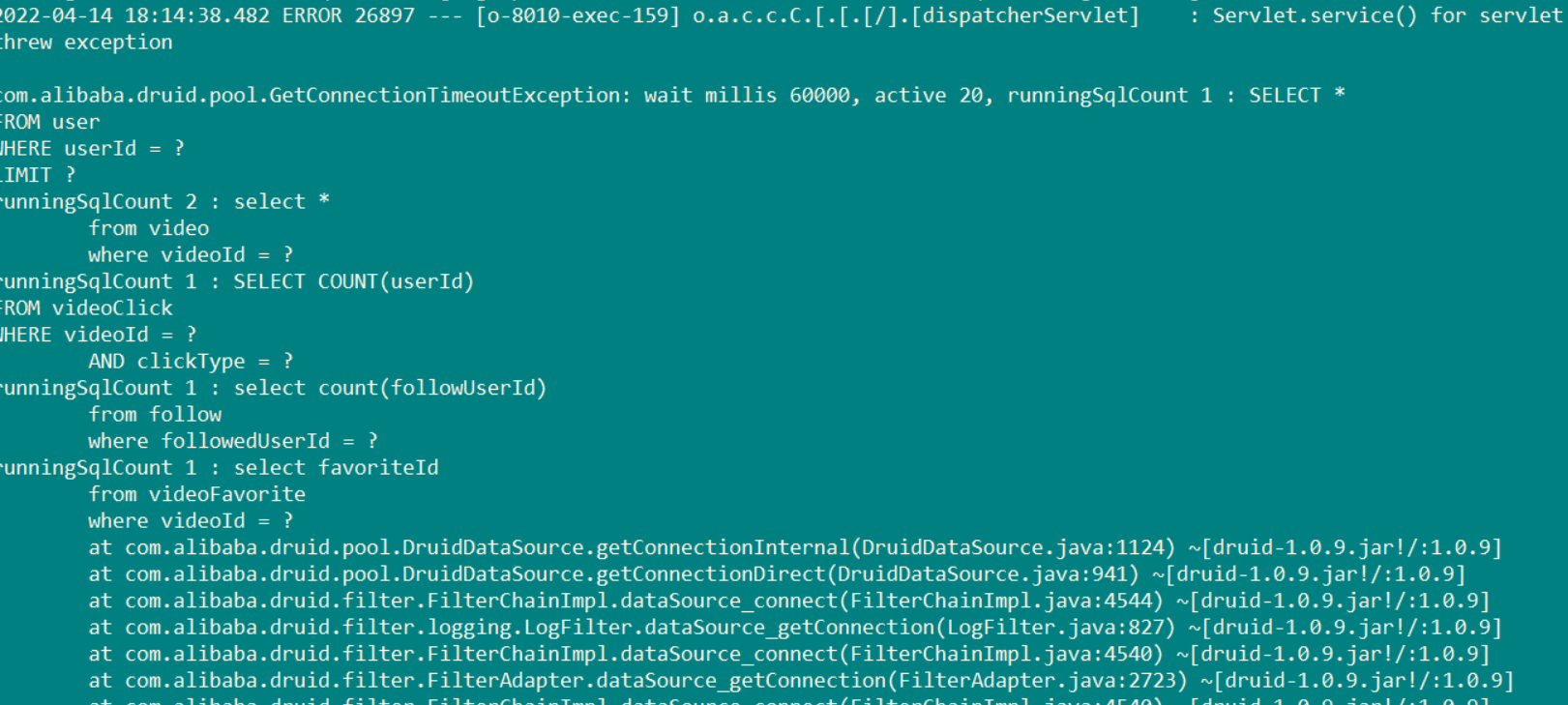
读测试(对热点视频的访问)

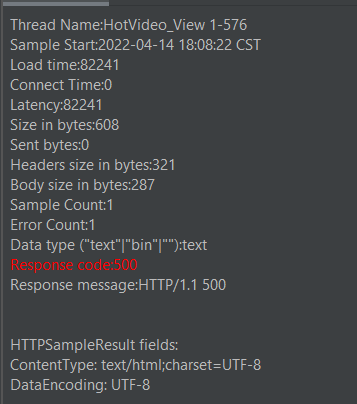
服务器带宽较小(1MB)，对数据传输(外)性能有硬性限制。

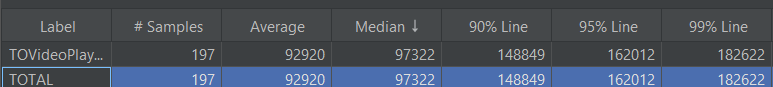
无优化(mvc)

