**LavaMQ性能测试报告**

文档管理信息表

|  |  |
| --- | --- |
| **主题** | LavaMQ性能测试报告 |
| **版本** | 1.0 |
| **内容** |  |
| **关键字** |  |
| **参考文档** |  |
| **创建时间** | 2020-01-10 |
| **创建人** | 王雷 |
| **最新发布日期** |  |

文档变更记录表

|  |  |  |
| --- | --- | --- |
| **修改人** | **修改时间** | **修改内容** |
| 王雷 | 2020-01-21 | 初版作成 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**目 录**

[1 测试说明 1](#_Toc29824655)

[1.1 测试项 1](#_Toc29824656)

[1.2 测试环境 1](#_Toc29824657)

[**1.2.1** 测试环境约定 1](#_Toc29824658)

[**1.2.2** Server端运行环境： 1](#_Toc29824659)

[**1.2.3** Client端运行环境： 1](#_Toc29824660)

[**1.2.4** 网络连接 1](#_Toc29824661)

[2 LavaMQ性能测试 1](#_Toc29824662)

[2.1 订阅推送性能测试 1](#_Toc29824663)

[**2.1.1** 场景一：单服务端对单客户端 1](#_Toc29824664)

[**2.1.2** 场景二：单服务端对双客户端 3](#_Toc29824665)

[**2.1.3** 场景三：单服务端对多客户端 4](#_Toc29824666)

[**2.1.4** 订阅推送性能测试结论 6](#_Toc29824667)

[2.2 异步请求响应性能测试 6](#_Toc29824668)

[**2.2.1** 场景一：单服务端对单客户端 6](#_Toc29824669)

[**2.2.2** 场景二：单服务端对双客户端 7](#_Toc29824670)

[**2.2.3** 异步请求响应性能测试结论 10](#_Toc29824671)

[2.3 同步请求响应性能测试 10](#_Toc29824672)

[**2.3.1** 场景：单服务端对单客户端 10](#_Toc29824673)

[**2.3.2** 同步请求响应性能测试结论 12](#_Toc29824674)

[3 LavaBox(C++)、Runme(Java)、LavaMQ(Java)性能测试对比 12](#_Toc29824675)

# 测试说明

## 测试项

LavaMQ性能测试主要关注订阅推送与异步请求响应两个测试项，同步请求响应测试会提供简要的测试数据列入本报告，其本质上与性能测试无关。

快照订阅仅仅是推送快照时会加同步锁(粒度为按topic锁定，其他topic的push不受影响)暂停push，待快照推送完毕解锁，后期流程与普通订阅推送一致，故性能测试以普通订阅推送的方式为主要参考。

## 测试环境

### 测试环境约定

为确保测试环境一致性，以利于性能参考与横向对比，做出如下简要约定：

1. 所有Server端，不论java的还是c++的，都统一在CentOS6台式机上运行。
2. 所有Client端，不论java的还是c++的，都统一在笔记本上运行。

### Server端运行环境：

台式机

系统：CentOS release 6.10 (Final)

