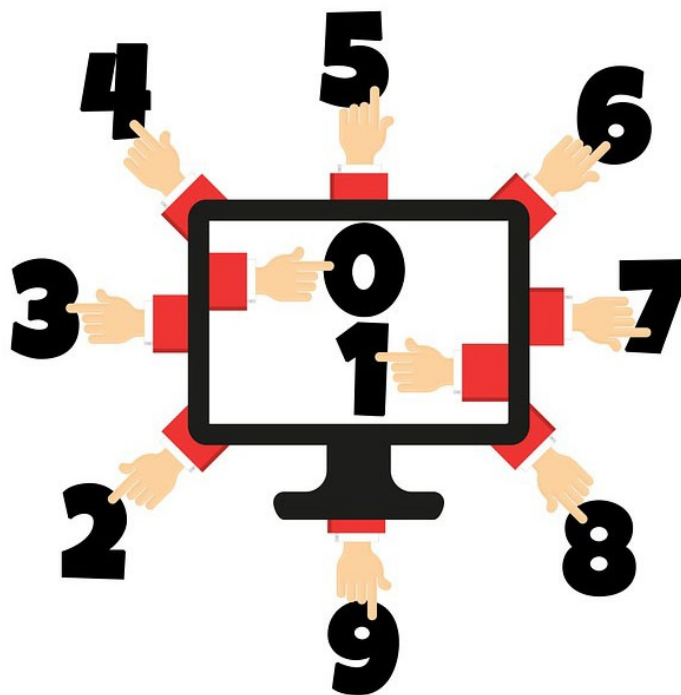


06 | Kafka线上集群部署方案怎么做？

2019-06-15 胡夕



专栏前面几期内容，我分别从Kafka的定位、版本的变迁以及功能的演进等几个方面循序渐进地梳理了Apache Kafka的发展脉络。通过这些内容，我希望你能清晰地了解Kafka是用来做什么的，以及在实际生产环境中该如何选择Kafka版本，更快地帮助你入门Kafka。

现在我们就来看看在生产环境中的Kafka集群方案该怎么做。既然是集群，那必然就要有多个Kafka节点机器，因为只有单台机器构成的Kafka伪集群只能用于日常测试之用，根本无法满足实际的线上生产需求。而真正的线上环境需要仔细地考量各种因素，结合自身的业务需求而制定。下面我就分别从操作系统、磁盘、磁盘容量和带宽等方面来讨论一下。

操作系统

首先我们先看看要把Kafka安装到什么操作系统上。说起操作系统，可能你会问Kafka不是JVM系的大数据框架吗？Java又是跨平台的语言，把Kafka安装到不同的操作系统上会有什么区别吗？其实区别相当大！

的确，如你所知，Kafka由Scala语言和Java语言编写而成，编译之后的源代码就是普通的“.class”文件。本来部署到哪个操作系统应该都是一样的，但是不同操作系统的差异还是给Kafka集群带来了相当大的影响。目前常见的操作系统有3种：Linux、Windows和macOS。应该说部署在Linux上的生产环境是最多的，也有一些Kafka集群部署在Windows服务器上。Mac虽然也有macOS Server，但是我怀疑是否有人（特别是国内用户）真的把生产环境部署在Mac服务器上。

如果考虑操作系统与Kafka的适配性，Linux系统显然要比其他两个特别是Windows系统更加适合部署Kafka。虽然这个结论可能你不感到意外，但其中具体的原因你也一定要了解。主要是在下面这三个方面上，Linux的表现更胜一筹。

- I/O模型的使用
- 数据网络传输效率
- 社区支持度

我分别来解释一下，首先来看I/O模型。什么是I/O模型呢？你可以近似地认为I/O模型就是操作系统执行I/O指令的方法。

主流的I/O模型通常有5种类型：阻塞式I/O、非阻塞式I/O、I/O多路复用、信号驱动I/O和异步I/O。每种I/O模型都有各自典型的使用场景，比如Java中Socket对象的阻塞模式和非阻塞模式就对应于前两种模型；而Linux中的系统调用select函数就属于I/O多路复用模型；大名鼎鼎的epoll系统调用则介于第三种和第四种模型之间；至于第五种模型，其实很少有Linux系统支持，反而是Windows系统提供了一个叫IOCP线程模型属于这一种。

你不必详细了解每一种模型的实现细节，通常情况下我们认为后一种模型会比前一种模型要高级，比如epoll就比select要好，了解到这一程度应该足以应付我们下面的内容了。

说了这么多，I/O模型与Kafka的关系又是什么呢？实际上Kafka客户端底层使用了Java的selector，selector在Linux上的实现机制是epoll，而在Windows平台上的实现机制是select。因此在这一点上将Kafka部署在Linux上是有优势的，因为能够获得更高效的I/O性能。

