内部资料

注意保密

**收银项目性能测试报告**

**版 本**

**文档编号：LZ-XNCS-BG-0001**

**版本说明：**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本号** | **版本日期** | **作者** | **备注** |
| V1.0 | 2020-10-13 | 王军 | 原始创建 |
|  |  |  |  |
|  |  |  |  |

**目录**

[1引言 2](#_Toc29046)

[1.1测试背景 2](#_Toc5465)

[1.2测试目的 2](#_Toc9130)

[1.3参考资料 2](#_Toc28027)

[2组织架构 2](#_Toc18858)

[3测试范围 3](#_Toc108)

[3.1业务模型 3](#_Toc20904)

[3.2性能需求指标 3](#_Toc14108)

[4测试实施 4](#_Toc3639)

[4.1测试环境 4](#_Toc31151)

[4.1.1测试环境网络拓扑 4](#_Toc3934)

[4.1.2测试环境配置 4](#_Toc1720)

[4.2测试工具 4](#_Toc30055)

[4.3测试挡板 5](#_Toc29014)

[4.4测试数据 5](#_Toc4868)

[4.5测试脚本 5](#_Toc25916)

[5测试策略 5](#_Toc8637)

[5.1测试执行策略 5](#_Toc2024)

[7.2测试监控策略 6](#_Toc8912)

[6测试结果分析 6](#_Toc18575)

[6.1性能概要 6](#_Toc6106)

[6.2单场景负载测试性能表现 7](#_Toc15509)

[6.2.1系统处理能力 7](#_Toc23045)

[6.2.2响应时间 7](#_Toc21632)

[6.2.3系统资源 8](#_Toc19357)

[7综合评估测试结论 8](#_Toc21961)

[8附件 9](#_Toc16172)

# 1引言

## 1.1测试背景

收银项目为新开发项目，特对该项目重要功能进行压测，保证项目高效、平稳运行。同时为系统优化，应用扩容提供数据支持

## 1.2测试目的

本次测试主要对收银项目的重要功能进行性能测试，找出高并发时的性能瓶颈，并对重构后的性能进行评估

## 1.3参考资料

无

# 2组织架构

|  |  |  |
| --- | --- | --- |
| **角色** | **职责** | **资源名** |
| 测试管理 | 1、负责测试方案、测试计划、测试报告制定  2、负责组织测试方案、测试报告的评审  3、负责测试进度跟进、协调测试工作开展  4、负责测试相关文档的归档处理 | 性能测试 |
| 测试实施 | 1、负责测试案例制定、测试场景设计  2、负责脚本编写、调试  3、负责测试数据准备  4、负责测试环境部署  5、负责测试任务执行、瓶颈分析定位、测试数据记录 | 性能测试  业务测试 |
| 应用及环境支持 | 1、负责待测系统的需求和相关应用信息确认  2、负责测试计划、测试方案、测试报告确认  3、负责测试过程中与系统相关问题的解决  4、配合测试工作的开展  5、负责测试环境部署联调 | 测试  产品  PM  研发  运维 |

# 3测试范围

## 3.1业务模型

开发提供的本次具体测试接口如下：

|  |  |  |
| --- | --- | --- |
| **序号** | **接口名称** | **接口URL** |
| 1 | 支付 | /api/v1/pos-manager-api/pos/pay |
| 2 | 支付订单查询 | /api/v1/pos-manager-api/pos/pay/query |

