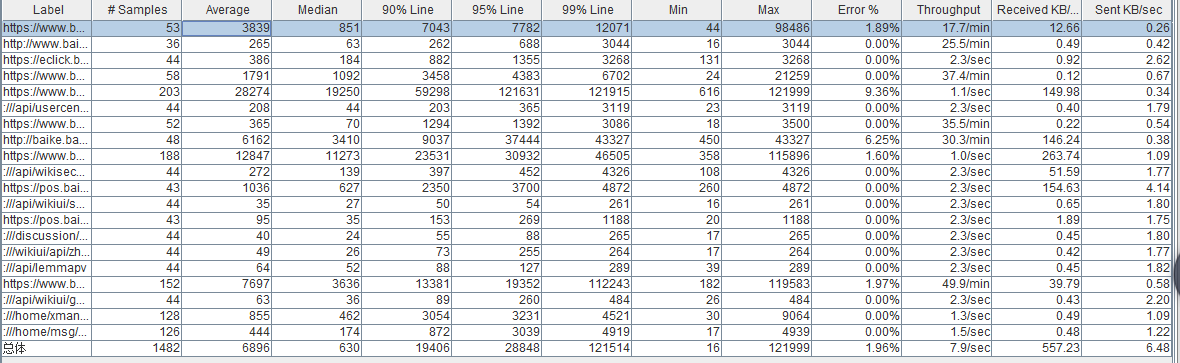


**表3-2 测试中需要记录的数据**

|  |  |
| --- | --- |
| 测试时间 | **30s** |
| 平均响应时间 | **见测试结果** |
| 成功次数 |  |
| 失败次数 |  |
| web服务器CPU利用率（平均、最大） |  |
| 数据库服务器CPU利用率（平均、最大） |  |