其次是网络传输效率的差别。你知道的，Kafka生产和消费的消息都是通过网络传输的，而消息保存在哪里呢？肯定是磁盘。故Kafka需要在磁盘和网络间进行大量数据传输。如果你熟悉Linux，你肯定听过零拷贝（Zero Copy）技术，就是当数据在磁盘和网络进行传输时避免昂贵的内核态数据拷贝从而实现快速地数据传输。Linux平台实现了这样的零拷贝机制，但有些令人遗憾的是在Windows平台上必须要等到Java 8的60更新版本才能“享受”到这个福利。一句话总结一下，在Linux部署Kafka能够享受到零拷贝技术所带来的快速数据传输特性。

最后是社区的支持度。这一点虽然不是什么明显的差别，但如果不了解的话可能比前两个因素对你的影响更大。简单来说就是，社区目前对Windows平台上发现的Kafka Bug不做任何承诺。虽然口头上依然保证尽力去解决，但根据我的经验，Windows上的Bug一般是不会修复的。因此，Windows平台上部署Kafka只适合于个人测试或用于功能验证，千万不要应用于生产环境。

磁盘

如果问哪种资源对Kafka性能最重要，磁盘无疑是要排名靠前的。在对Kafka集群进行磁盘规划时经常面对的问题是，我应该选择普通的机械磁盘还是固态硬盘？前者成本低且容量大，但易损

坏；后者性能优势大，不过单价高。我给出的建议是使用普通机械硬盘即可。

Kafka大量使用磁盘不假，可它使用的方式多是顺序读写操作，一定程度上规避了机械磁盘最大的劣势，即随机读写操作慢。从这一点上来说，使用**SSD**似乎并没有太大的性能优势，毕竟从性价比上来说，机械磁盘物美价廉，而它因易损坏而造成的可靠性差等缺陷，又由**Kafka**在软件层面提供机制来保证，故使用普通机械磁盘是很划算的。

关于磁盘选择另一个经常讨论的话题就是到底是否应该使用磁盘阵列（**RAID**）。使用**RAID**的两个主要优势在于：

- 提供冗余的磁盘存储空间
- 提供负载均衡

以上两个优势对于任何一个分布式系统都很有吸引力。不过就**Kafka**而言，一方面**Kafka**自己实现了冗余机制来提供高可靠性；另一方面通过分区的概念，**Kafka**也能在软件层面自行实现负载均衡。如此说来**RAID**的优势就没有那么明显了。当然，我并不是说**RAID**不好，实际上依然有很多大厂确实是把**Kafka**底层的存储交由**RAID**的，只是目前**Kafka**在存储这方面提供了越来越便捷的高可靠性方案，因此在线上环境使用**RAID**似乎变得不是那么重要了。综合以上的考量，我给出的建议是：

- 追求性价比的公司可以不搭建**RAID**，使用普通磁盘组成存储空间即可。
- 使用机械磁盘完全能够胜任**Kafka**线上环境。

磁盘容量

Kafka集群到底需要多大的存储空间？这是一个非常经典的规划问题。**Kafka**需要将消息保存在底层的磁盘上，这些消息默认会被保存一段时间然后自动被删除。虽然这段时间是可以配置的，但你应该如何结合自身业务场景和存储需求来规划**Kafka**集群的存储容量呢？

我举一个简单的例子来说明该如何思考这个问题。假设你所在公司有个业务每天需要向**Kafka**集群发送1亿条消息，每条消息保存两份以防止数据丢失，另外消息默认保存两周时间。现在假设消息的平均大小是1KB，那么你能说出你的**Kafka**集群需要为这个业务预留多少磁盘空间吗？

我们来计算一下：每天1亿条1KB大小的消息，保存两份且留存两周的时间，那么总的空间大小就等于 $1\text{亿} * 1\text{KB} * 2 / 1000 / 1000 = 200\text{GB}$ 。一般情况下**Kafka**集群除了消息数据还有其他类型的数据，比如索引数据等，故我们再为这些数据预留出10%的磁盘空间，因此总的存储容量就是220GB。既然要保存两周，那么整体容量即为 $220\text{GB} * 14$ ，大约3TB左右。**Kafka**支持数据的压缩，假设压缩比是0.75，那么最后你需要规划的存储空间就是 $0.75 * 3 = 2.25\text{TB}$ 。

总之在规划磁盘容量时你需要考虑下面这几个元素：

- 新增消息数

- 消息留存时间
- 平均消息大小
- 备份数
- 是否启用压缩

带宽

对于Kafka这种通过网络大量进行数据传输的框架而言，带宽特别容易成为瓶颈。事实上，在我接触的真实案例当中，带宽资源不足导致Kafka出现性能问题的比例至少占60%以上。如果你的环境中还涉及跨机房传输，那么情况可能就更糟了。