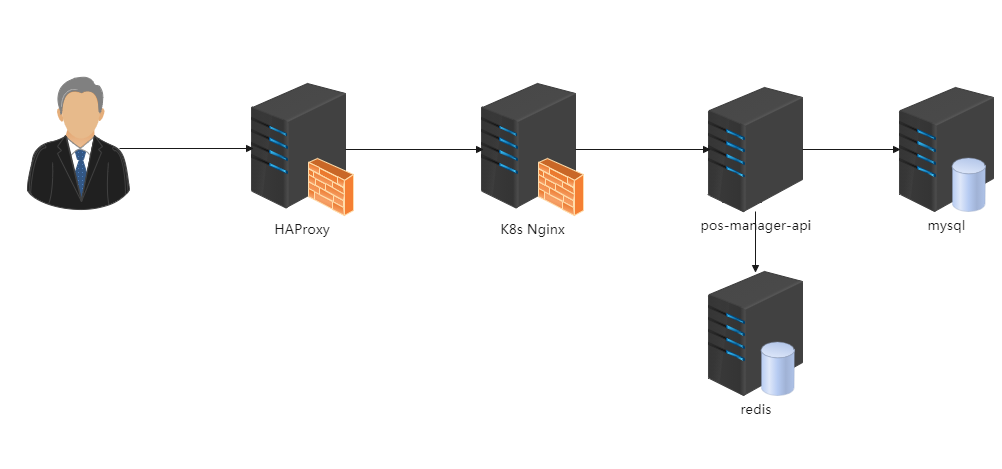
## 3.2性能需求指标

本次测试以性能评估为目的，不涉及具体指标

# 4测试实施

## 4.1测试环境

### 4.1.1测试环境网络拓扑



### 4.1.2测试环境配置

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **服务名称** | **数量** | **资源配置** | **软件版本** | **JVM主要参数** |
| pos-manager-api | 1 | 2C2G | dev | -Xmx1024m  -Xms1024m  -Xmn512m  -Xss512k  -XX:MetaspaceSize=128m  -XX:MaxMetaspaceSize=320m |
| mysql | 1 | 未知 | 未知 |  |
| redis | 1 | 未知 | 未知 |  |

## 4.2测试工具

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **硬件名称** | **数量** | **硬件配置** | **软件配置** | **备注** |
| 负载机 | 1 | 4C8G | Jmeter |  |
| 监控机 | 1 | 4C8G | Influxdb+Grafana |  |

## 4.3测试挡板

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **数量** | **功能** | **开发负责人** |
| 微信支付挡板 | 1 | 模拟微信支付功能，由开发直接在代码中mock，模拟微信支付耗时100ms | 邓名瑶 |

## 4.4测试数据

|  |  |  |  |
| --- | --- | --- | --- |
| **数据类型** | **数据类别** | **表名** | **数据量** |
| 基础数据 | 支付数据 | pay\_record\_offline | 30W |
| 测试数据 | 支付数据 | pay\_record\_offline | 3W |

## 4.5测试脚本



# 5测试策略

1、本次测试将在内网环境下通过Jmeter来模拟用户的使用行为，在测试过程中我们将尝试不同的测试场景，对系统的负载能力进行测试。测试开始前会准备足够多的铺底数据(铺底数据会直接影响系统性能)，测试过程中也会生成足够多的业务数据。同时也会监控系统硬件性能指标与中间件及数据库性能指标，确保能够全面的对系统性能进行评估。

2、本次测试会存在以下难点：

对于请求参数的加、解密和签名无法实现

1. 针对以上难点给出如下解决方法：

让开发设置取消加、解密和签名验证的开关

## 5.1测试执行策略

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **测试类型** | **运行时间** | **并发数** | **运行策略** | **目的** |
| 基准测试 | 5min | 1 | 单用户循环执行 | 检查业务本身是否存在性能缺陷。同时为后续的并发测试提供参考依据 |
| 并发测试 | 5min | 50 | 多用户并发 | 查找每个业务的性能瓶颈 |
| 单场景负载测试 | 10min | 200 | 200并发，分10步加载，每步20用户，每步加载后运行60秒，最后一步加载完成后运行60秒停止全部用户 | 评估每个业务在不同负载下的性能表现 |

## 7.2测试监控策略

系统指标监控

|  |  |  |  |
| --- | --- | --- | --- |
| **监控工具** | **指标类型** | **指标名称** | **指标描述** |
| Jmeter | 交易响应时间 | ResponseTimesOverTime | 客户端发起业务请求到得到响应的整个过程所经历的时间 |
| 并发用户数 | Concurrent users | 同时对应用系统发起业务请求的用户数量 |
| 系统处理能力 | TransactionsPerSecond | 每秒处理事务数（处理客户的请求数） |
| 请求错误率 | Error% | 指请求失败的数量占总请求量的百分比 |

