**TA跳号问题修复测试报告**

1. **测试时间**：

2021/04/07-2021/04/09

1. **问题原因**：

在同一台机器操作时系统在同一时间发送了两条请求，当程序处理请求时，造成相 互调用，从数据库获取TA值产生异常，默认提供编号1。

1. **测试环境**：

Lab环境10.1.23.245，使用与生产环境1001门店相同的dll和spin程序。

1. **测试方案**：

使用Jmeter压测工具调用TPCS销售接口，模拟顾客使用收银系统购买商品，通过 控制请求的线程数和线程调用间隔模拟系统请求频率。

1. 复现TA跳号场景，定位产生问题的条件。
2. 安装修复程序，模拟生产门店正常压力和极端场景的交易情况。
3. 长时间模拟生产门店相符场景。
4. 确定可上线版本。
5. **测试数据**：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TA跳号测试** | | | | | | | | | | | | | |
| **状态** | **分组** | **编号** | **ClientName数** | **线程数** | **线程间隔** | **持续时间** | **开始日期** | **开始时间** | **结束时间** | **是否存在TA跳号** | **是否存在线程卡死** | **是否存在接口报错** | **备注** |
| 升级前 | 第一组 | 1 | 1 | 1 | 1 | 5min | 2021/4/7 |  |  | 否 | 否 | 否 |  |
| 2 | 1 | 4 | 1 | 5min | 2021/4/7 |  |  | 是 | 否 | 否 | 容易出现跳号 |
| 3 | 1 | 4 | 4 | 5min | 2021/4/7 |  |  | 是 | 否 | 否 | 容易出现跳号 |
| 4 | 1 | 2 | 1 | 5min | 2021/4/7 |  |  | 否 | 否 | 否 |  |
| 第二组 | 5 | 5 | 4 | 1 | 5min | 2021/4/7 |  |  | 否 | 否 | 否 |  |
| 6 | 5 | 8 | 1 | 5min | 2021/4/7 |  |  | 是 | 否 | 否 | 容易出现跳号 |
| 7 | 5 | 8 | 4 | 5min | 2021/4/7 |  |  | 否 | 否 | 否 |  |
| 8 | 5 | 5 | 4 | 5min | 2021/4/7 |  |  | 否 | 否 | 否 |  |
| 第一次升级 | 第三组 | 9 | 1 | 1 | 1 | 5min | 2021/4/8 | 13:05 | 13:10 | 否 | 否 | 否 |  |
| 10 | 1 | 4 | 1 | 5min | 2021/4/8 | 12:59 | 13:04 | 是 | 否 | 否 | 4笔跳号 |
| 11 | 1 | 4 | 4 | 5min | 2021/4/8 | 12:53 | 12:58 | 否 | 否 | 否 |  |
| 12 | 1 | 4 | 4 | 10min | 2021/4/8 | 14:28 | 14:38 | 是 | 否 | 否 | 1笔跳号 |
| 13 | 1 | 2 | 1 | 5min | 2021/4/8 | 10:47 | 10:52 | 否 | 否 | 否 |  |
| 第四组 | 14 | 5 | 4 | 1 | 5min | 2021/4/8 | 10:40 | 10:45 | 否 | 否 | 否 |  |
| 15 | 5 | 8 | 1 | 5min | 2021/4/8 | 10:32 | 10:37 | 否 | 否 | 否 |  |
| 16 | 5 | 8 | 1 | 10min | 2021/4/8 | 13:15 | 13:25 | 是 | 否 | 否 | 1笔跳号 |
| 17 | 5 | 8 | 4 | 5min | 2021/4/8 | 10:25 | 10:30 | 否 | 否 | 否 |  |
| 18 | 5 | 5 | 4 | 5min | 2021/4/8 | 10:18 | 10:23 | 否 | 否 | 否 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 针对易出现跳号的场景再次测试 | | | | | | | | | | | | | |
| **状态** | **分组** | **编号** | **ClientName数** | **线程数** | **线程间隔** | **持续时间** | **开始日期** | **开始时间** | **结束时间** | **是否存在TA跳号** | **是否存在线程卡死** | **是否存在接口报错** | **备注** |
| 第一次升级 | 第五组 | 1 | 1 | 4 | 1 | 5min | 2021/4/8 | 15:05 | 15:10 | 是 | 否 | 否 | 2笔跳号 |
| 2 | 1 | 4 | 4 | 5min | 2021/4/8 | 15:15 | 15:20 | 是 | 否 | 否 | 1笔跳号 |
| 3 | 5 | 8 | 1 | 5min | 2021/4/8 | 15:46 | 15:51 | 否 | 否 | 否 |  |
| 4 | 5 | 8 | 1 | 10min | 2021/4/8 | 15:46 | 15:56 | 否 | 否 | 否 |  |
| 5 | 5 | 8 | 4 | 5min | 2021/4/8 | 16:03 | 16:07 | 是 | 否 | 否 | 1笔跳号 |
| 6 | 5 | 8 | 1 | 5min | 2021/4/8 | 16:13 | 16:18 | 是 | 否 | 否 | 1笔跳号 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 第二版修复测试 | | | | | | | | | | | | | |
| **状态** | **分组** | **编号** | **ClientName数** | **线程数** | **线程间隔** | **持续时间** | **开始日期** | **开始时间** | **结束时间** | **是否存在TA跳号** | **是否存在线程卡死** | **是否存在接口报错** | **备注** |
| 第二次升级 | 第六组 | 1 | 1 | 4 | 4 | 20min | 2021/4/9 | 15:02 | 15:22 | 否 | 否 | 否 | 全量包dll |
| 2 | 1 | 4 | 4 | 20min | 2021/4/9 | 15:29 | 15:49 | 是 | 否 | 否 | 旧dll，出现3次跳号 |
| 3 | 2 | 8 | 4 | 20min | 2021/4/9 | 15:59 | 16:19 | 否 | 否 | 否 | 全量包dll |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 上线全量包测试 | | | | | | | | | | | | | |
| **状态** | **分组** | **编号** | **ClientName数** | **线程数** | **线程间隔** | **持续时间** | **开始日期** | **开始时间** | **结束时间** | **是否存在TA跳号** | **是否存在线程卡死** | **是否存在接口报错** | **备注** |
| 上线版测试 | 第七组 | 1 | 1 | 4 | 4 | 20min | 2021/4/9 | 16:27 | 16:47 | 否 | 否 | 否 | 新全量包 |
| 2 | 2 | 8 | 4 | 20min | 2021/4/9 | 16:48 | 17:27 | 否 | 否 | 否 | 新全量包 |

1. **测试分析**

正常压力，1台机器处理1条请求，1台机器处理2条请求时比较难模拟出TA跳号的情况。

增加压力，1台机器处理4条请求，修复前在5分钟和10分钟比较容易模拟出TA跳号的情况，程序第一次修复后出现的TA跳号的频率有所降低，单依旧存在跳号情况，第二次修复后，压力持续20分钟不再出现TA跳号情况，满足上线要求。

极端压力，1台机器处理4条请求，1秒内发送全部请求时，升级后存在低频率TA跳号的情况。