|  |  |  |
| --- | --- | --- |
| **测试目标：talent-aio框架收发速度**  **机 器：**   * CPU: i7 4790， * 内存：8G、 * 操作系统：windows7 (客户机和服务器位于同一台机器)   **测试步骤：**   1. 下载talent-aio所有代码，然后转到其根目录 2. 双击start-im-server.bat，启动im server 3. 双击start-im-client.bat, 启动im client 4. 在弹出界面中，点击”连接并进入群组”按钮 5. 点击“群发”按钮 6. 待消息收发完毕后，重复执行第5步，然后在界面中观察测试数据，多观察几次数据 | | |
| **测试项** | **测试结果** | **备注** |
| 接收处理消息数（每秒） | 消息条数：**283.5821万条79403.236KB(约80M)** | 不同CPU此值相差很大，譬如本人五年前的笔记本就只能测到114万条消息左右(宏基、i5、8G) |
| 发送消息数（每秒） | 消息条数：**283.5821万条**  **79403.236KB(约80M)** | 同上 |

|  |
| --- |
| **测试目标：talent-aio框架对TCP长连接数据 的支持能力**  **机 器：**   1. 服务器  * centos6.x, 虚拟机，一个4核E5 CPU，内存16G  1. 客户机11台  * windows，硬件没什么特别要求   **测试步骤：**   1. 服务器  * 修改centos操作系统参数，使之支持更大的长连接数，细节略（可百度之） * 把talent-aio\dist\talent-aio-examples-im-server-1.0.2.v20170303-RELEASE上传至centos并运行startup.sh（注意把脚本修改成: nohup java -Xms64m –Xmx8192m -Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.EPollSelectorProvider -XX:+HeapDumpOnOutOfMemoryError -Dtalent.aio.default.read.buffer.size=512 -XX:HeapDumpPath=./java\_talent-aio-im-server\_pid.hprof -jar talent-aio-im-server.jar &）  1. 客户机  * 修改talent-aio\dist\talent-aio-examples-im-client-1.0.2.v20170303-RELEASE/config/app.conf，使其ip指向服务器 ip * 把talent-aio\dist\talent-aio-examples-im-client-1.0.2.v20170303-RELEASE拷到各客户机并运行startup.bat * 在弹出的swing界面中，点击 “连接并进入群组”按钮   **测试结果：**11个客户机 ，每个客户机连16200个TCP连接，服务器一共承受**17.82万TCP长连接**，服务器内存只消耗**800M**，CPU使用率极低(平时都是在1%-10%间，收到心跳瞬间会峰一下) |