主机资源监控

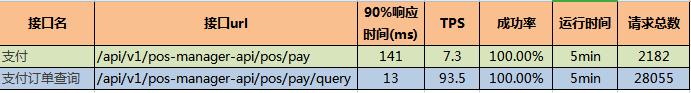
测试环境中应用服务操作系统均为Linux操作系统，采用grafana进行监控。监控指标如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **监控工具** | **指标类型** | **指标名称** | **指标描述** |
| grafana | CPU | CPU-User% | CPU的使用率 |
| Memory | Memory-User% | JVM内存使用率 |

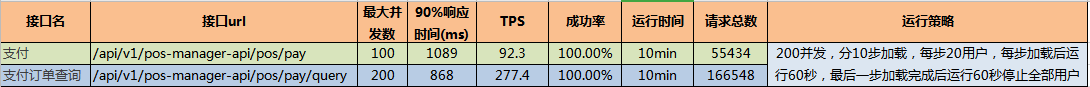
# 6测试结果分析

## 6.1性能概要

基准测试：

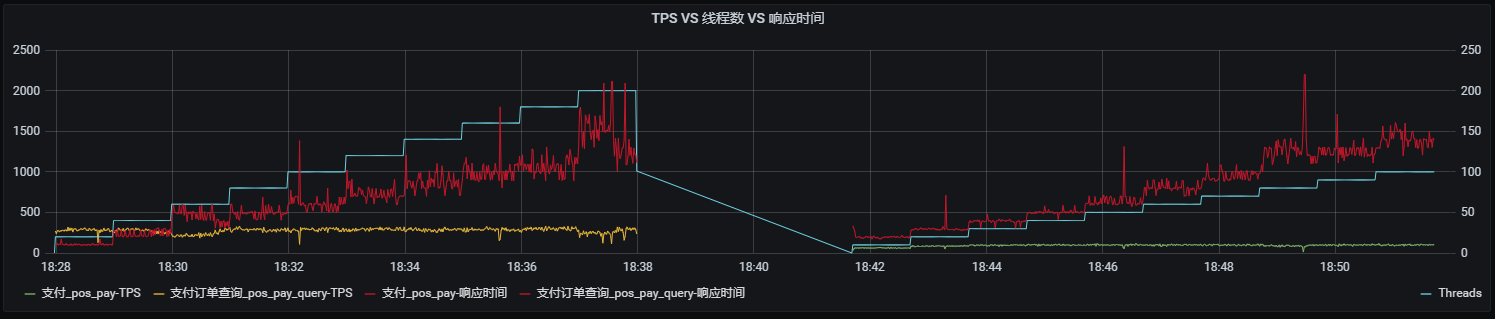


单场景负载测试



1. 支付接口模拟了微信支付100ms的耗时，最大并发数为100，整个负载过程中支付接口平均TPS为92.3，90%请求的响应时间在1089ms内
2. 支付订单查询接口最大并发数200，平均TPS为277.4，90%请求的响应时间在868ms内

