

## 压力测试

9种压力测试工具

### jmeter

#### 了解压力测试

官网

jmeter window安装

jmeter目录说明

设置中文字体

```
language=zh_CN  
  
csvdataset.file.encoding_list=UTF-8|UTF-16|ISO-8859-15|US-ASCII  
  
sampleresult.default.encoding=UTF-8
```

了解压力测试

## 压力测试实践

测试-批量-样例

压力测试样例

## 分隔符问题

如果是，分割； 如果是|分割

数据-> 分列->分割符号->其他| ->完成

## linux下

历史版本

```
-- http://archive.apache.org/dist/jmeter/binaries/  
  
-- 下载 apache-jmeter-5.4.1.tgz  
  
-- 192.168.73.128
```

```

cd /usr/local/java/jmeter
-- 上传文件
rz -be xx.
-- 解压
tar -xzvf apache-jmeter-5.4.1.tgz
-- 修改环境变量
vi /etc/profile

export JMETER_HOME=/usr/local/java/jmeter/apache-jmeter-
5.4.1
export
CLASSPATH=$JMETER_HOME/lib/ext/ApacheJMeter_core.jar:$JMET
ER_HOME/lib/jorphan.jar:$CLASSPATH
export PATH=$JMETER_HOME/bin:$PATH:$HOME/bin

-- 刷新环境变量
source /etc/profile

-- 查看是否安装完成
jmeter -v

-- 使用
1) 将jmx脚本使用xftp上传至服务器，进入到jmeter下的bin目录，输入命
令： ./jmeter.sh -n -t /路径/test.jmx -l /路径/test.jtl ， 运
行jmeter脚本。

cd /usr/local/java/jmeter/apache-jmeter-5.4.1/bin

./jmeter.sh -n -t /jmeterResult/jmxShell/linux-test-
api.jmx -l /jmeterResult/jmxResult/result.jtl1 -e -o
/jmeterResult/jmxResult/temp1

--
cd /jmeterResult/jmxResult

-- 说明
/jmeterResult/practice.jmx 这是脚本的目录
/jmeterResult/result.jtl 这是存放结果的目录及文件名称
-e -o /jmeterResult/temp 这是生成HTML报告以及报告存放路径

jmeter -n -t 南网提单接口压力测试.jmx -l F:\jmeter\result.txt
-e -o F:\jmeter\webreport1

```

## 场景

压力测试分两种场景：一种是单场景，压一个接口的；第二种是混合场景，多个有关联的接口。压测时间，一般场景都运行10-15分钟。如果是疲劳测试，可以压一天或一周，根据实际情况来定。

## 压测任务需求的确认

压测前要明确压测功能和压测指标，一般需要确定的几个问题：

固定接口参数进行压测还是进行接口参数随机化压测？ 要求支持多少并发数？  
TPS（每秒钟处理事务数）目标多少？响应时间要达到多少？ 压服务器名称还是压服务器IP，一般都是压测指定的服务器

## 压测设置

线程数：并发数量，能跑多少量。具体说是一次存在多少用户同时访问 **Rame-Up Period(in seconds)**:表示JMeter每隔多少秒发动并发。理解成准备时长：设置虚拟用户数需要多长时间全部启动。如果线程数是20，准备时长为10，那么需要10秒钟启动20个数量，也就是每秒钟启动2个线程。循环次数：这个设置不会改变并发数，可以延长并发时间。总请求数=线程数\*循环次数 调度器：设置压测的启动时间、结束时间、持续时间和启动延迟时间。

## 压测结果查看

运行完后，聚合报告会显示压测的结果。主要观察**Samples**、**Average**、**error**、**Throughput**。

**Samples**:表示一共发出的请求数 **Average**: 平均响应时间，默认情况下是单个**Request**的平均响应时间（ms） **Error%**:测试出现的错误请求数量百分比。若出现错误就要看服务端的日志，配合开发查找定位原因 **Throughput**:简称**tps**,吞吐量，默认情况下表示每秒处理的请求数，也就是指服务器处理能力，**tps**越高说明服务器处理能力越好。