### 3.2 测试结果及分析

聚合报告：



Samples -- 本次场景中一共完成了多少个Transaction

Average -- 平均响应时间

Median -- 统计意义上面的响应时间的中值

90% Line -- 所有transaction中90%的transaction的响应时间都小于xx

Min -- 最小响应时间

Max -- 最大响应时间

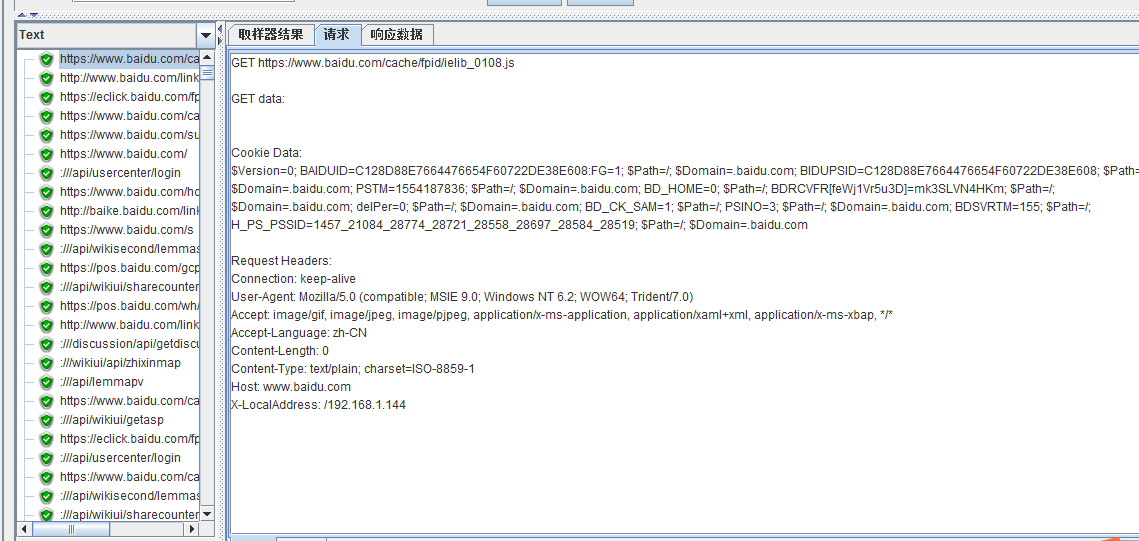
PS: 以上时间的单位均为ms

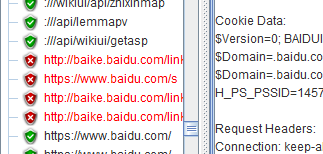
Error -- 出错率

Troughput -- 吞吐量，单位：transaction/sec

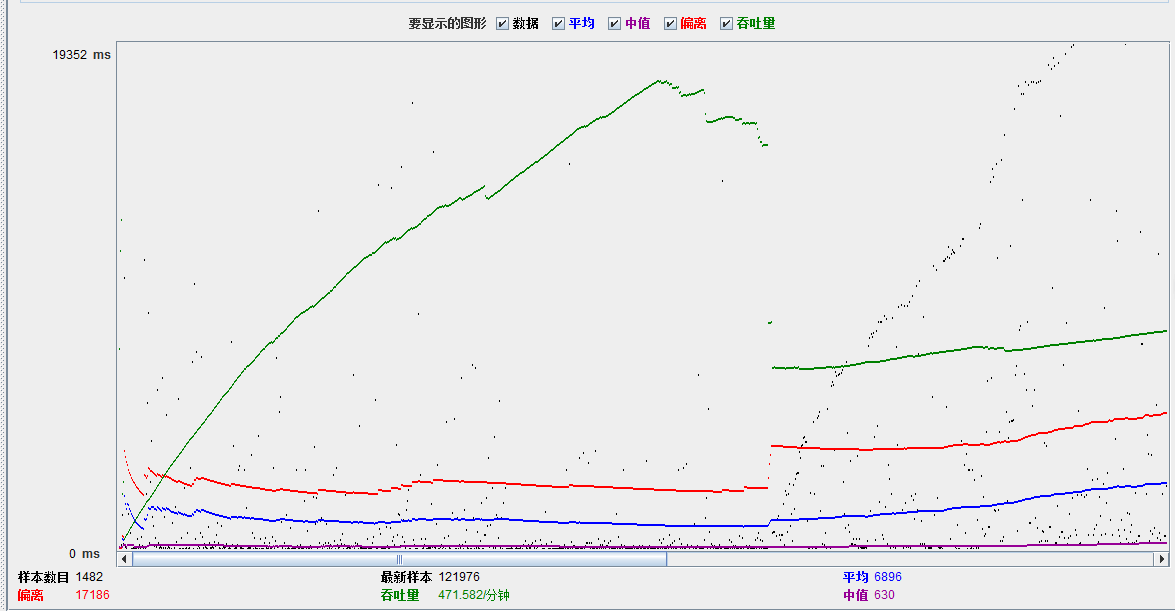
KB/sec -- 以流量做衡量的吞吐量

察看结果树：





图形结果：



200个用户对百度进行30秒的访问，可以从聚合报告中直观的看到所有上述变量对应的数值，出错率中有的不是百分之0，在结果树中可以看到对应的错误事例。在图形结果中可以直观的看到吞吐率等值的变化。随着发送到服务器的请求数越来越多，偏离数量越来越大，服务器越来越不稳定； 发送到服务器的请求数增加，吞吐量先减少后增加。

## 4. 负载测试

### 4.1 负载测试详细步骤

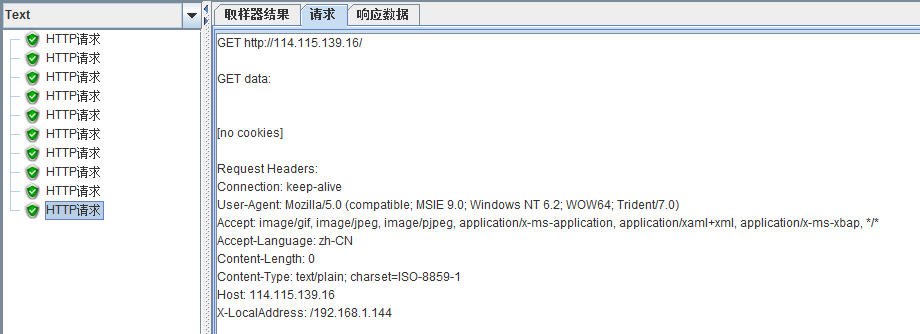
分别使用不同个线程对网站进行访问（访问的网站为室友的自己的网站，承受力不足，比较好测试）