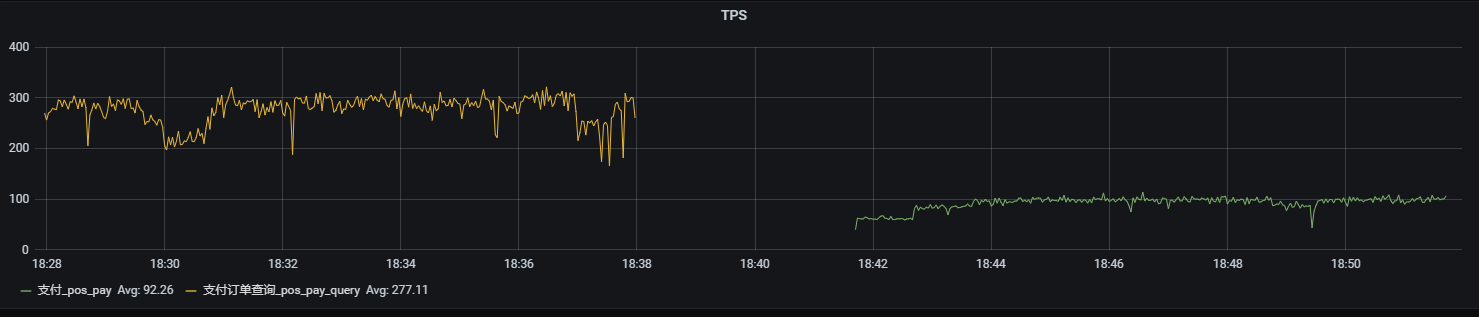
## 6.2单场景负载测试性能表现



1、支付接口发数为10-100，随着并发数的增加响应时间在不断增加，TPS呈现平衡状态

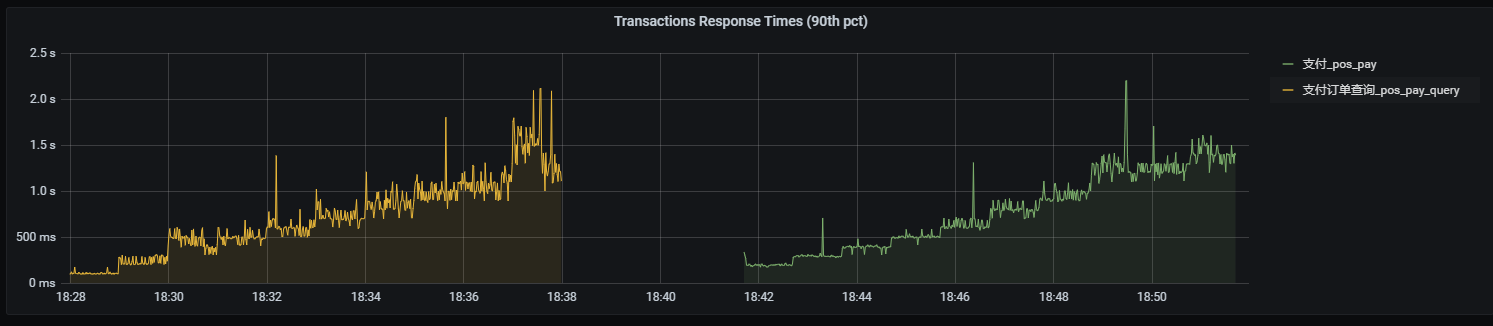
2、支付订单查询并发数为20-200，随着并发数的增加响应时间在不断增加，TPS呈现平衡状态

