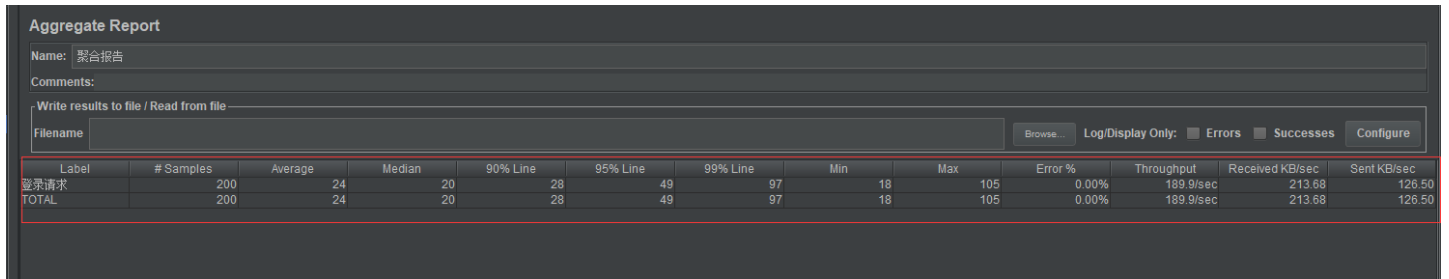


# JMeter笔记1：聚合报告参数分析



Aggregate Report

Name: 聚合报告

Comments:

Write results to file / Read from file

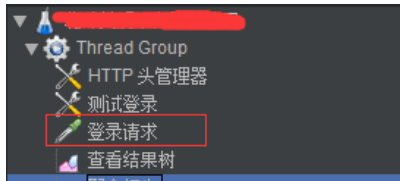
Filename:  Browse...

Log/Display Only: ☐ Errors ☒ Successes

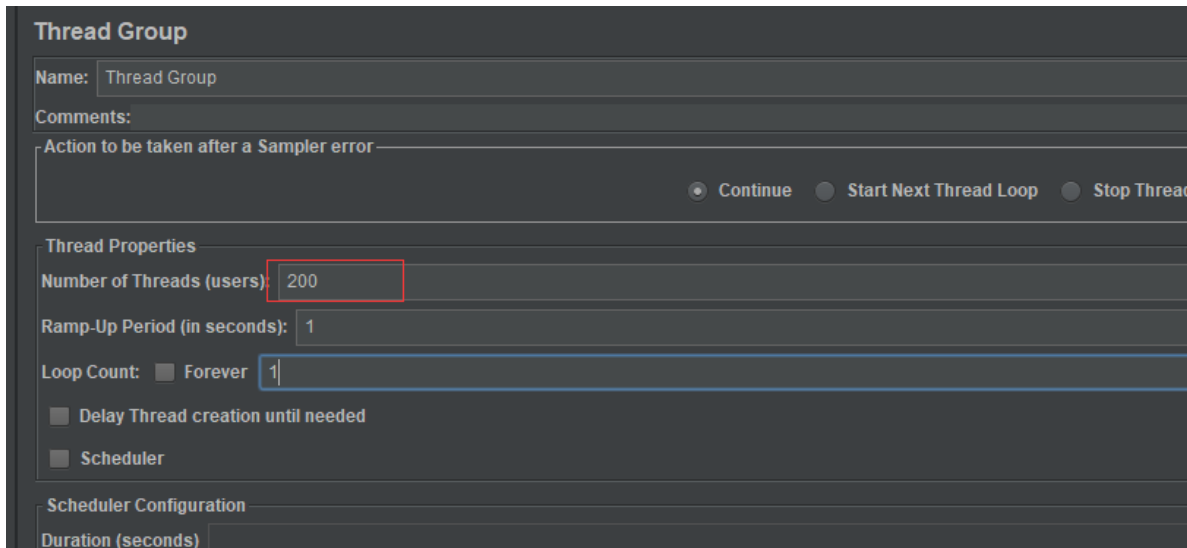
Label	# Samples	Average	Median	90% Line	95% Line	99% Line	Min	Max	Error %	Throughput	Received KB/sec	Sent KB/sec
登录请求	200	24	20	28	49	97	18	105	0.00%	189.9/sec	213.68	126.50
TOTAL	200	24	20	28	49	97	18	105	0.00%	189.9/sec	213.68	126.50

如下：

Label：每个请求的名称，比如HTTP请求等



#Samples：发给服务器的请求数量（如图是200个请求，若模拟100个用户，循环10次，请求数是1000）



Thread Group

Name: Thread Group

Comments:

Action to be taken after a Sampler error

☒ Continue ☐ Start Next Thread Loop ☐ Stop Thread

Thread Properties

Number of Threads (users): 200

Ramp-Up Period (in seconds): 1

Loop Count: ☐ Forever

☐ Delay Thread creation until needed

☐ Scheduler

Scheduler Configuration

Duration (seconds):

Average：单个请求的平均响应时间。默认是单个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，也就是50%用户的响应时间

90%Line：90%用户的响应时间

95%Line：95%用户的响应时间

99%Line：99%用户的响应时间

Min：最小的响应时间

Max：最大的响应时间

注：为什么要有\*%用户响应时间？因为在评估一次测试的结果时，仅仅有平均事物响应时间是不够的。假如有一次测试，总共有100个请求被响应，其中最小响应时间为0.02秒，最大响应时间为110秒，平均事务响应时间为4.7秒，你会不会想到最小和最大响应时间如此大的偏差是否会导致平均值本身并不可信？

我们可以在95 th之后继续添加96/ 97/ 98/ 99/ 99.9/ 99.99 th，并利用Excel的图表功能画一条曲线，来更加清晰表现出系统响应时间的分布情况。这时候你也许会发现，那个最大值的出现几率只不过是千分之一甚至万分之一，而且99%的用户请求的响应时间都是在性能需求所定义的范围之内的；如下图则是最低响应时间的值出现几率是很小的，实际99%的用户请求响应时间都要20000+。

Error%: 错误率, 本次测试中出现错误的请求的数量/请求的总数

Throughput: 吞吐量。默认情况下表示每秒完成的请求数, 吞吐量=请求数/总时间

Received KB/sec: 每秒从服务器端接收到的数据量, 即: 收到的千字节每秒的吞吐量测试

Sent KB/sec: 每秒从客户端发送的请求的数量, 即: 发送的千字节每秒的吞吐量测试

聚合报告是累加的, 即每次运行的结果统计都是基于前一次运行的结果进行统计, 包括发起的请求样本数等都是叠加的, 比如我11:00运行一次, 发起10个请求, 11:20运行一次, 发起10个请求, 这时聚合报告显示请求数为20个, 而此时的吞吐量和第一次运行相差甚远, 它把11: 00到11: 20期间非运行状态的时间也算进去了。所以, 总时间增大, 吞吐量变小。