CPU：Intel(R) Core(TM) i5-4460 CPU @ 3.20GHz 物理4核

内存：8G

### Client端运行环境：

笔记本

系统：Windows10

CPU：Intel Core i5-8350U 1.7G 物理4核，8逻辑核

内存：8G

### 网络连接

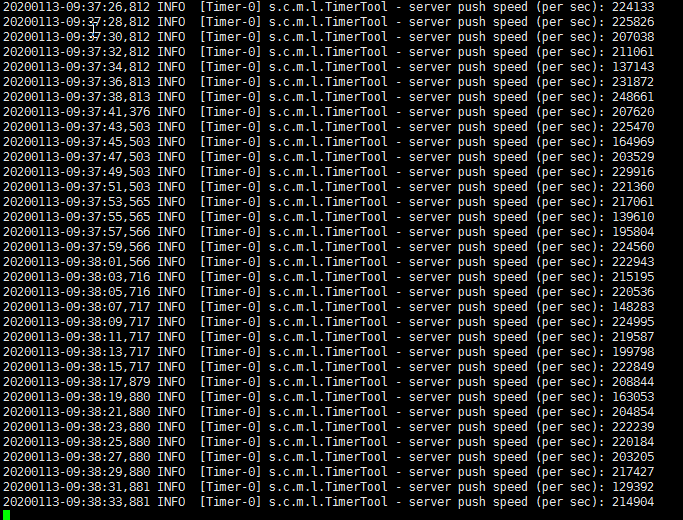
以太局域网，最大带宽1Gbps

# LavaMQ性能测试

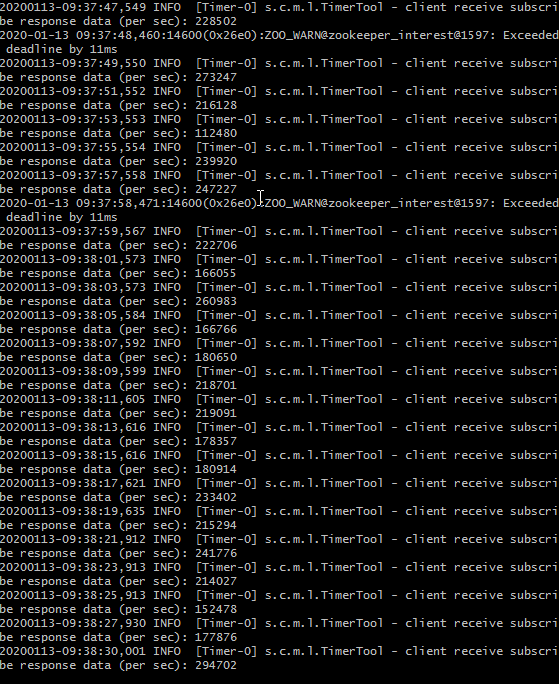
## 订阅推送性能测试

### 场景一：单服务端对单客户端

1. 一个LavaMQ Server负责推送数据，一个LavaMQ Client订阅并接收