### 6.2.1系统处理能力



整个负载过程中各接口TPS比较平稳

### 6.2.2响应时间



### 6.2.3系统资源

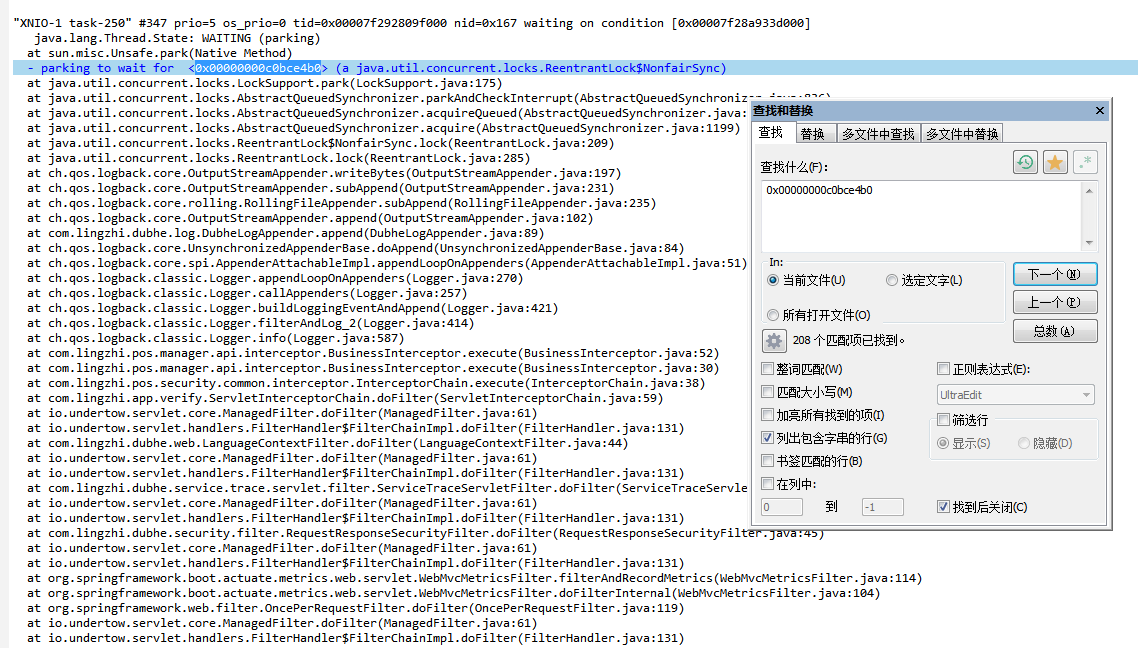


两个接口负载时服务器的CPU和内存都达到饱和状态，CPU已经成为瓶颈

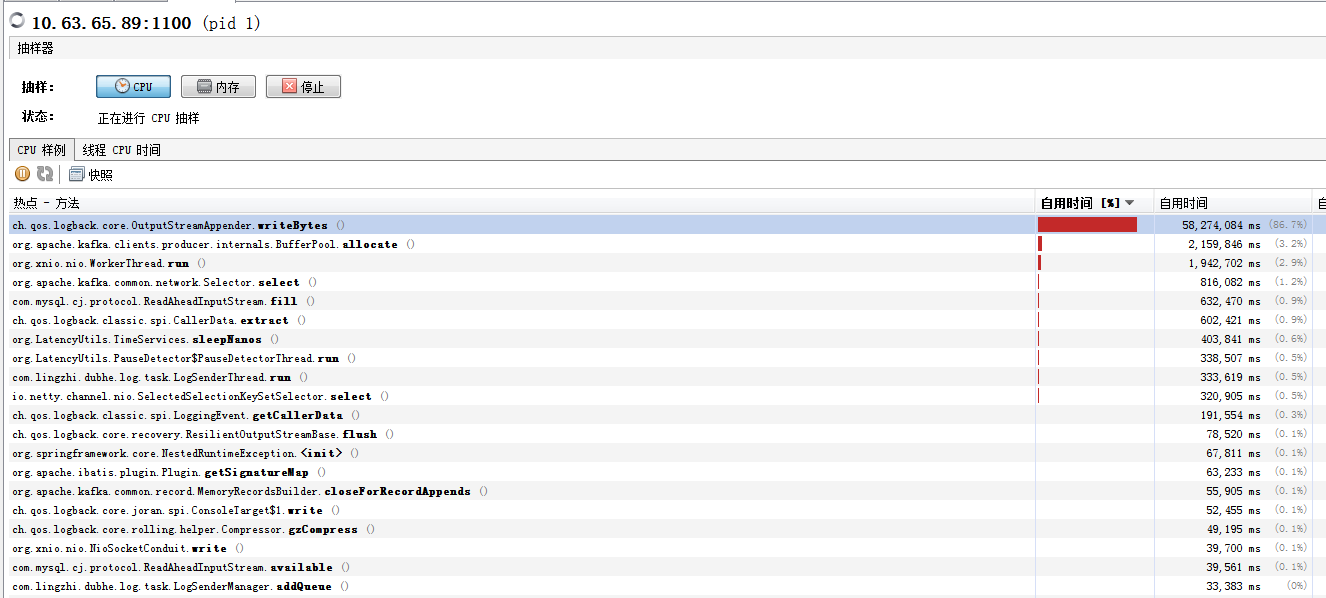
# 7综合评估测试结论

1. 不进行加、解密和签名验证(微信支付耗时100ms)情况下，负载时支付接口100并发，TTPS约为92.3，90%请求的响应时间在1089ms内，CPU使用率为**100%达到饱和**
2. 支付查询接口200并发，TPS约为277.4**，**90%请求的响应时间在868ms内，CPU使用率为**100%达到饱和**
3. 目前日志输出方式为同步输出，并发时存在大量线程等待写日志，logback同步写日志很消耗CPU，建议将日志输出方式改为异步

大量工作线程在等待写日志



通过监控发现logback的相应方法占用了大量的CPU



# 8附件

* 《收银项目性能测试结果记录》