## 压测结果的分析

有错误率同开发确认，确定是否允许错误的发生或者错误率允许在多大的范围内；

**Throughput**吞吐量每秒请求的数大于并发数，则可以慢慢的往上面增加；若在压测的机器性能很好的情况下，出现吞吐量小于并发数，说明并发数不能再增加了，可以慢慢的往下减，找到最佳的并发数；

压测结束，·登陆相应的**web**服务器查看**CPU**等性能指标，进行数据的分析；

最大的**tps**:不断的增加并发数，加到**tps**达到一定值开始出现下降，那么那个值就是最大的**tps**。

最大的并发数：最大的并发数和最大的**tps**是不同的概率，一般不断增加并发数，达到一个值后，服务器出现请求超时，则可认为该值为最大的并发数。压测过程出现性能瓶颈，若压力机任务管理器查看到的**cpu**、网络和**cpu**都正常，未达到90%以上，则可以说明服务器有问题，压力机没有问题。影响性能考虑点包括：数据库、应用程序、中间件（**tomact**、**Nginx**）、网络和操作系统等方面。

## jmeter在linux下进行压力测试

jmeter 在linux安装 简单说下，就是要先安装jdk,同时再配置环境变量，最后再上传jmeter压缩的安装包，在linux下解压完安装包就可以使用了。推荐博客：<http://blog.csdn.net/zhemeteor/article/details/51315874>

jmeter在linux运行 进入jmeter下的bin目录下运行脚本，未配置jmeter环境变量的条件下，运行的命令：

```
./jmeter -n -t a.jmx -l res.jtl
```

其中a.jmx是准备好的jmeter脚本,res.jtl是测试结果文件，测试结果文件可以导入到jmeter察看结果树下查看

## 了解压力测试

1.什么是压力测试 压力测试考察当前软硬件环境下系统所能承受的最大负荷并帮助找出系统瓶颈所在。压测都是为了系统在线上的处理能力和稳定性维持在一个标准范围内，做到心中有数。使用压力测试，我们有希望找到很多种用其他测试方法更难发现的错误。有两种错误类型是：内存泄漏，并发与同步。有效的压力测试系统将应用以下这些关键条件:重复，并发，量级，随机变化。

2.性能指标 响应时间（Response Time: RT） 响应时间指用户从客户端发起一个请求开始，到客户端接收到从服务器端返回的响 应结束，整个过程所耗费的时间。 HPS（Hits Per Second）：每秒点击次数，单位是次/秒。

TPS（Transaction per Second）：系统每秒处理交易数，单位是笔/秒。

QPS（Query per Second）：系统每秒处理查询次数，单位是次/秒。对于互联网业务中，如果某些业务有且仅有一个请求连接，那么TPS=QPS=HPS，一般情况下用TPS 来衡量整个业务流程，用QPS 来衡量接口查询次数，用HPS 来表示对服务器单击请求。无论TPS、QPS、HPS,此指标是衡量系统处理能力非常重要的指标，越大越好，根据经验，一般情况下：金融行业：