如果你不是超级土豪的话，我会认为你和我平时使用的都是普通的以太网络，带宽也主要有两种：1Gbps的千兆网络和10Gbps的万兆网络，特别是千兆网络应该是一般公司网络的标准配置了。下面我就以千兆网络举一个实际的例子，来说明一下如何进行带宽资源的规划。

与其说是带宽资源的规划，其实真正要规划的是所需的Kafka服务器的数量。假设你公司的机房环境是千兆网络，即1Gbps，现在你有个业务，其业务目标或SLA是在1小时内处理1TB的业务数据。那么问题来了，你到底需要多少台Kafka服务器来完成这个业务呢？

让我们来计算一下，由于带宽是1Gbps，即每秒处理1Gb的数据，假设每台Kafka服务器都是安装在专属的机器上，也就是说每台Kafka机器上没有混布其他服务，毕竟真实环境中不建议这么做。通常情况下你只能假设Kafka会用到70%的带宽资源，因为总要为其他应用或进程留一些资源。

根据实际使用经验，超过70%的阈值就有网络丢包的可能性了，故70%的设定是一个比较合理的值，也就是说单台Kafka服务器最多也就能使用大约700Mb的带宽资源。

稍等，这只是它能使用的最大带宽资源，你不能让Kafka服务器常规性使用这么多资源，故通常要再额外预留出2/3的资源，即单台服务器使用带宽 $700\text{Mb} / 3 \approx 240\text{Mbps}$ 。需要提示的是，这里的2/3其实是相当保守的，你可以结合你自己机器的使用情况酌情减少此值。

好了，有了240Mbps，我们就可以计算1小时内处理1TB数据所需的服务器数量了。根据这个目标，我们每秒需要处理2336Mb的数据，除以240，约等于10台服务器。如果消息还需要额外复制两份，那么总的服务器台数还要乘以3，即30台。

怎么样，还是很简单的吧。用这种方法评估线上环境的服务器台数是比较合理的，而且这个方法能够随着你业务需求的变化而动态调整。

小结

所谓“兵马未动，粮草先行”。与其盲目上马一套Kafka环境然后事后费力调整，不如在一开始就思考好实际场景下业务所需的集群环境。在考量部署方案时需要通盘考虑，不能仅从单个维度上

进行评估。相信今天我们聊完之后，你对如何规划Kafka生产环境一定有了一个清晰的认识。现在我来总结一下今天的重点：

因素	考量点	建议
操作系统	操作系统I/O模型	将Kafka部署在Linux系统上
磁盘	磁盘I/O性能	普通环境使用机械磁盘，不需要搭建RAID
磁盘容量	根据消息数、留存时间预估磁盘容量	实际使用中建议预留20% ~ 30%的磁盘空间
带宽	根据实际带宽资源和业务SLA预估服务器数量	对于千兆网络，建议每台服务器按照700Mbps来计算，避免大流量下的丢包

开放讨论

对于今天我所讲的这套评估方法，你有什么问题吗？你还能想出什么改进的方法吗？

欢迎你写下自己的思考或疑问，我们一起讨论。如果你觉得有所收获，也欢迎把文章分享给你的朋友。

 极客时间

Kafka 核心技术与实战

全面提升你的 Kafka 实战能力

胡夕

人人贷计算平台部总监
Apache Kafka Contributor



新版升级：点击「👤请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。



1000*1000/(60*60)=277,这个2336MB是怎么换算来的,还有什么要考虑的吗?

2019-06-15

作者回复

277MB, 乘以8, 大致等于2300+Mb (小b)。带宽资源一般用Mbps而不是MBps衡量

2019-06-15



A_NATE_

3

我们曾经也认为用普通硬盘就行, 换成普通硬盘导致生产者堵塞写入负载偏高, 换成SSD就没事了, 我们每天消息数大概50亿。

2019-06-15



Nero

2

好了, 有了 240Mbps, 我们就可以计算 1 小时内处理 1TB 数据所需的服务器数量了。根据这个目标, 我们每秒需要处理 2336Mb 的数据, 老师这一段中2334Mb是怎么算出来的, 没看懂, 能否解释一下, 谢谢。

2019-06-15

作者回复

1024*1024/3600*8

2019-06-15



QQ怪

1

老师尽然把我模糊不清的GB和Gb的区别给复习了一遍

2019-06-15



我自成魔

1

您好! 我在使用Kafka中, 出现消息阻塞不消费的问题, 换了消费组之后过段时间又不消费了, 不知什么原因, 望老师解惑。谢谢!