2. Server端测试截图

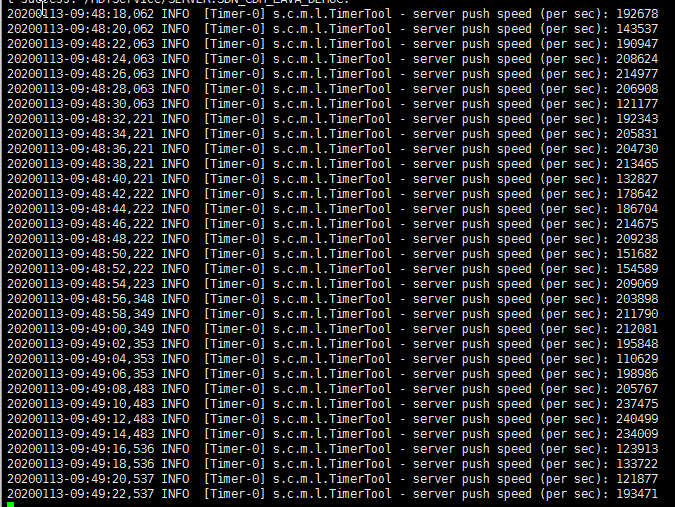


1. Client端测试截图

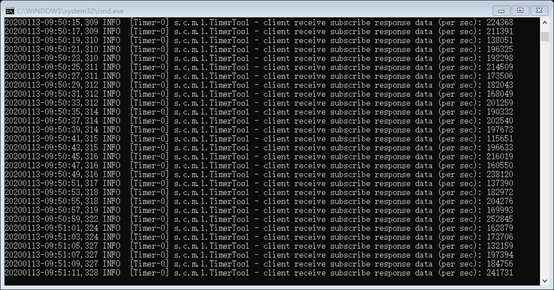


### 场景二：单服务端对双客户端

1. 一个LavaMQ Server负责推送数据，两个LavaMQ Client订阅并接收
2. Server端测试截图

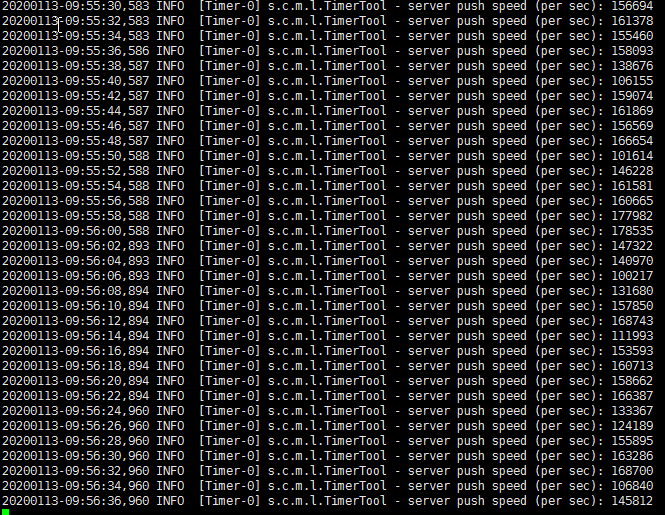


1. Client端测试截图

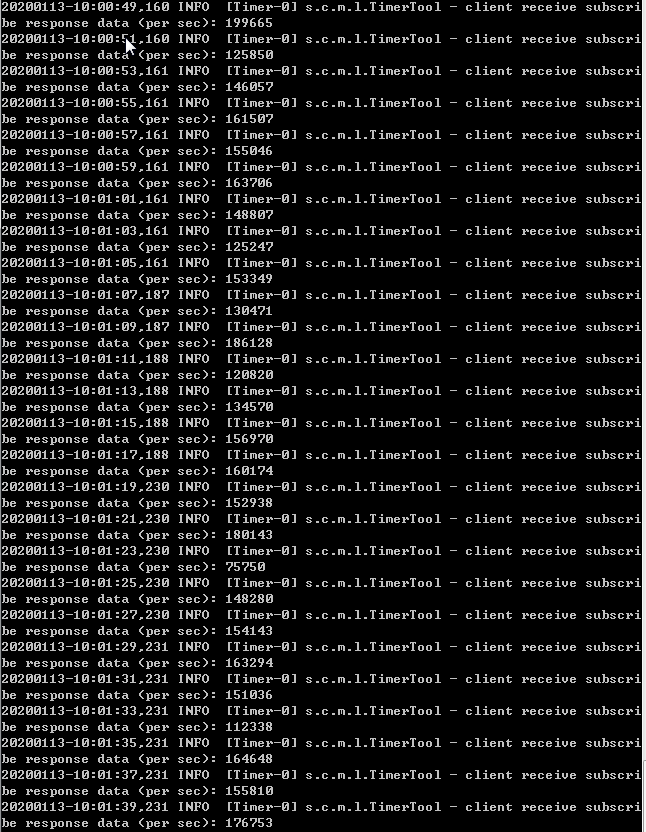


### 场景三：单服务端对多客户端

1. 一个LavaMQ Server负责推送数据，3个LavaMQ Client订阅并接收
