****

**软件测试技术第三次实验报告**



**学 院 智能与计算学部**

**专 业 软件工程**

**年 级 2016级**

**成员及分工:**

**邓杰 3016218051**

**2019年 4月 14 日**

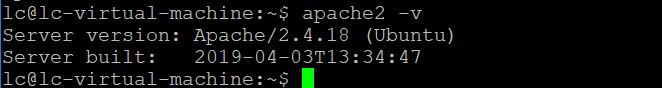
# 软件测试技术第三次实验报告

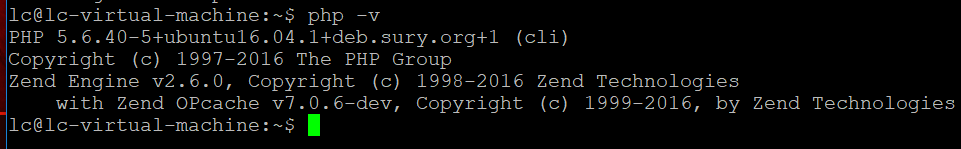
1. 需求分析（描述具体需求）
2. **整体目标：**
   1. 安装虚拟机，并安装一套LAMP（Linux+Apache+Mysql+PHP）待测系统，推荐ECShop（http://www.ecshop.com），基于此进行Jmeter压力测试，并在测试后得出Jmeter测试报告，并根据sysstat得出Linux服务器的CIMN（CPU、IO、Memory以及Network）的性能。
3. **实验要求**
   1. 基本要求：完成任务，实践压力测试工具Jmeter
   2. 相关日期：
      1. 提交报告日期：2019.4.15日按照格式提交
   3. 实验报告应包括：
      1. 报告封皮
         1. 实验名称、团队名称、队长、成员以及每个人完成任务的信息
      2. 实验相关内容：
         1. Linux下top命令结果截图
         2. 访问的B/S系统截图
         3. Jmeter的Testplan展开截图
         4. Beanshell代码
         5. 运行Jmeter测试之后的Aggregate Report Result
         6. 运行Jmeter测试之后的服务器性能截图
4. **参考**
   1. Jmeter：<http://jmeter.apache.org>
   2. LAMP及ECShop安装请自行搜索
5. 概要设计（简单描述设计思路）
6. 详细设计（详细描述具体如何实现，附代码及说明）
7. 安装LAMP