1000TPS~50000TPS，不包括互联网化的活动 保险行业：100TPS~100000TPS，

不包括互联网化的活动 制造行业：10TPS~5000TPS 互联网电子商务：

10000TPS~1000000TPS 互联网中型网站：1000TPS~50000TPS 互联网小型网

站：500TPS~10000TPS 最大响应时间（Max Response Time）指用户发出请求

或者指令到系统做出反应（响应）的最大时间。最少响应时间（Mininum

ResponseTime）指用户发出请求或者指令到系统做出反应（响 应）的最少时

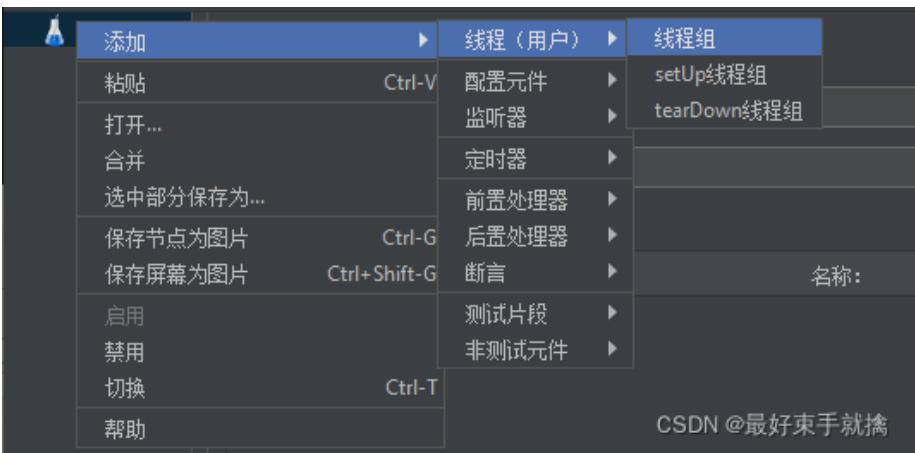
间。90%响应时间（90% Response Time）是指所有用户的响应时间进行排