2. Server端测试截图



1. Client端测试截图



### 订阅推送性能测试结论

服务端：

服务端推送能力在每秒15-25万之间，服务器配置越高越接近25万，客户端数目的增加基本对服务端的推送能力不构成影响。

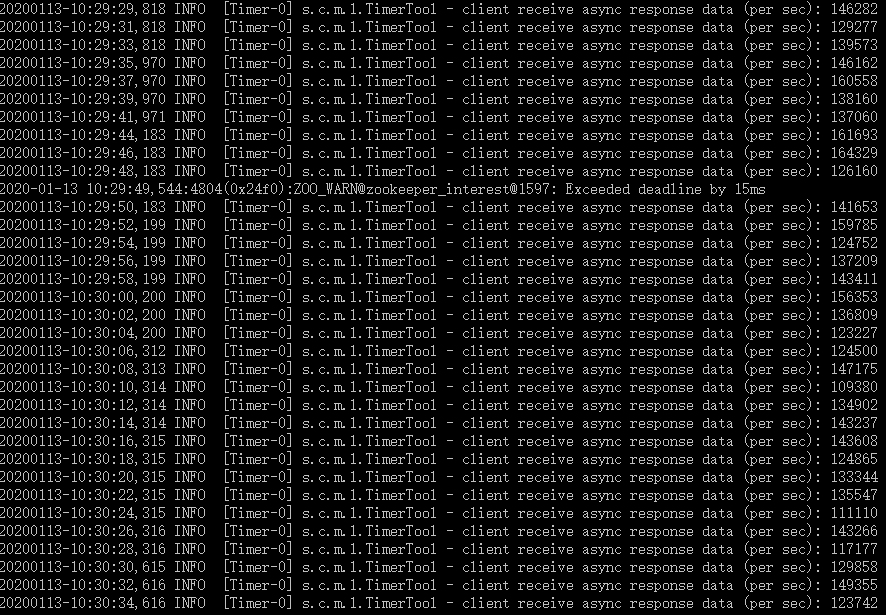
客户端：

客户端的接收能力在每秒15-25万之间，与服务端推送能力相匹配。

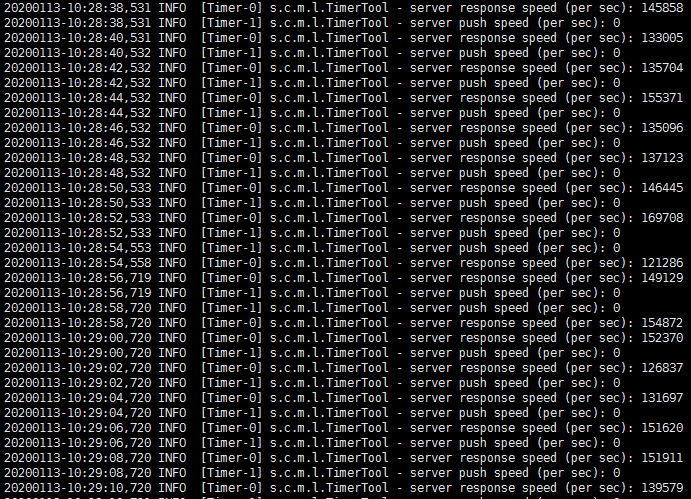
## 异步请求响应性能测试

### 场景一：单服务端对单客户端

Server端测试截图：

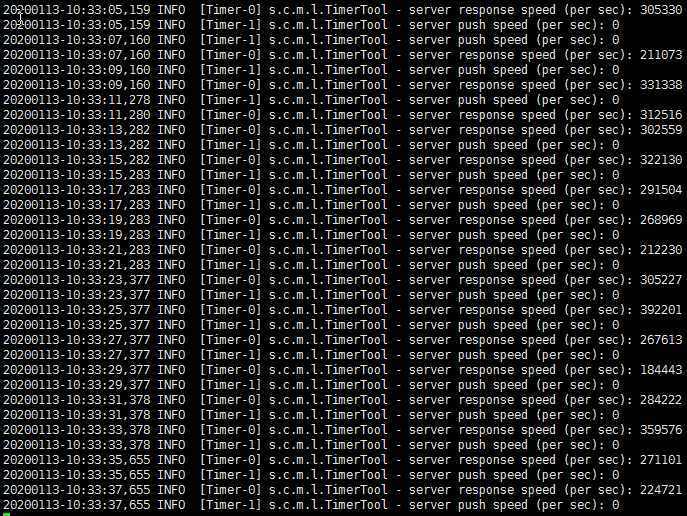