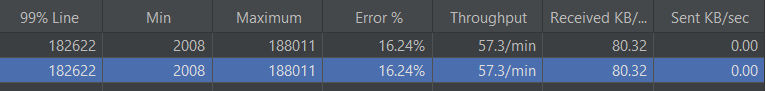
访问热点视频

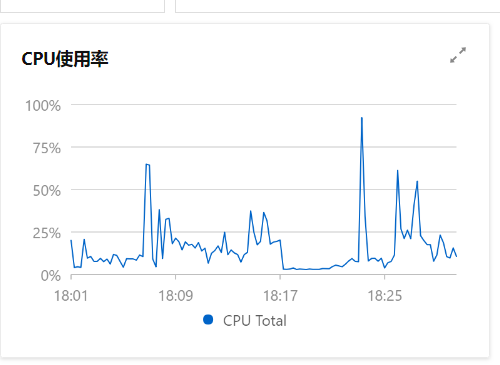
后台数据库出现了超时异常

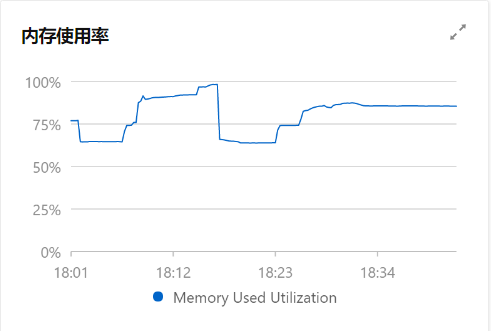


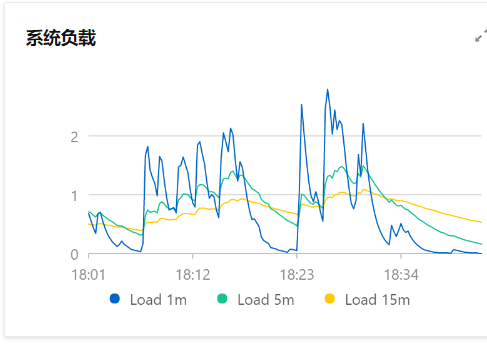


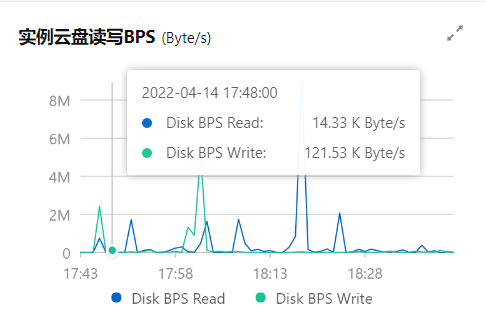


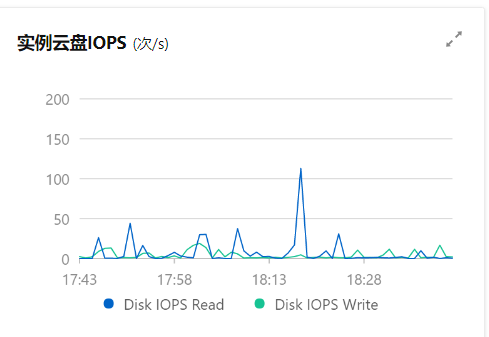


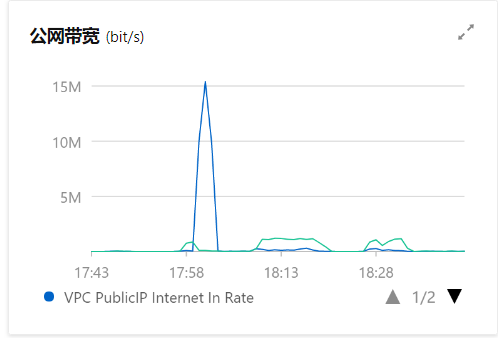


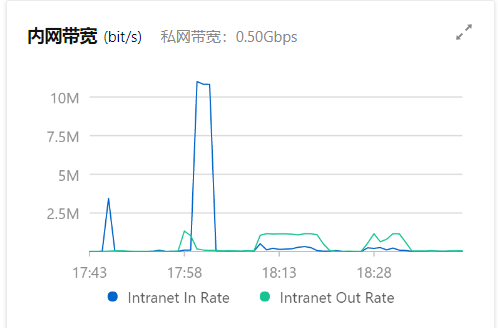




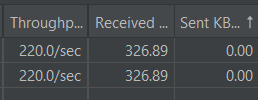
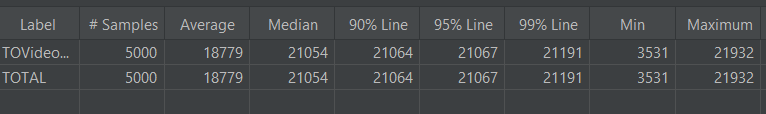








读测试



读取优化策略

Ⅰ 配置springboot中内嵌tomcat的配置文件，暴漏并优化tomcat keepAlive的配置

Ⅱ 热点视频内容存入redis缓存，使用数据预热，将一批热点数据提前加载到内存中，防z缓存雪崩。

Ⅲ redis集群,单机模拟一主二从，读写分离，搭配哨兵(sentinel)模式，即使主缓存(master)崩掉也能选举出新的主缓存。定时任务进行数据的持久化保存，保证可靠性。