2019-06-15

作者回复

两个可能的原因, ①就是没有新消息可供消费了, ②某天消息格式有问题导致解析不了了, 不过这种情况很罕见, 一般是因为网络传输出问题导致

2019-06-15



Royal

1

您好, 我想请教下kafka metric相关的知识, 比如kafka produce速率获取等

2019-06-15

作者回复

kafka producer速率可以监控这个JMX指标:

```
kafka.producer:type=[consumer|producer|connect]-node-metrics,client-id=([-.\w]+),node-id=([0-9]+)
```

2019-06-15



zhisheng

0



zhihengy

老师，如果要跨集群不停机迁移 **Kafka** 有什么可供参考的方案吗？

0

2019-06-15



Royal

您好，我的意思是**Kafka produce**等metric是怎么计算的？老师可以解答下么

0

2019-06-15



曹飞

老师您好，您文章中讲到磁盘方面可以抛弃**RAID**,直接使用普通磁盘上去。那么一旦这块磁盘出了问题了，数据不就丢失了吗。如果**raid**的话，还有其他磁盘有相应的冗余数据的。麻烦帮忙解答下，谢谢！！

0

2019-06-15

作者回复

Kafka 1.1开始正式支持**JBOD**了。再说**Kafka**本身在软件层面也提供了冗余的机制来对抗磁盘损坏。

2019-06-16



WL

有三个问题请教一下老师：

1. 上文提到对于千兆网卡**kafka**服务器最多使用**700M**的带宽资源，这**700M**的资源是单机使用的还是集群共用的，为什么不能作为常规使用呢？
2. 文章举例是1小时**1T**的数据处理目标，那一秒中是不是 $1024/3600 = 0.284G = 285M$ ，请问下文中的**2336M**是咋算出来的。
3. 文章中的例子**kafka**单机要达到**240M**的读写能力，**CPU**应该配几核的？

2019-06-15

作者回复

1. 这个**700Mb**只是经验值罢了。另外预留**buffer**的意思是即使你最好不要让**broker**常规占用**700Mb**的资源。一旦碰到峰值流量，很容易将带宽打满。故做了一些资源预留
2. **285M**是大B，即字节啊，乘以8之后就是**2336Mb**。带宽资源一般用**Mbps**而非**MBps**衡量
3. 我没有谈及**CPU**，是因为通常情况下**Kafka**不太占用**CPU**，因此没有这方面的最佳实践出来。但有些情况下**Kafka broker**是很耗**CPU**的：1. **server**和**client**使用了不同的压缩算法；2. **server**和**client**版本不一致造成消息格式转换；3. **broker**端解压缩校验

其中前两个都能规避，第三个目前无法规避。不过相比带宽资源，**CPU**通常都不是瓶颈

2019-06-15



刚子

kafka 的分区数量的设置需要参考每秒传输的字节数计算吗？谢谢老师

0

2019-06-15

作者回复

通常不必这么细粒度。网上有一些分区制定的建议，我觉得这个粗粒度的方法就很好，值得一

试:

1. 首先你要确定你业务的SLA, 比如说你希望你的producer TPS是10万条消息/秒, 假设是T1
2. 在你的真实环境中创建一个单分区的topic测试一下TPS, 假设是T2
3. 你需要的分区数大致可以等于 $T1 / T2$
- 3.

2019-06-15



Nero

0

明白了, 这个是小转大要乘以8, 谢谢老师

2019-06-15



兆祥

0

带宽是bit, 规划的时候要除以八

2019-06-15



墙角儿的花

0

弱弱的问一句老师, “根据这个目标, 我们每秒需要处理 2336Mb 的数据, 除以 240, 约等于 10 台服务器”, 机房入口带宽 1Gbps, 怎么能做到1秒处理2336Mb的数据的

2019-06-15

作者回复

这里是指单机带宽, 机房总带宽不可能这么小的。。。

2019-06-15



Fouy_飞虎

0

您好, 对于“带宽是 1Gbps, 即每秒处理 1Gb 的数据”。1Gbps传输的是byte, 我们通常说的存储是字节为单位的。8byte=1字节, 所以我认为应该是“带宽是 1Gbps, 即每秒处理 1/8 Gb 的数据”

2019-06-15

作者回复

带宽单位一般是小B, 也就是比特。另外字节就是BYTE啊

2019-06-15



莱茵金属

0

能讲讲kafka的性能测试脚本怎么使用吗?

2019-06-15

作者回复

直接敲命令有参数提示, 可以详细看下。另外我的建议是: kafka-producer-perf-test脚本还不错, kafka-consumer-perf-test有点难用

2019-06-15



13501018051

0

zookeeper部署方案和kafka一样么? 分开部署还是部署到一起? 我们现在都是一个机器上部署一个zookeeper和一个kafka

2019-06-15

作者回复

如果配置好ZooKeeper事务日志（比如设置好`autopurge.purgeInterval`及定期删除snapshot等），它对IO的需求不大，混布也是可以的。

2019-06-15