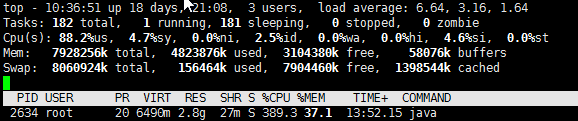
Client端测试截图：

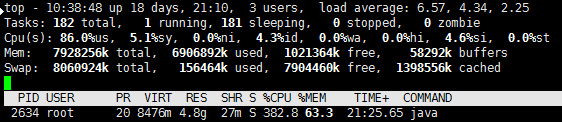


### 场景二：单服务端对双客户端

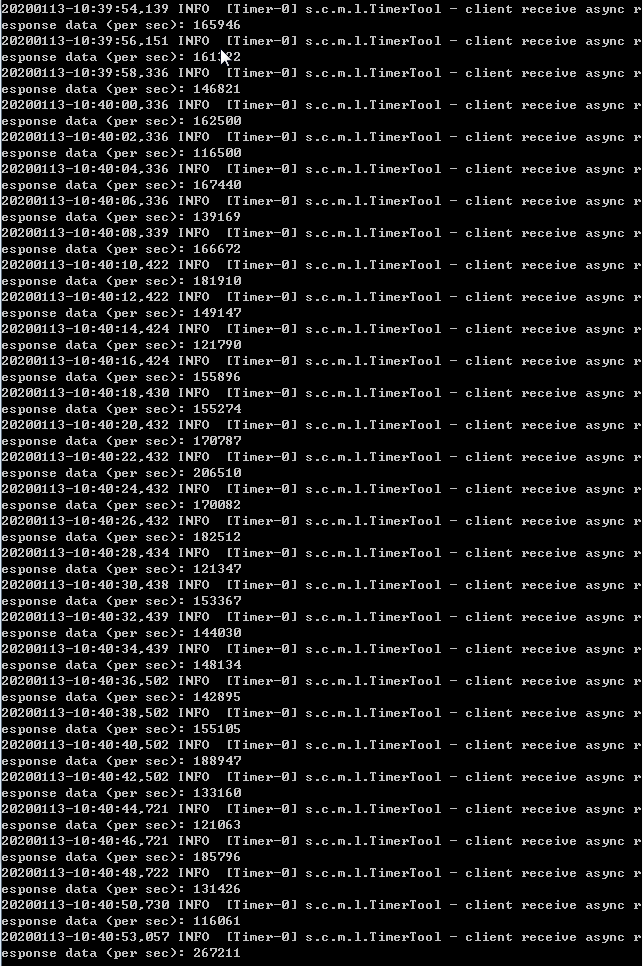
Server端测试截图：



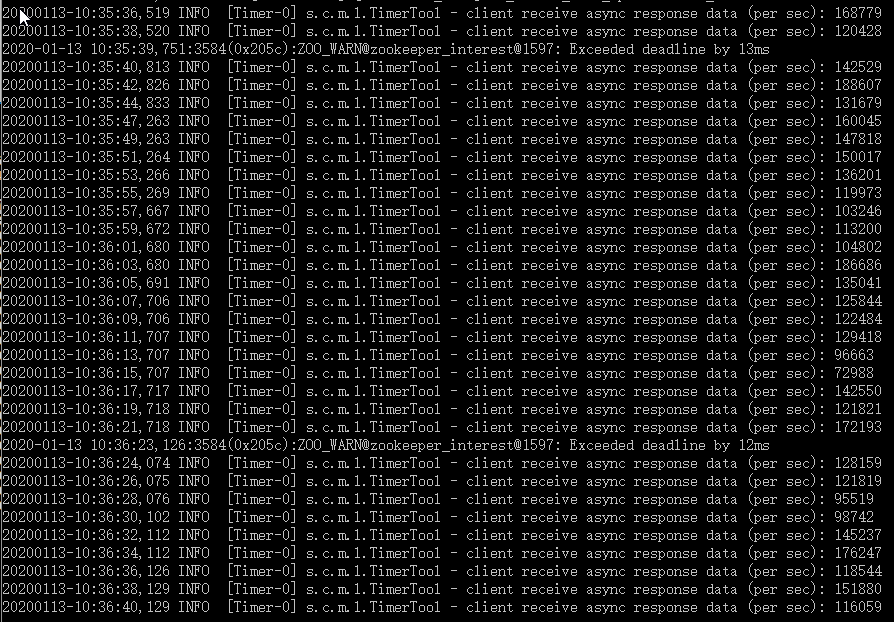




Client1测试截图：



Client2测试截图：



### 异步请求响应性能测试结论

服务端：

服务端处理性能与客户端发送量相关，目前服务端处理能力基本稳定在每秒10-15万之间，如果超过最大每秒15万的发送量，将导致服务端来不及处理消息，进而导致服务状态不稳定。