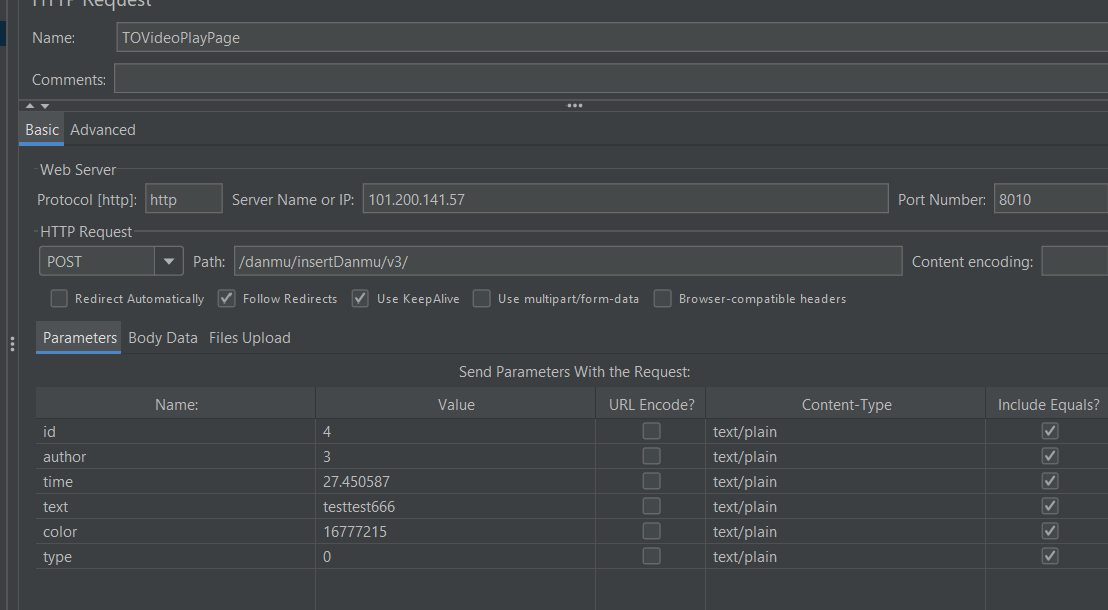
Ⅳ springboot2的redis底层使用以lettuce(线程安全，速度慢)实现，由于配置冲突，导致压力测试时出现超时和服务关闭的异常，更改为jedis。

Ⅴ nginx反向代理

带权轮询,开启多个端口，最大限度使用资源(单机模拟)。

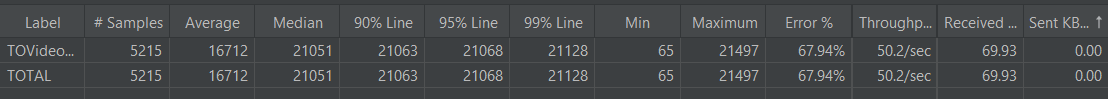
写测试：发送弹幕(热点视频高并发场景下弹幕系统的稳定性)

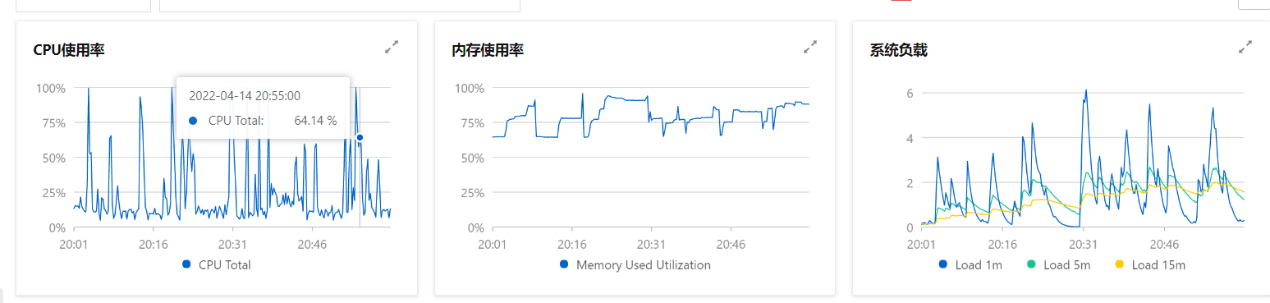
测试用例 线程5000 10000 20000



写测试(未优化)

使用mybatis对mysql进行写入，数据量大时错误率较高





写优化策略:Rabbitmq对短时间内发来的大量弹幕进行限流处理,每秒只向数据库写入少量的弹幕，保护数据库,在写入队列的同时将弹幕写入redis,保证消息的可靠性

