

Apache Kafka 2.3 发布，新特性讲解

Apache Kafka 近期发布了 2.3.0 版本，主要的新特性如下：

- Kafka Connect REST API 已经有了一些改进。
- Kafka Connect 现在支持增量协同重新均衡（incremental cooperative rebalancing）
- Kafka Streams 现在支持内存会话存储和窗口存储；
- AdminClient 现在允许用户确定他们有权对主题执行哪些操作；
- broker 增加了一个新的启动时间指标；
- JMXTool 现在可以连接到安全的 RMI 端口。
- 添加了增量 AlterConfigs API。旧的 AlterConfigs API 已被弃用。
- 那些低于最小 ISR 计数的分区现在是可以被监控的；
- Consumers 现在可以选择不自动创建 topic，即使在 broker 启用也是如此；
- Kafka 组件现在可以使用外部配置存储（KIP-421）
- 改进了在遇到错误时副本提取程序的行为。

主要特性讲解

KIP-351和KIP-427：改进了对 lost replicas 的分区监控

为了保证 Kafka 的数据安全，Kafka 在不同的 Broker 上创建了几个副本。除非分区具有最小数量的同步副本，否则 Kafka 将不允许继续写入，这被称为“最小ISR”。

Kafka 已经有指标可以显示分区数量少于同步副本的最小数量。在此版本中，KIP-427 添加了其他指标，显示具有最小同步副本数的分区。通过监视这些指标，用户可以看到哪些分区的副本将不足，从而影响写。

此外，KIP-351 将 `-under-min-isr` 标签添加到 `kafka-topics` 命令里面。这使用户可以轻松查看哪些主题的副本数量少于最小 ISR 数量。

KIP-354：添加最大日志压缩延迟

对于一阶近似（first-order approximation），压缩主题中的键的先前值在写入最新密钥之后的某个时间被压缩，只有最新值可用，而以前的值不可用。但是，始终可以设置密钥在压缩之前保持的最短时间，因此我们不会太快丢失旧值。现在，使用 KIP-354，可以设置旧值将保持的最大时间量。新参数 `max.log.compatation.time.ms` 指定旧值可能存在于压缩主题中的时间长度，这可用于遵守GDPR等数据保留规定。

KIP-402：提高 SocketServer 处理器的公平性

以前，Kafka 会优先考虑在处理现有连接时打开新的 TCP 连接。如果客户端尝试在短时间内创建许多新连接，则可能会出现这个问题。KIP-402 将现有连接优先用于新连接，从而提高了 broker 对连接风暴的恢复能力。这个KIP 还为每个 broker 添加了 `max.connections` 参数。

KIP-461：改进 Replica Fetcher 中的故障处理

为了使副本保持最新，每个 broker 都维护一个 Replica Fetcher 线程池。池中的每个线程负责为一些跟随者分区获取副本。以前，如果其中一个分区失败，整个线程将失败，从而导致可能数百个分区的副本不足。使用此KIP，如果给定副本提取程序线程管理的单个分区失败，则该线程将继续处理其余分区。

KAFKA-7283：减少 broker 在启动时扫描日志文件所花费的时间

当 broker 在不正常关闭后启动时，它会检查日志以确保它们没有被破坏。此 JIRA 优化该过程，以便 Kafka 仅检查尚未显式刷新到磁盘的日志段。现在，日志恢复所需的时间不再与日志数成比例。相反，它与未刷新的日志段的数量成比例。Zhanxiang Huang 在 JIRA 上讨论的一些基准测试显示，broker 的启动时间缩短了50%。
更多关于 Apache Kafka 2.3.0 的新特性可以参见：<https://www.confluent.io/blog/whats-new-in-apache-kafka-2-3>。

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。
本文链接: [【】](#)（）