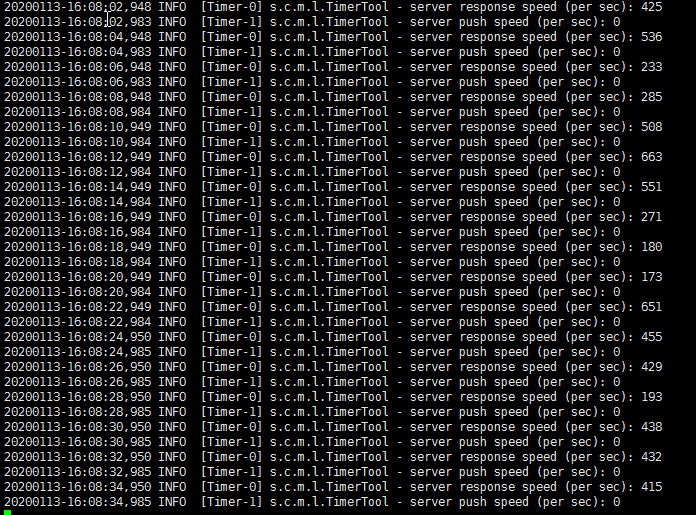
客户端：

单客户端的发送能力在10-15万左右，表现稳定。

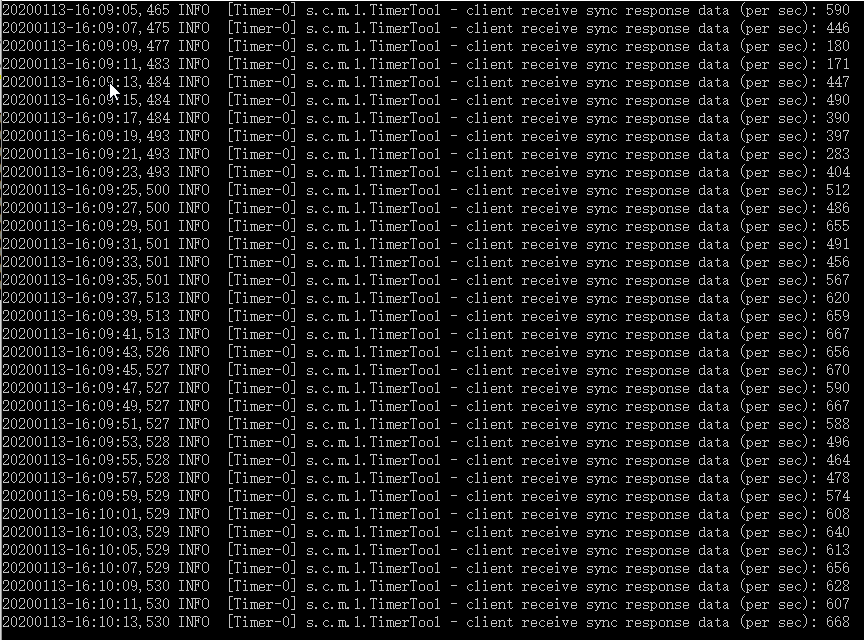
## 同步请求响应性能测试

### 场景：单服务端对单客户端

Server端测试截图：



Client端测试截图：



### 同步请求响应性能测试结论

同步请求响应只关注数据拉取，本质上与性能无关，这里列出性能测试数据仅提供参考，从数据可以看出其效率低下，不能用于与性能相关场合。

# LavaBox(C++)、Runme(Java)、LavaMQ(Java)性能测试对比

|  |  |  |  |
| --- | --- | --- | --- |
| 测试类型 | LavaBox(C++) | LavaMQ(Java) | Runme(Java) |
| 服务端推送 | 每秒25-35万 | 每秒15-25万 | 每秒15-25万 |
| 服务端异步响应 |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |