

图文 081、如何对线上系统的OOM异常进行监控和报警！

605 人次阅读 2019-09-24 07:00:00

详情 评论



狸猫技术窝

进店逛

如何对线上系统的OOM异常进行监控和报警！

狸猫技术窝专栏上新，基于真实订单系统的消息中间件（mq）实战，重磅推荐：



相关频道



从0开始
战高手
已更新1

未来3个月，我的好朋友原子弹大侠将带你一起，全程实战，360度死磕MQ

(点击下方蓝字进行试听)

[从0开始带你成为消息中间件实战高手](#)

重要说明：

如何提问：每篇文章都有评论区，大家可以尽情在评论区留言提问，我都会逐一答疑

(ps：评论区还精选了一些小伙伴对**专栏每日思考题**的作答，有的答案真的非常好！大家可以通过看别人的思路，启发一下自己，从而加深理解)

如何加群：购买了狸猫技术窝专栏的小伙伴都可以加入**狸猫技术交流群**。

(群里有不少一二线互联网大厂的助教，大家可以一起讨论交流各种技术)

具体**加群方式**请参见文末。

(注：以前通过其他专栏加过群的同学就不要重复加了)

1、前文回顾

我们之前已经把OOM的一些核心原理以及事发现场和一些典型案例都给大家讲清楚了，相信大家现在对OOM已经比较了解了

那么现在的一个核心问题就是一旦发生了各种场景下的OOM，我们到底应该如何处理呢？

所以本周将会从OOM问题的监控开始，给大家讲OOM的排查、定位和解决的一系列思路

然后下周开始再给大家分析多个各种场景下的奇怪的OOM案例，让大家去积累丰富的OOM问题解决经验。

2、最佳的解决方案

我们先给大家说一种最佳的OOM监控方案，其实说白了也很简单，之前一直给大家强调，公司最好是应该有一种监控平台，比如Zabbix、Open-Falcon之类的监控平台。

如果有监控平台的话，就可以接入系统异常的一些监控和报警，你可以设置一旦系统出现了OOM异常，就发送报警给对应的开发人员，通过邮件、短信或者钉钉之类的IM工具。

这个是中大型公司里最常用的一种方案了，一般来说我们都对线上系统有以下几个层面的监控：

机器（CPU、磁盘、内存、网络）资源的负载情况，JVM的GC频率和内存使用率，系统自身的业务指标，系统的异常报错。

这些东西都会基于监控平台接入对应的监控项，同时设定关键监控项的一些报警阈值。

下面我来分层给大家解释一下这个所谓的监控体系的思路，我们这个专栏虽然不是专门教大家做监控的，但是因为说到了系统的JVM监控，因此可以顺带给大家说一下思路，大家再结合自己公司的监控系统去考虑一下怎么做。

3、一个比较成熟的系统监控体系的建议

首先通过监控平台是可以看到你的所有线上系统所在的机器资源的负载情况的，比如CPU负载，这个可以看到现在你的CPU目前的使用率有多高，比如你的CPU使用率都达到100%了，此时一定有问题了，你得检查一下为什么CPU负载那么高。

而且可以看到你的机器上磁盘IO的一些负载，包括磁盘上发生了多少数据量的IO，一些IO的耗时等等。

当然一般的业务系统本身不会直接读写自己本地的磁盘IO，最多就是写一些本地日志而已。

但是你应该关注的是你本地磁盘的使用量和剩余空间，因为有的系统因为一些代码bug，可能会一直往本地磁盘写东西，万一把你的磁盘空间写满了就麻烦了，这样也会导致你的系统无法运行。

其次可以看到你机器上的内存使用量，这个是从机器整体层面去看的，看看机器对内存使用的一些变化。

当然内存这块，比较核心的还是JVM这块的监控，我们是可以看到JVM各个内存区域的使用量的一个变化的。

最后就是机器上的网络负载，就是通过网络IO读写了多少数据，一些耗时，等等。

还有一个比较关键的，就是JVM的Full GC的频率，这个一般会用一段时间内的Full GC次数来监控，比如5分钟内发生了几次Full GC。

其实线上机器最容易出问题的主要三大块，一个是CPU，必须要对CPU的使用率做一个监控，如果CPU负载过高，比如长期使用率超过90%，就得报警了；

一个是内存，同样得监控内存的使用率，如果机器内存长期使用率超过了一定的阈值，比如长期使用率超过90%，那肯定是有问题的，随时机器内存可能就不够了；

一个是JVM的Full GC问题，假设5分钟内发生了10次Full GC，那一定是频繁Full GC了。

所以建议大家去看看自己公司是否有监控平台，同时是否建立起来基本的监控体系，对CPU、内存、Full GC等核心问题进行了监控和自动报警。

另外比较常见的就是对系统的业务指标的监控，比如你可以在每次系统创建一个订单就上报一次监控，然后监控系统会收集你1分钟内的订单数量。然后你可以设定一个阈值，比如1分钟内要是订单数量超过了100就报警。

因为可能订单过多涉及到了一些刷单之类的行为，这就是业务系统的指标监控，这个都是你自己去进行指标上报的。

最后一个，就是系统所有的try catch中的异常报错，必须要接入报警，一旦有异常，都需要上报到监控平台，然后监控平台会告诉你，最近有一次异常，只要系统报错，你立马可以收到报警。

因此非常核心的一点，就是要对线上系统的异常进行监控，一旦JVM有OOM之类的问题可以立马感知到。

4、一种比较Low的JVM OOM问题的被动发现方法

如果大家实在没有上述那种监控和报警的体系，那就没办法主动感知到JVM OOM问题了，此时只能用最Low的土办法就发现OOM了。

被动发现OOM问题，主要靠两个：

第一个是线上系统假设因为OOM突然挂掉，此时一定会导致用户无法使用，然后迅速反馈到客服，客服反馈给你，你就知道自己的系统挂掉了。

第二个就是你必须去检查一下系统的线上日志，一般来说，系统有异常的时候，必须通过log4j之类的日志框架写入本地日志文件，如果这个都不做，那实在是没办法说什么了。只要你检查日志，就会发现之前给大家演示过的OOM问题。

5、本文总结

今天给大家总结了一下**发现OOM的两种办法**：一种是通过监控系统，一种是被动等待系统挂掉后客服来通知你

大家可以思考一下，在自己的公司，到底应该怎么来做这个OOM的监控，只有你监控到OOM问题了，接着才能去分析到底为什么会发生这个问题。

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

如何加群？

添加微信号：Lvgu0715_（微信名：绿小九），狸猫技术窝管理员

发送Jvm专栏的购买截图

由于是人工操作，发送截图后请耐心等待被拉群

最后再次提醒：通过其他专栏加过群的同学，就不要重复加了

狸猫技术窝其他精品专栏推荐：

[21天互联网java进阶面试训练营（分布式篇）](#)