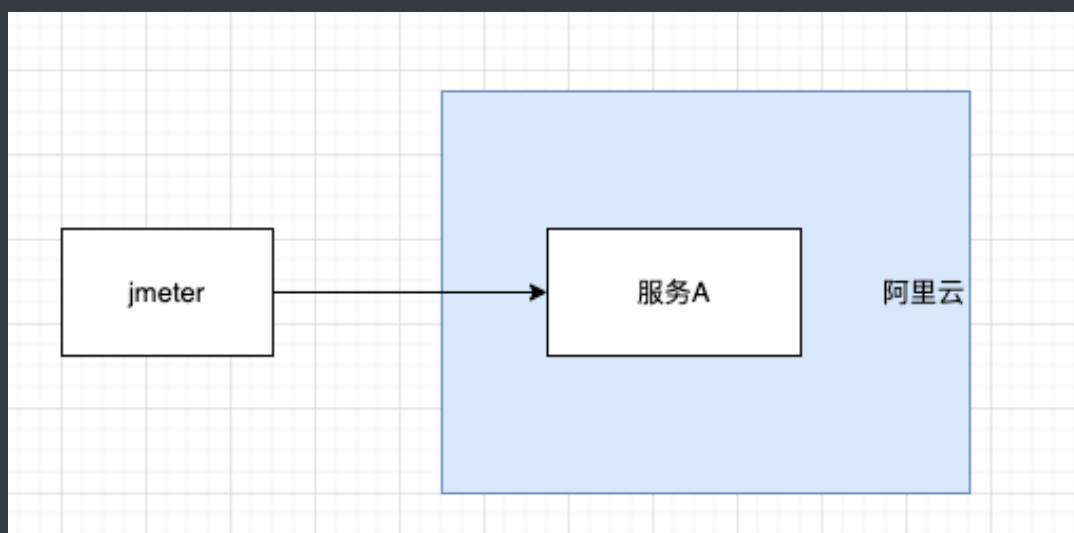


week01作业-项目性能测试报告

01-测试目的

- 搭建测试平台，学会使用jmeter测试工具进行压测。
- 尝试分析出某个具体场景下系统的瓶颈和问题所在。

02-测试工具



03-测试环境

3.1 环境

指标	参数
机器	4C8G
集群规模	单机
网络带宽	1Mbps
数据库	2C2G
Hero_web版本	1.0

3.2 设置启动参数

```
1  nohup java -server -Xms512m -Xmx512m -Xmn256 -XX:MetaspaceSize=128m -  
   XX:MaxMetaspaceSize=320m -jar hero_web-1.0-SNAPSHOT-default.jar >>  
   ./hero_test.out 2>&1 &
```