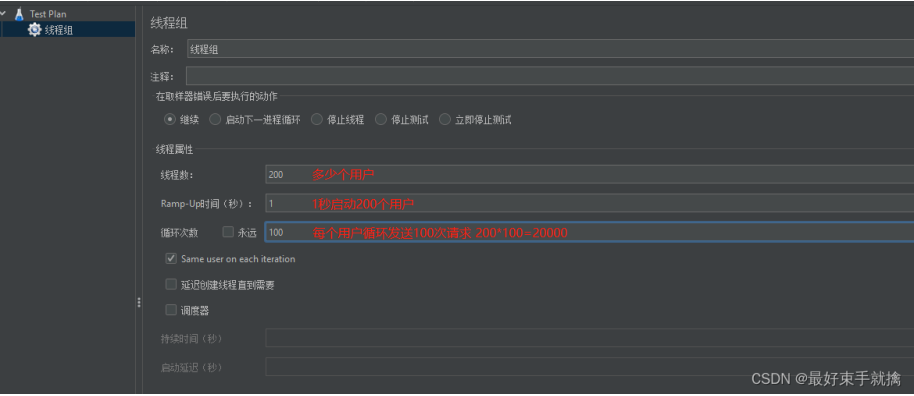
序，第90%的响 应时间。从外部看，性能测试主要关注如下三个指标 吞吐量：

每秒钟系统能够处理的请求数、任务数。响应时间：服务处理一个请求或一个

任务的耗时。错误率：一批请求中结果出错的请求所占比例。 3.JMeter 下载

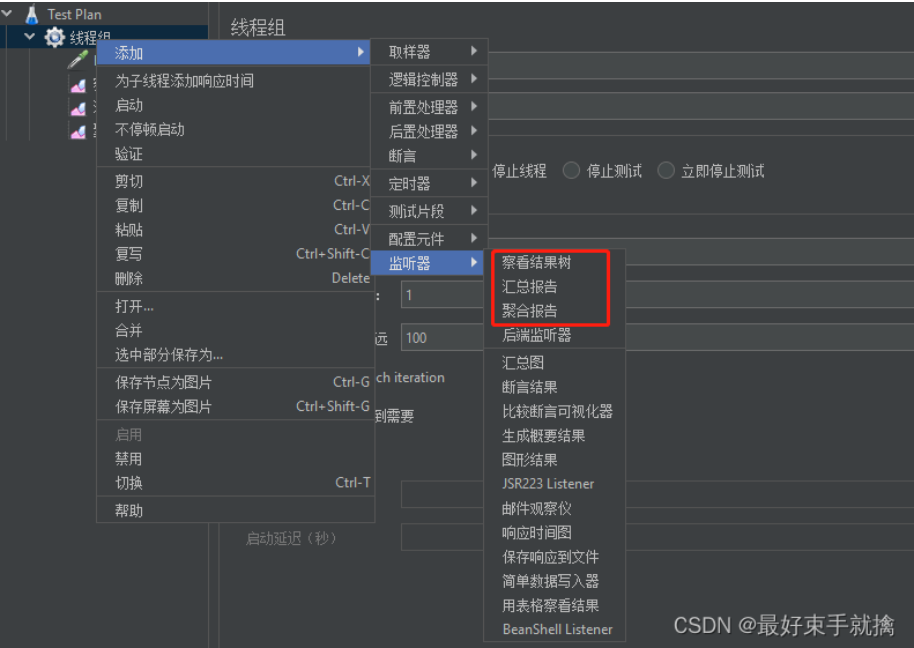
### 1.为测试计划添加线程组



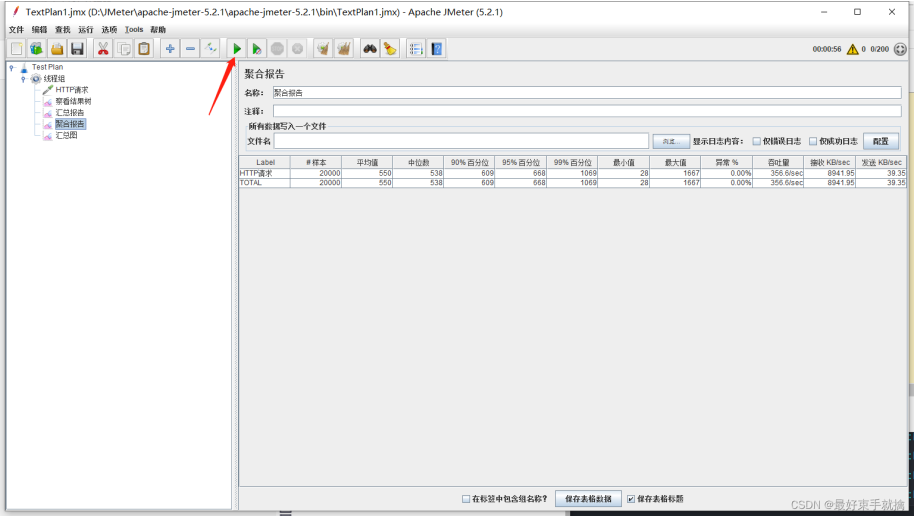


2.为线程组添加取样器 取样器可以测试多种请求

3.添加常用监听器



4.开始测试



优化：提高内存-Xmx512m

5.JMeter Address Already in use 错误解决

windows 本身提供的端口访问机制的问题。Windows 提供给TCP/IP 链接的端口为1024-5000，并且要四分钟来循环回收他们。就导致我们在短时间内跑大量的请求时将端口占满了。1.cmd 中，用regedit 命令打开注册表 2.在HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters 下， 1.右击parameters，添加一个新的DWORD，名字为MaxUserPort 2.然后双击MaxUserPort，输入数值数据为65534，基数选择十进制（如果是分布式运行的话，控制机器和负载机器都需要这样操作哦）

3. 修改配置完毕之后记得重启机器才会生效 <https://support.microsoft.com/zh-cn/help/196271/when-you-try-to-connect-from-tcp-ports-greater-than-5000-you-receive-tcp-timedwait-delay-30>

3.性能优化 Throughput 吞吐量每秒请求的数大于并发数，则可以慢慢的往上面增加；若在压测的机器性能很好的情况下，出现吞吐量小于并发数，说明并发数不能再增加了，可以慢慢的往下减，找到最佳的并发数；

压测结束，登陆相应的web 服务器查看CPU 等性能指标，进行数据的分析；最大的tps，不断的增加并发数，加到tps 达到一定值开始出现下降，那么那个值就是最大的tps。最大的并发数：最大的并发数和最大的tps 是不同的概率，一般不断增加并发数，达到一个值后，服务器出现请求超时，则可认为该值为最大的并发数。压测过程出现性能瓶颈，若压力机任务管理器查看到的cpu、网络和网络和cpu 都正常，未达到90%以上，则可以说明服务器有问题，压力机没有问题。影响性能考虑点包括：数据库、应用程序、中间件（tomcat、Nginx）、网络和操作系统等方面