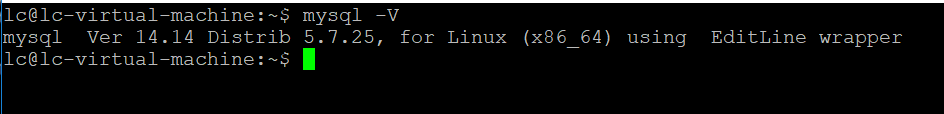
linux服务器选择的是ubuntu

通过相应命令安装Apache，PHP，Mysql

安装成功后查看版本截图：

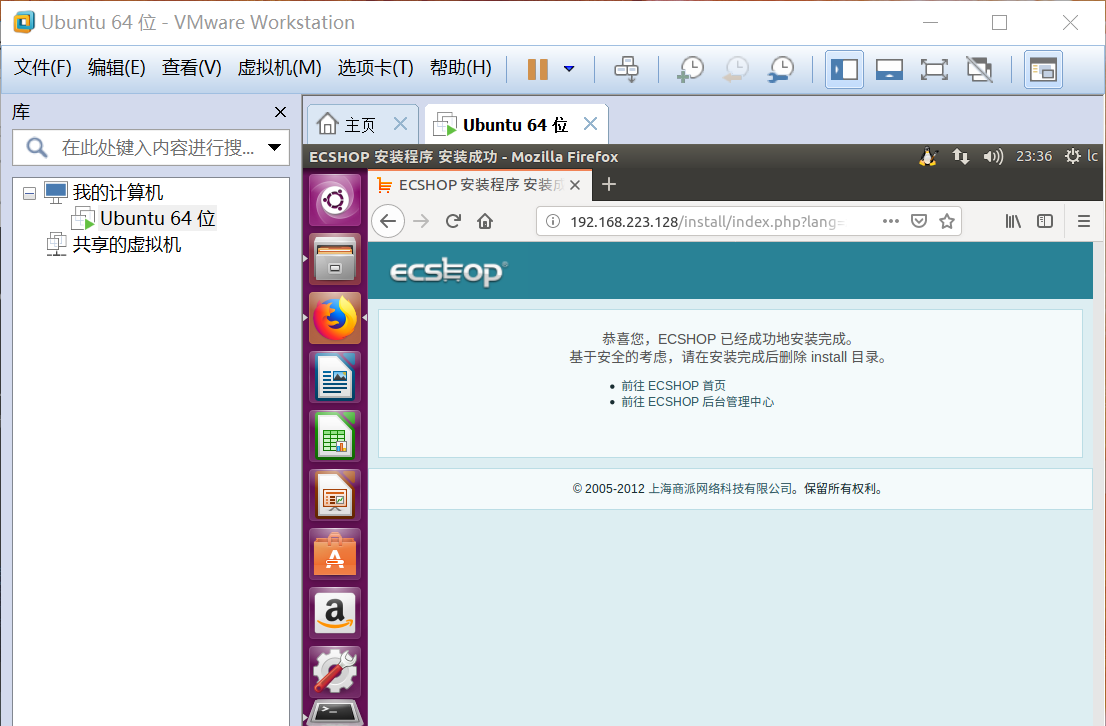






1. ECShop安装

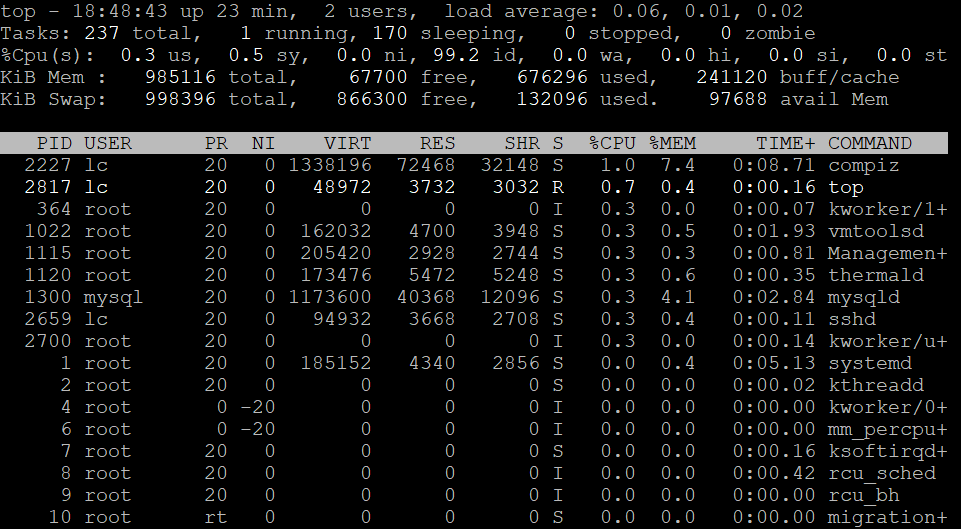
安装成功：



物理机访问ECShop



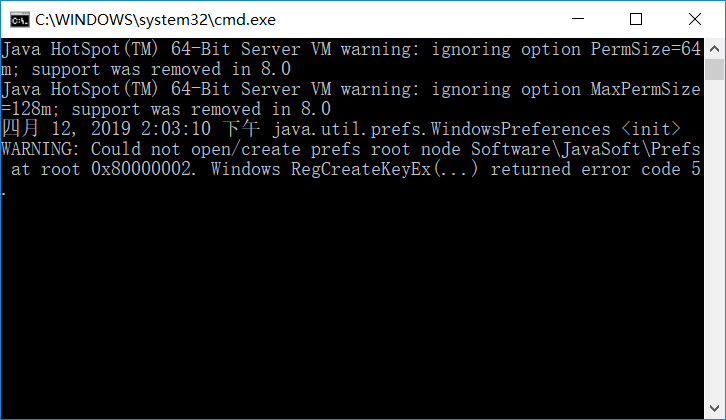