04-测试场景

验证hero_web服务获取商品信息接口在不同并发规模的表现。

情况1： 用户访问接口并发逐渐增加的过程。接口的响应时间为60ms，线程梯度：5、10、15、20、25、30、35、40个线程，200次；

- 时间设置： Ramp-up period(秒)的值设为1s。
- 测试时长： 约等于60ms x 200次 x 8 = 96s

05-核心接口测试结果

一、商品详情页涉及到的接口

1、获取商品信息接口

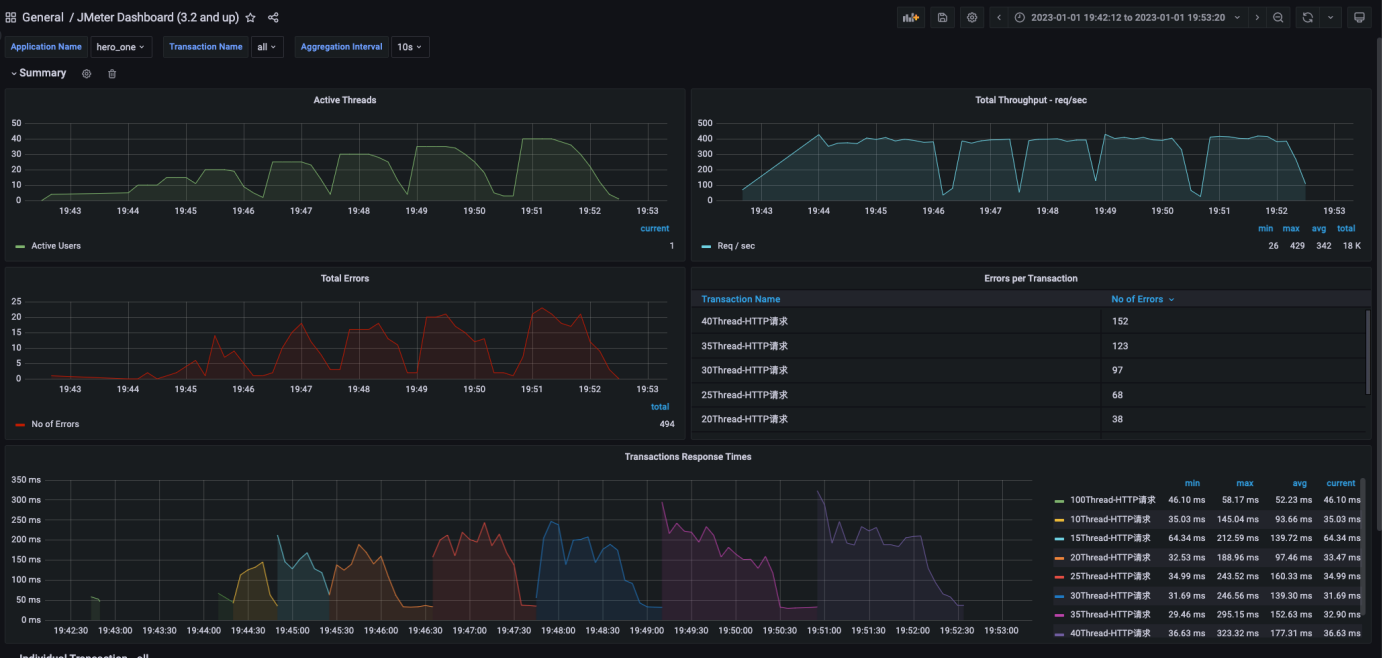
验证hero_mall服务获取商品信息接口能力性能。目标峰值TPS：10000，P99响应时间：300ms

Active Threads：注意压力机的活动线程数并不等同于在线用数中，并发的用户数。压力活动线程数是用来模拟用户发出并发请求的操作。

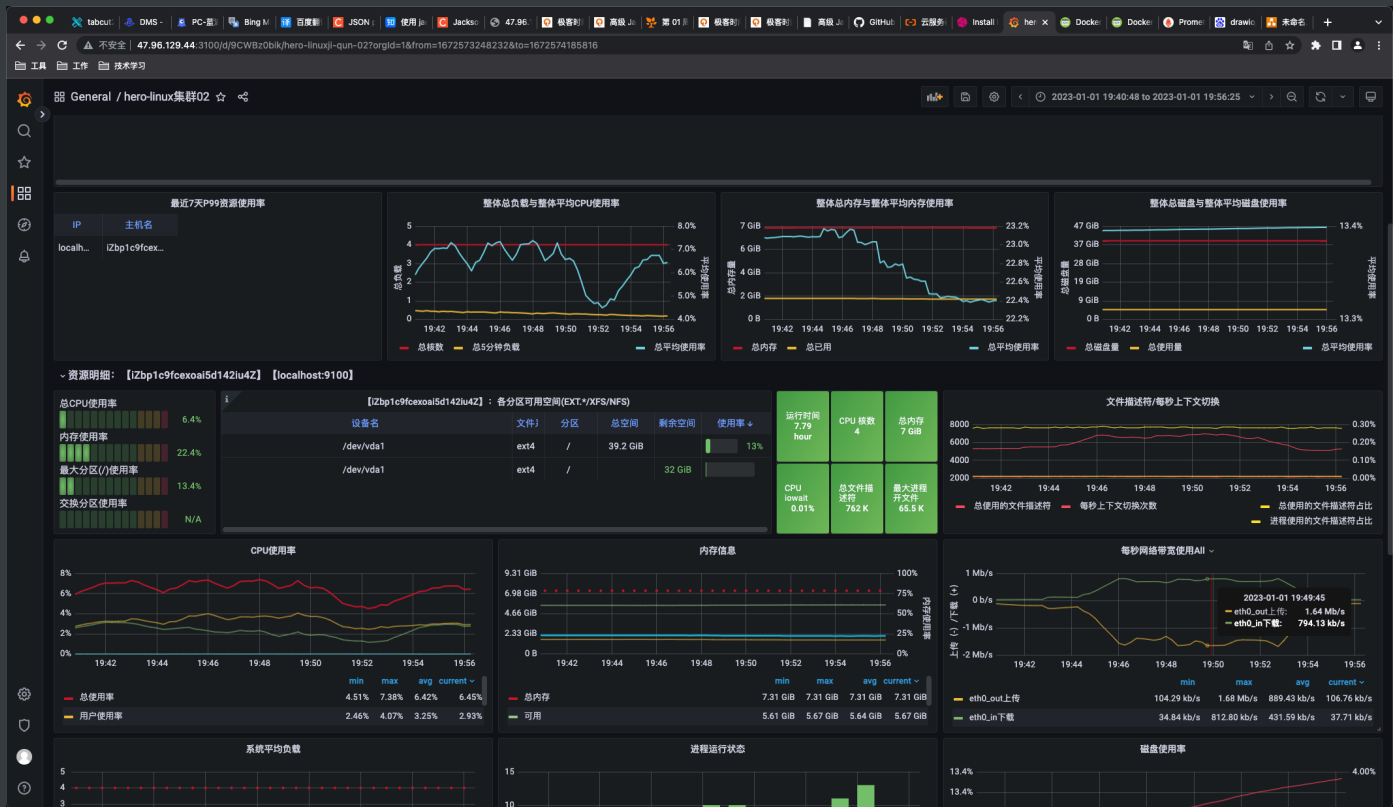
- 压力机-活动线程数 =模拟=> 系统并发用户数
- 系统并发用户数 =不等于=> 压力机-活动线程数

TPS、RT

Label ↑	# 样本	平均值	中位数	90% 百分位	95% 百分位	99% 百分位	最小值	最大值	异常 %	吞吐量	接收 KB/sec	发送 KB/sec
10Thread-H...	2000	119	35	328	362	839	27	7324	0.15%	71.8/sec	76.20	0.00
15Thread-H...	3000	164	36	350	792	1776	27	7534	0.40%	79.8/sec	84.67	0.00
20Thread-H...	4000	218	35	349	775	1993	27	58654	0.95%	57.9/sec	61.45	0.00
25Thread-H...	5000	272	38	381	826	3632	27	31618	1.36%	70.6/sec	74.93	0.00
30Thread-H...	6000	308	37	372	828	3741	26	59721	1.62%	73.2/sec	77.73	0.00
35Thread-H...	7000	350	37	381	828	3746	26	60872	1.76%	66.0/sec	70.01	0.00
40Thread-H...	8000	376	38	561	855	3774	27	64663	1.90%	78.5/sec	83.31	0.00
5Thread-HT...	1000	56	33	117	198	352	27	818	0.00%	73.2/sec	77.70	0.00
总体	36000	287	36	367	820	3647	26	64663	1.37%	70.7/sec	75.06	0.00



Linux



06-测试结论

针对单机的hero_web系统进行性能测试，综合jmeter聚合报告和grafana结果可以得出以下结论：

- 在当前的1Mbps、4C8GECS的系统下，商品信息的接口TPS为在线程数15个时达到峰值79.8/sec，此时系统的95%请求的RT都能在1s以内，但是99%RT达到1.7秒，不太理想。
- 继续增加系统压力，错误数逐渐上升，从jmeter的结果树中看出是超出了3s的响应时间所致。
- 单个接口本身的响应时间实际上是很短的，但是压力上来会导致大量的请求数堆积，大量请求的等待时间变长，调整系统的最大线程数应该可以解决掉一部分的压力。
- 根据Linux系统报告可以看出，系统压力接近网络带宽上限，但是内存和CPU使用率较低。
- 根据以上结论，可以从以下几点提升系统瓶颈：
 - 增大系统的JVM内存，进而可以调大最大线程数，来达到提升TPS的效果。

- 上面的参数翻倍，可能会先达到Linux的网络带宽上限，进而可以考虑提升网络带宽进一步提高TPS。