**表3-3 压力测试/负载测试**

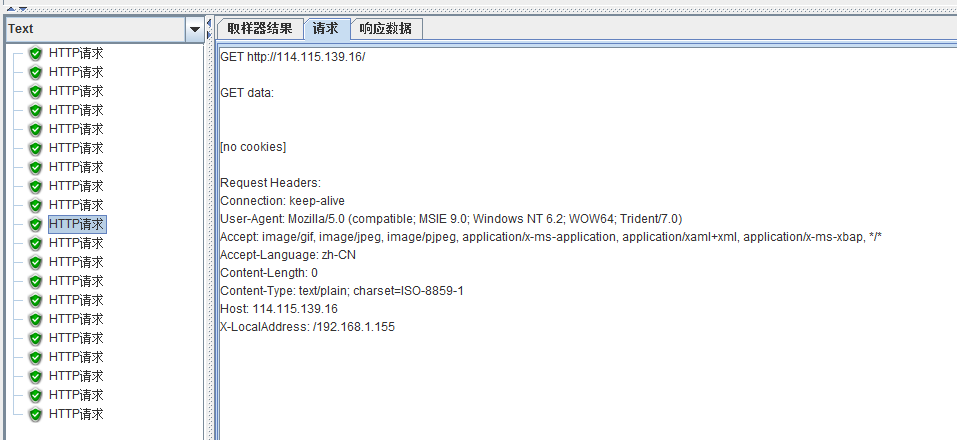
|  |  |  |
| --- | --- | --- |
| 输入/动作 | 输出/响应 | 能否正常工作 |
| 5个用户操作 |  | 能 |
| 10个用户操作 |  |  |
| 20个用户操作 |  |  |
| 30个用户操作 |  |  |
| 50个用户操作 |  |  |
| ……… |  |  |

### 4.2 测试结果及分析

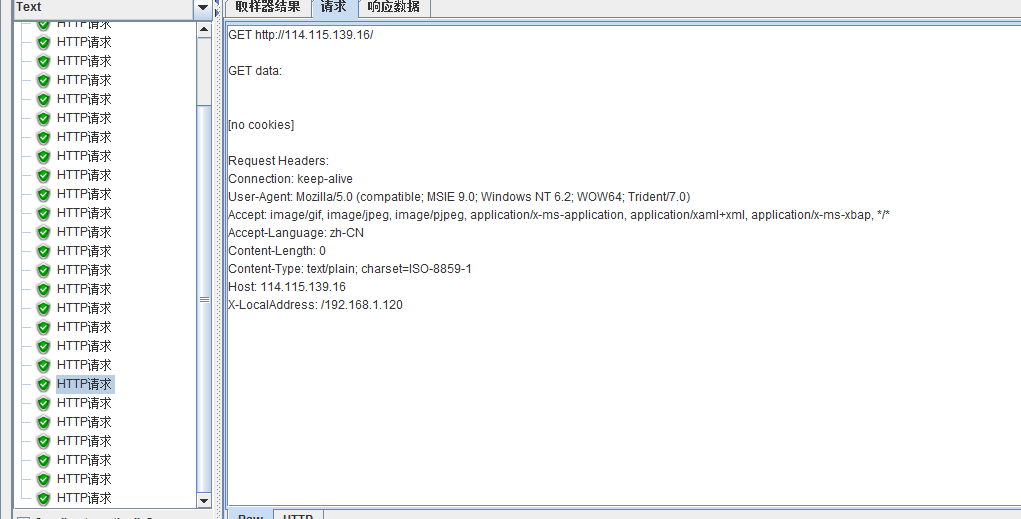
10个用户：能



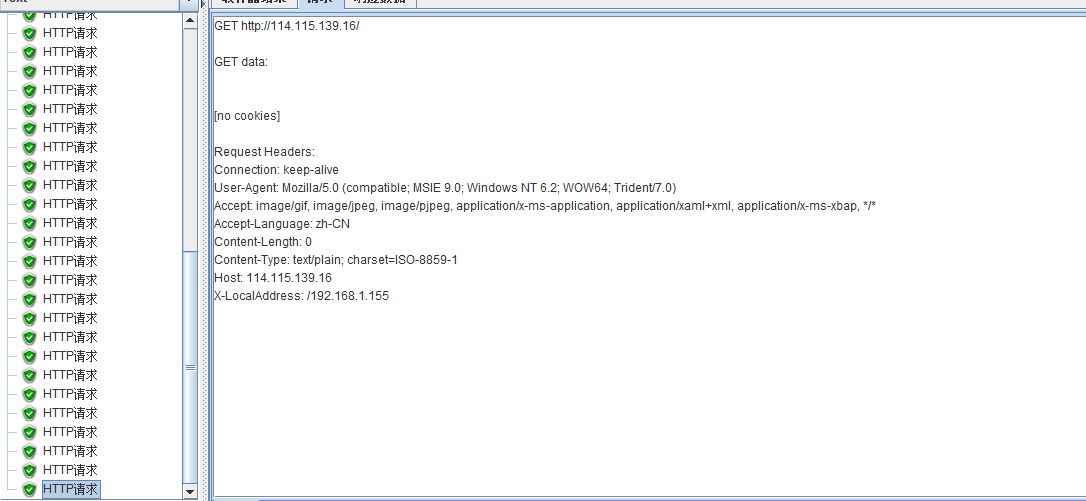
20个用户：能



30个用户：能



50个用户：能



120个用户：爆了