1. top命令截图

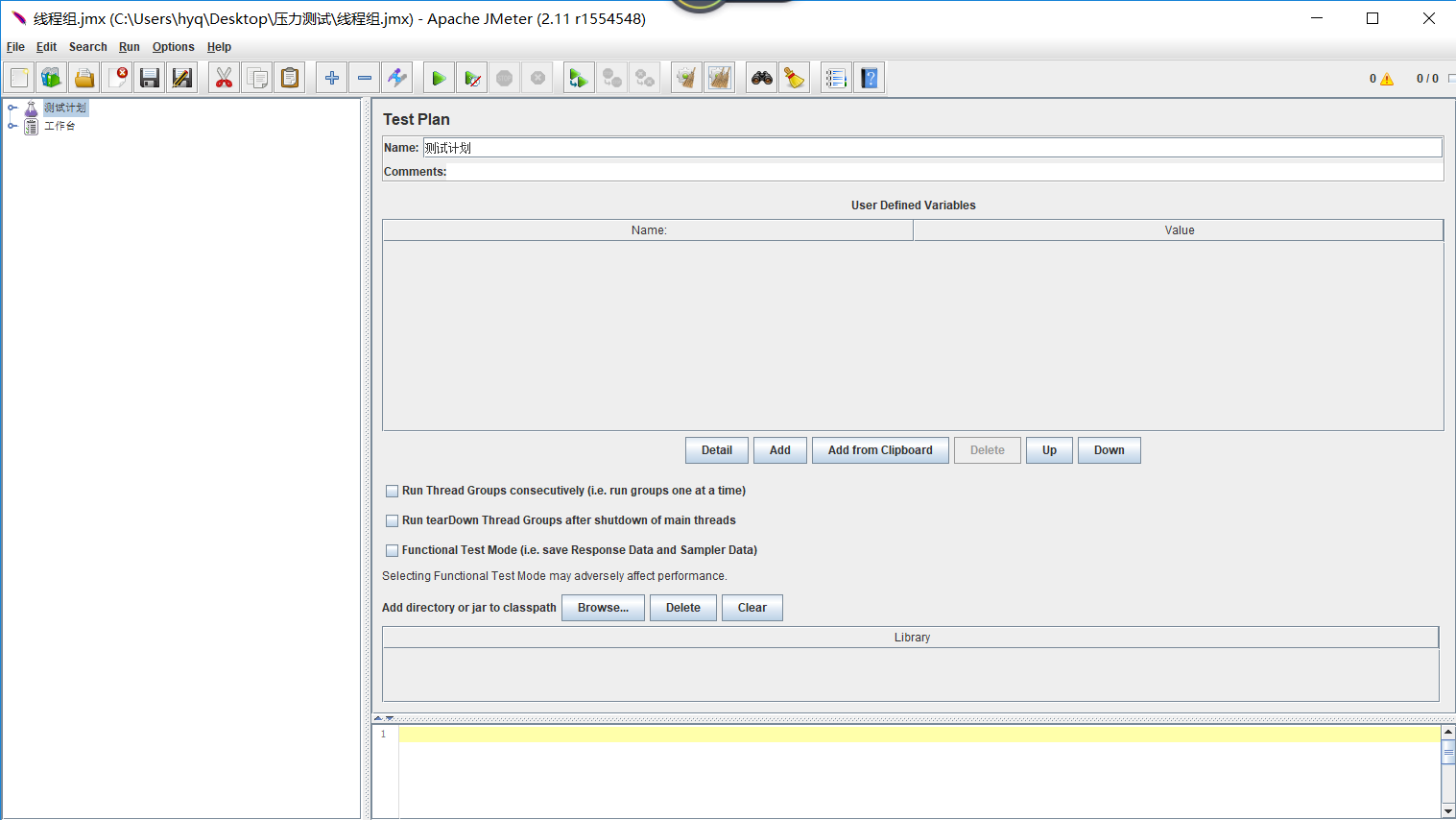


1. 通过Jmeter对ecshop网站进行测试,测试分别在5\*10和50\*10

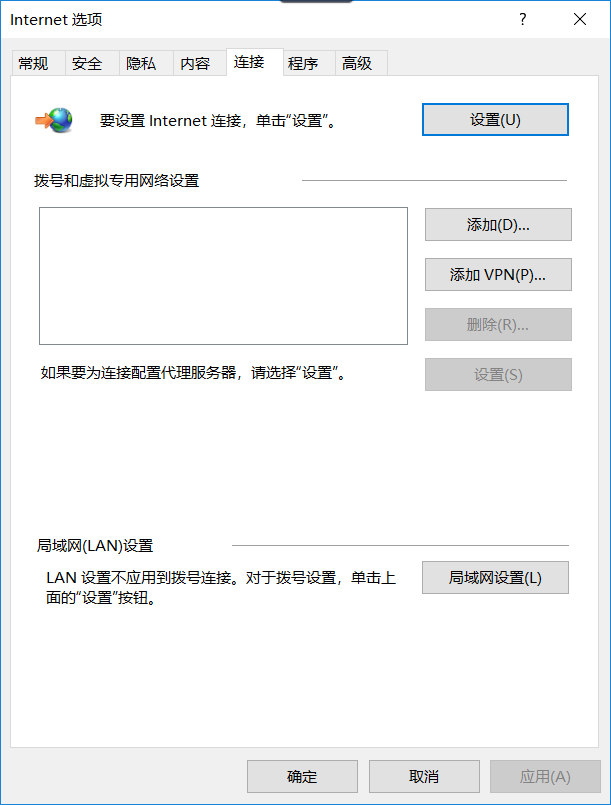
情况下的ecshop主页，登录页面和购买手机页面。

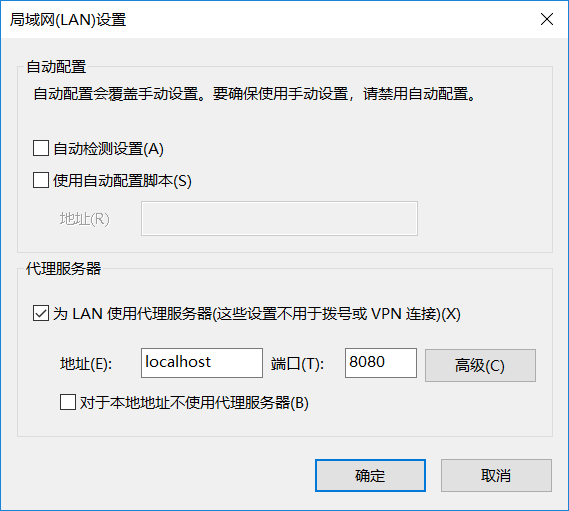
1. 下载并解压Jmeter2.11压缩包，然后开启jmeter.bat文件



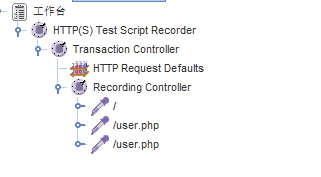


1. 先进行录制，通过jmeter的代理服务器访问ecshop网站，然后访问其中的登录和注册页面。

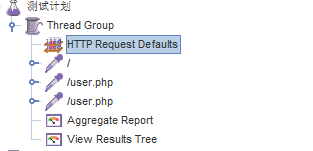


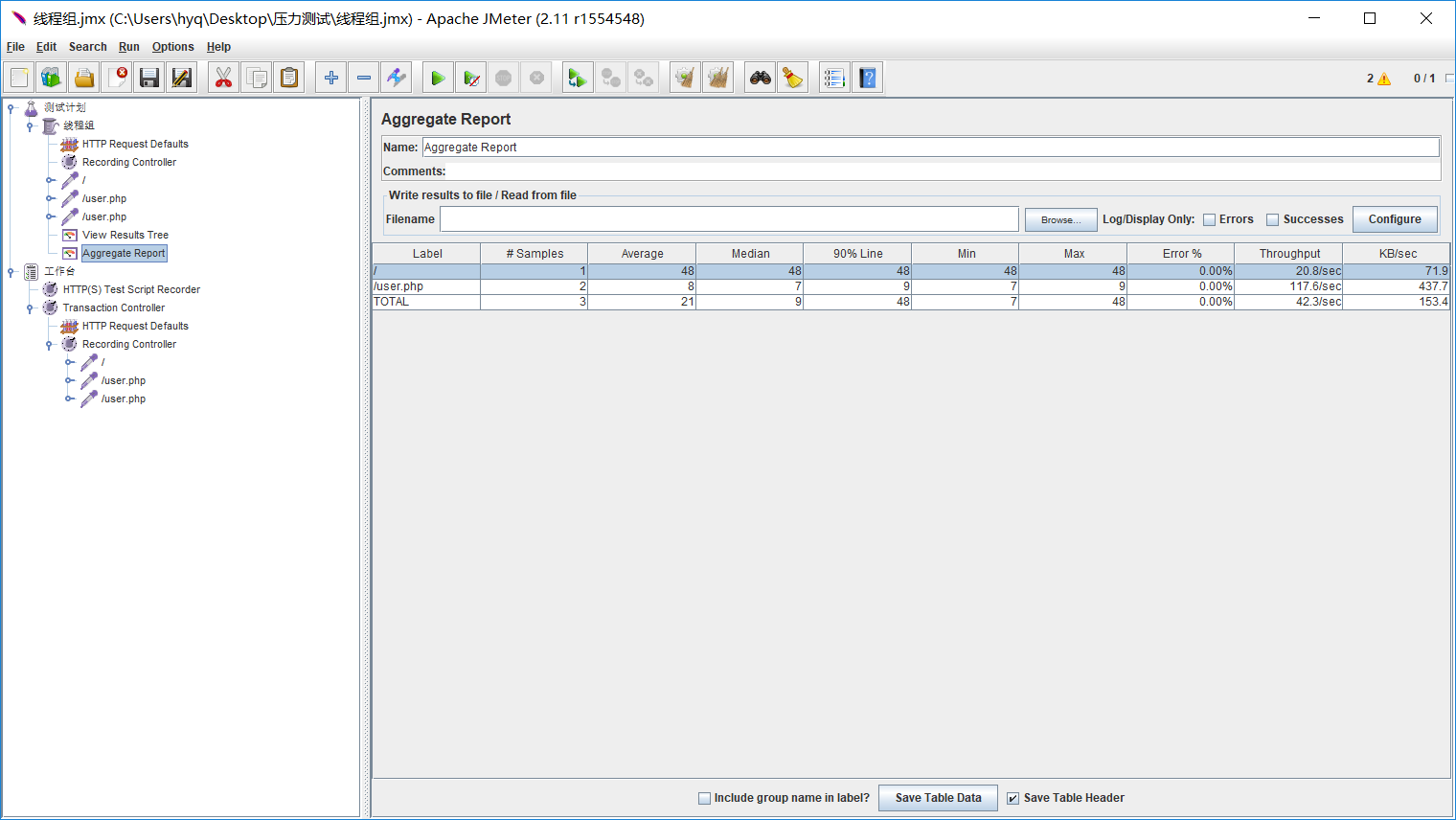
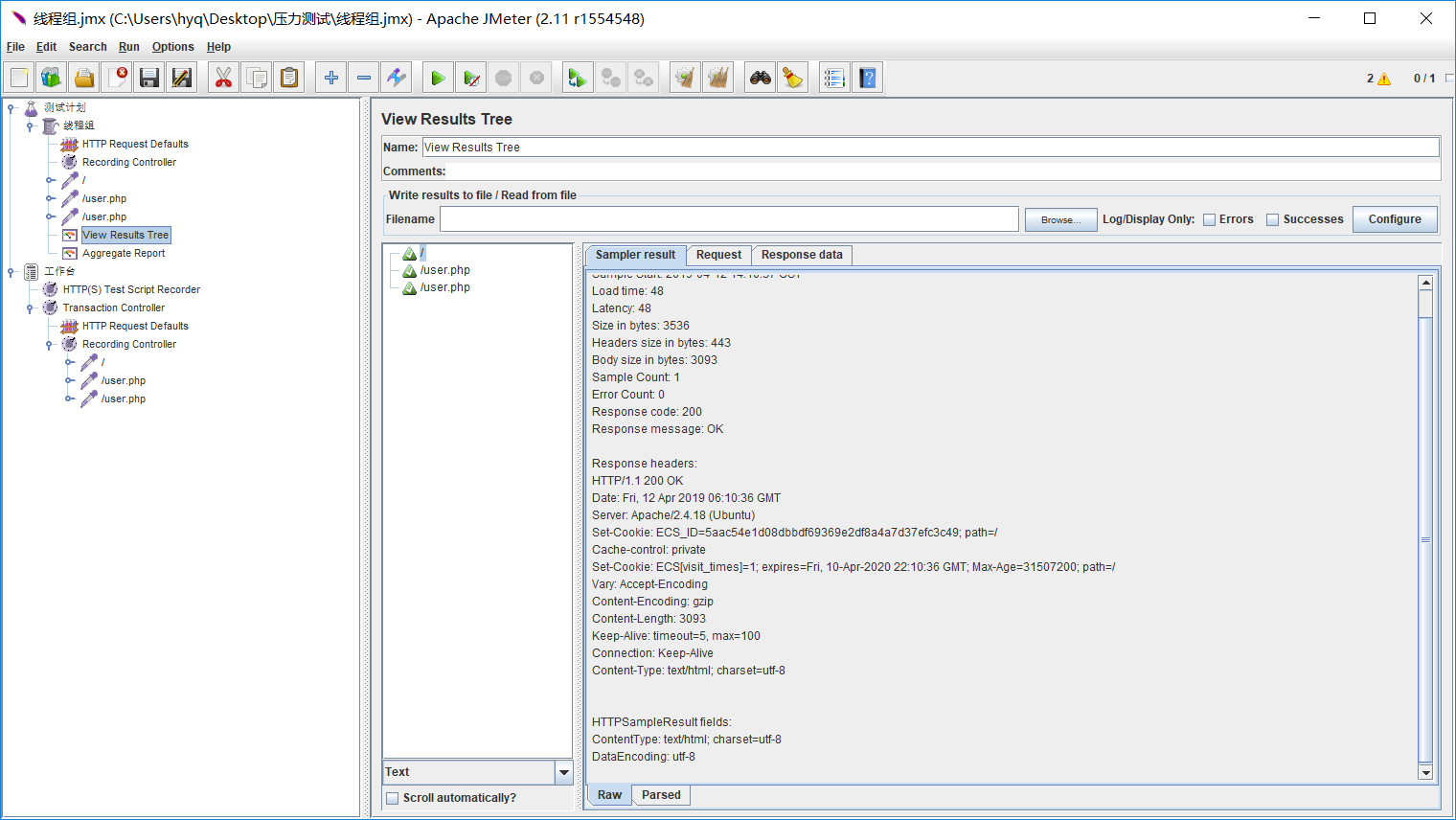




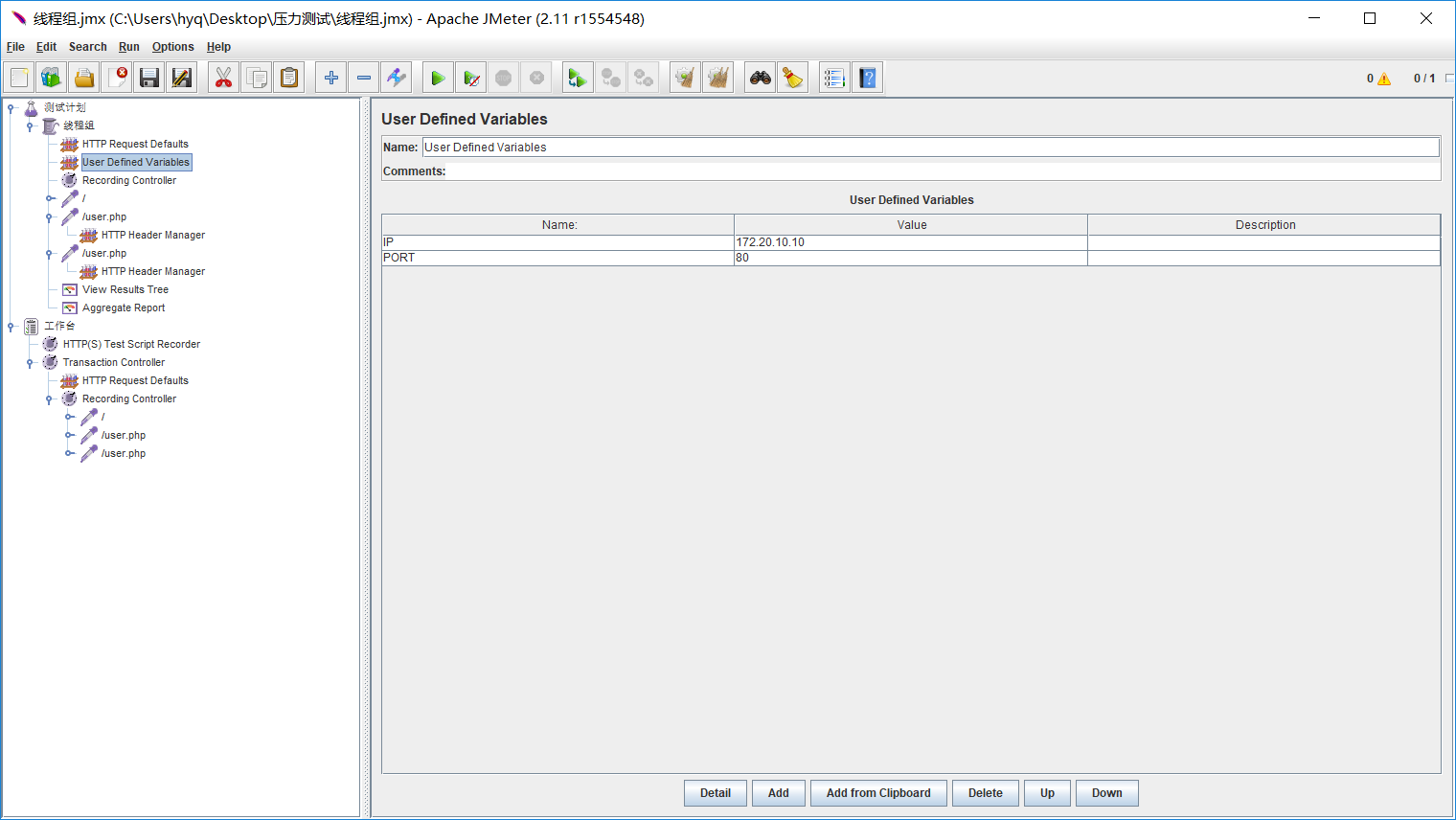


1. 再进行回放，添加两个listener控件来对访问结果进行监控

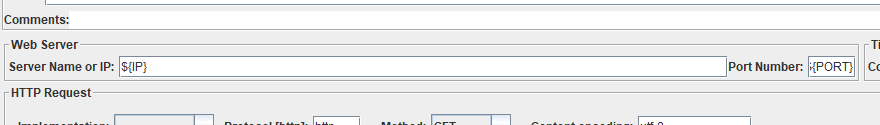




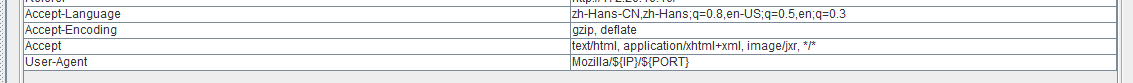
1. 将测试脚本参数化



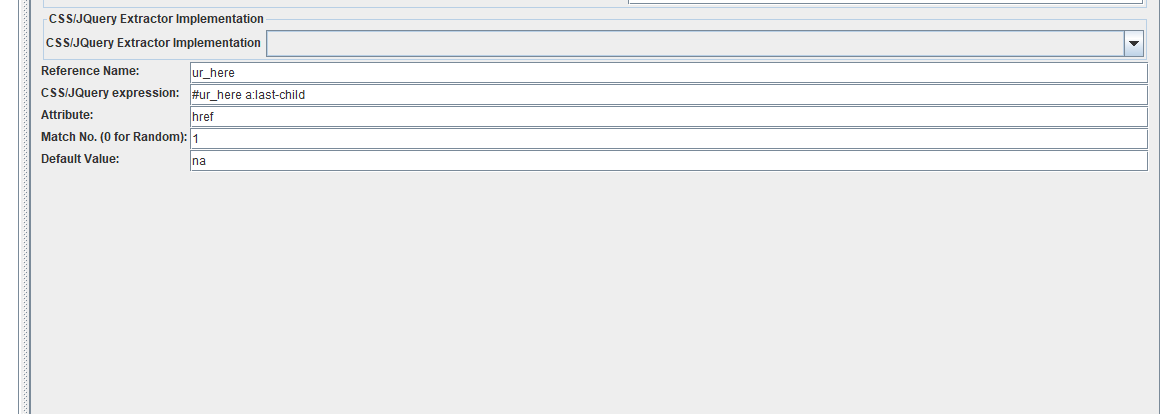
将IP和端口改为变量名${IP},${PORT}.



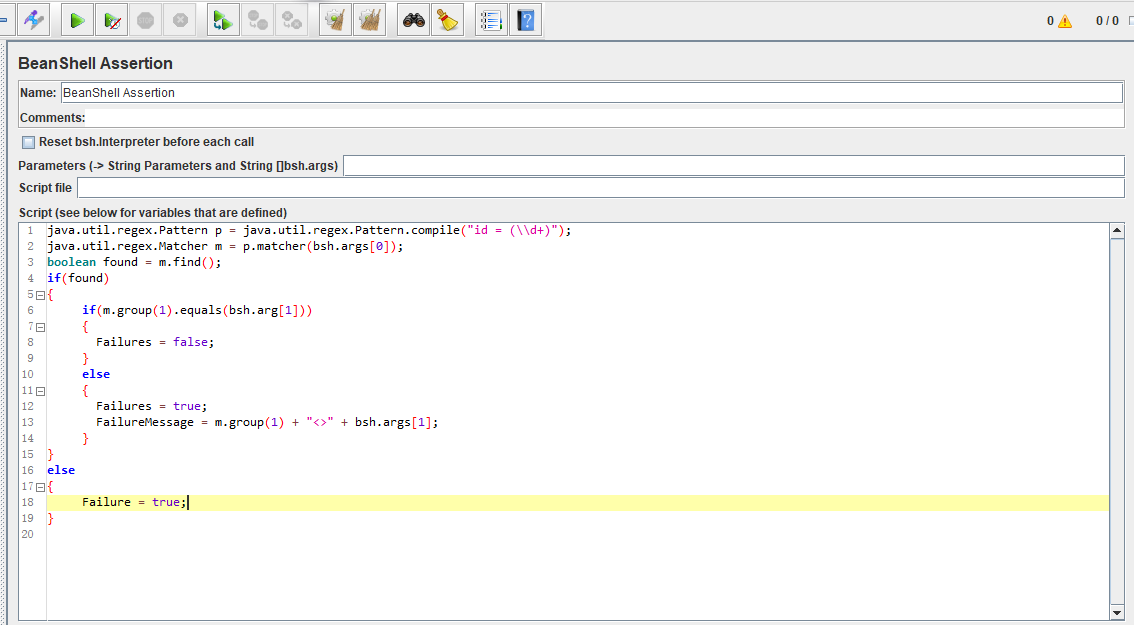
将相应文件中的User-Agernt后面的IP和端口进行更改。



e)使用CSS/JQuery Extractor 获取相应数据

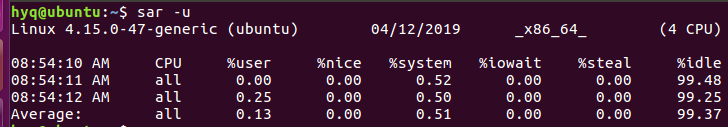


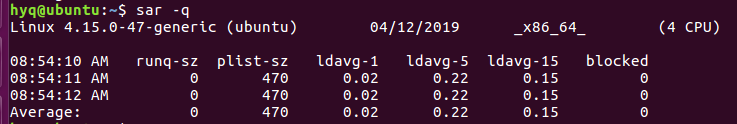
f)使用BeanShell Assertion来判断响应是否正确。



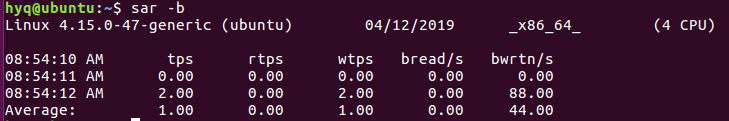
1. sysstat测试:

CPU测试：

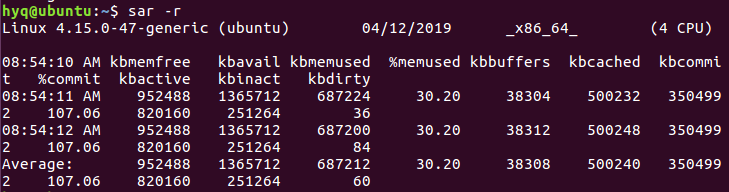




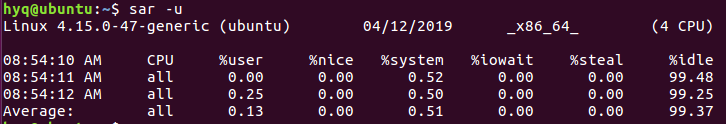
IO：



Memory:



Network:



1. 调试分析（在实验过程中遇到的问题以及如何解决）

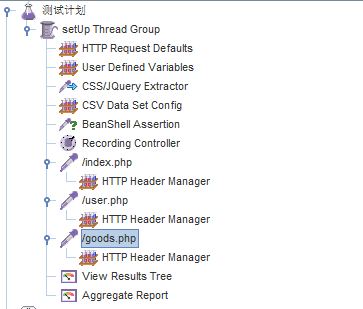
1.由于校园网没有进行IP隔离，所以没有给虚拟机分配IP地址，所以不能进 行连接，后面改用实验室的网

2.ECShop开始搭建环境时，由于各版本不一致，PHP版本过低导致运行失败，后面进行卸载重新安装合适版本

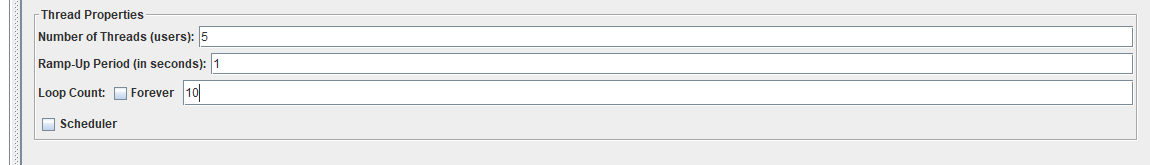
3.打开ECShop网页后，页面报错，原因同样是PHP版本过低，但是只需修改对应文件的内容即可

4.利用sysstat测试运行Jmeter测试之后的服务器性能时，没能得到正确结果，原因是二者测试进行未能同步

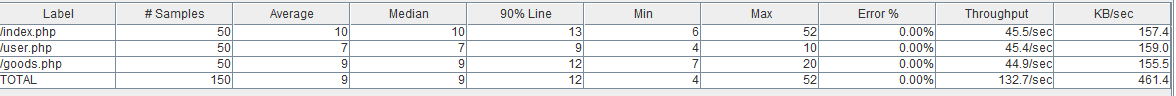
1. 测试结果（描述输入和输出）
2. Jmeter的Testplan展开图：



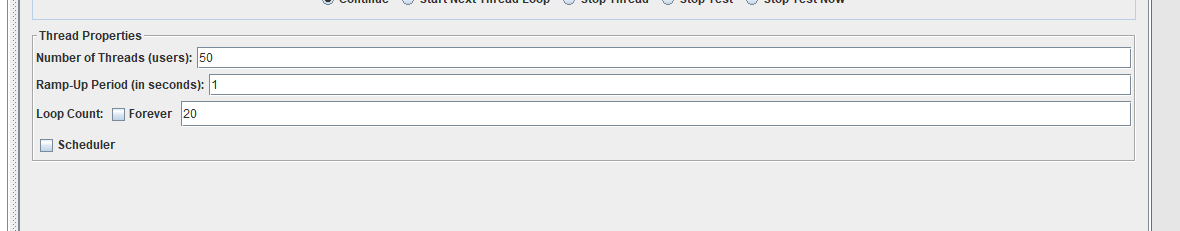
1. Jmeter 5\*10测试结果：



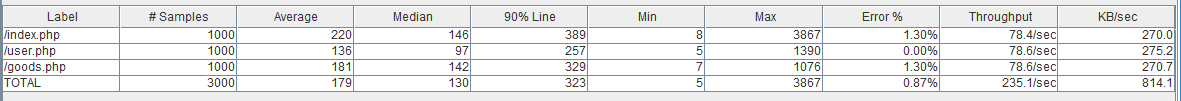
Aggregate Report结果：



1. Jmeter50\*20测试结果：



Aggregate Report结果



**Jmeter聚合报告参数详解：**

Label - 如图，在不勾选"Include group name in label?"复选框的情况下，为请求取样器的名称，否则为“请求取样器所在线程组:请求取样器名称”

# Samples - 用同一个请求取样器，发送请求的数量(注意：该值是不断累计的)。比如，10个线程数设置为10，迭代10次，那么每运行一次测试，该值就增加10\*10=100

Average - 默认情况下是单个Request的平均响应时间，当使用了Transaction Controller 时，也可以以Transaction为单位显示平均响应时间

Median - 中位数。表示响应时间本不大于该时间值的请求样本数占总数的50%

90% Line - 表示响应时间不大于该时间值的请求样本数占总数的90%

Min - 针对同一请求取样器，请求样本的最小响应时间

Max - 针对同一请求取样器，请求样本的最大响应时间

Error % - 出现错误的请求样本的百分比

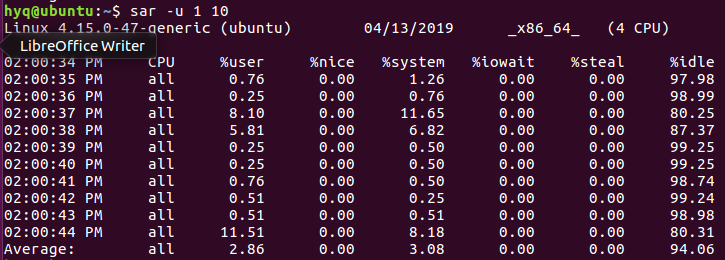
Throughput - 吞吐量以“requests/second、requests /minute、requests /hour”来衡量。

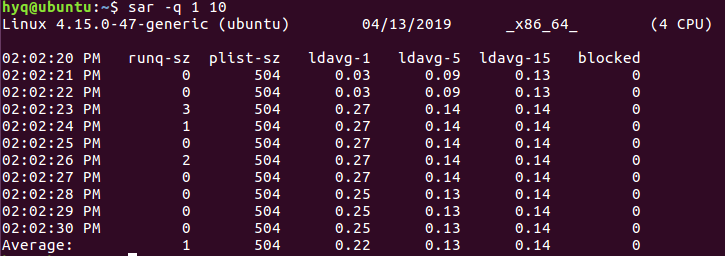
Kb/sec - 以Kilobytes/seond来衡量的吞吐量

1. sysstat测试

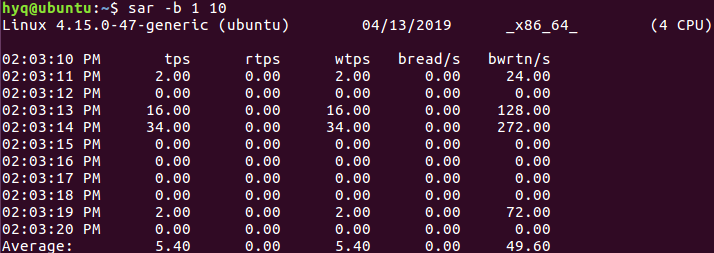
5\*10测试：

**CPU测试：**

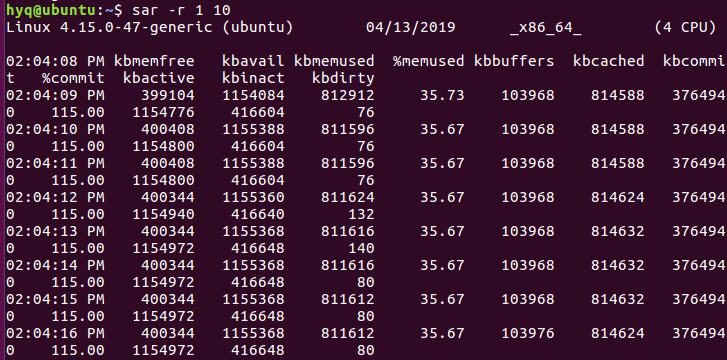




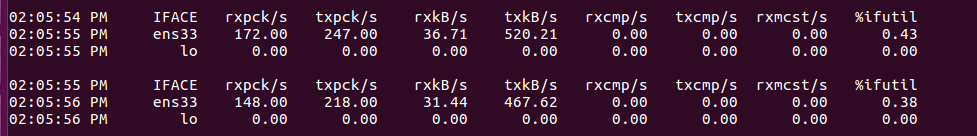
**IO**



**Memory**

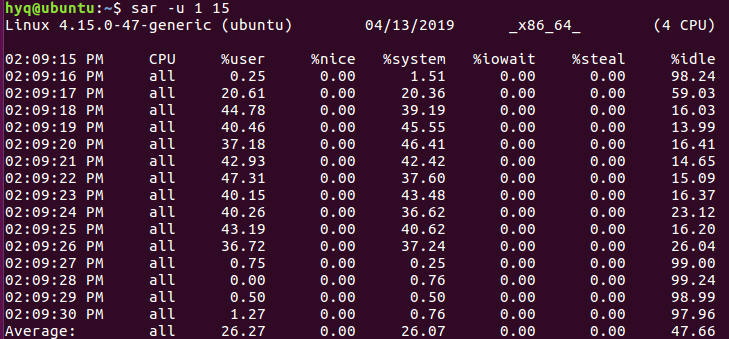


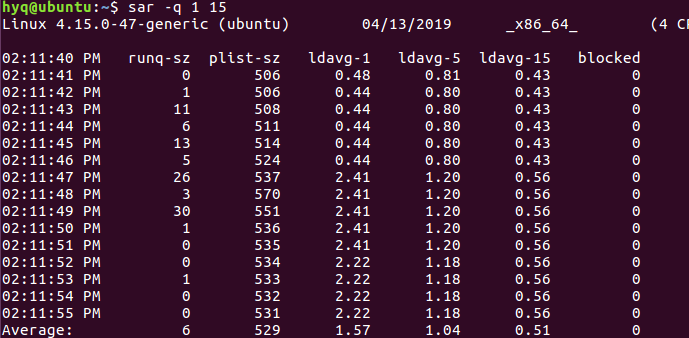
**Network**



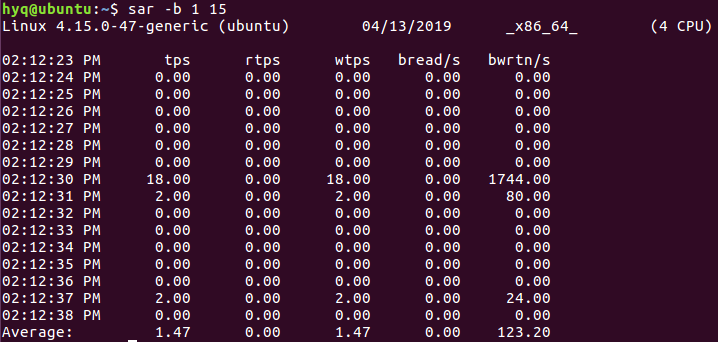
50\*20测试

**CPU：**

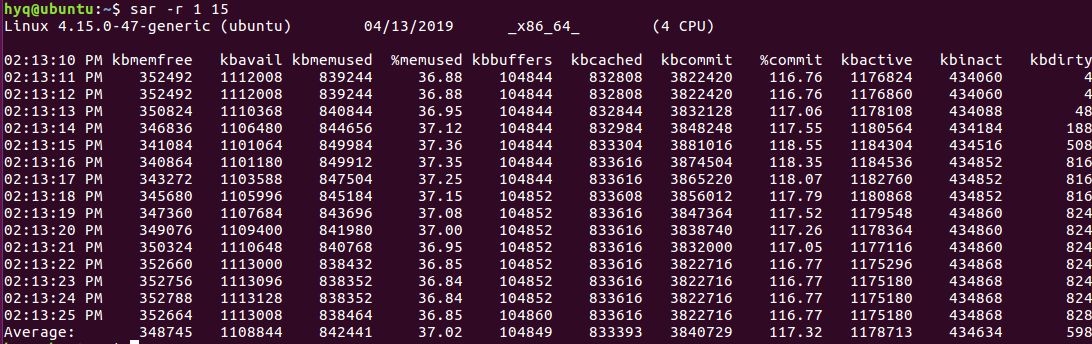




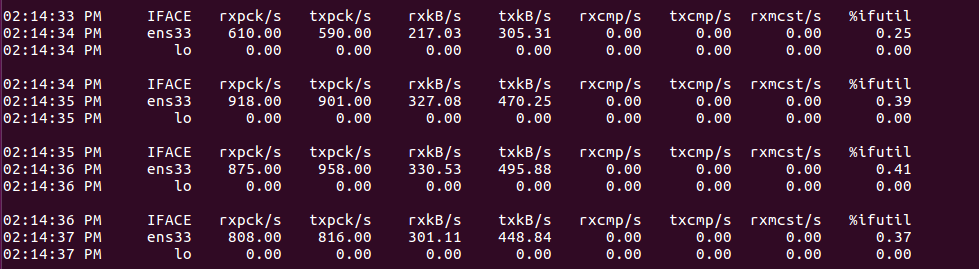
**IO**



**Memory**



**Network**



1. 总结

通过这次实验，我认识到搭建环境之前要确定版本的序号，防止版本不统一而导致环境搭建失败，环境搭建好后，通过动手实践，我基本了解啦如何用jmeter对系统进行压力测试并生成Jmeter测试报告，并通过sysstat得出Linux服务器的性能