

Apache Pulsar 2.6.1 版本正式发布：功能增强，新增 OAuth2 支持

在 Apache Pulsar 2.6.0 版本发布后的 2 个月，2020 年 8 月 21 日，Apache Pulsar 2.6.1 版本正式发布！



cloud-native
streaming
Apache Pulsar
下一代云原生分布式消息流平台
messaging

如果想及时了解 Spark、Hadoop 或者 Hbase 相关的文章，欢迎关注微信公共帐号：iteblog_hadoop

Apache Pulsar 2.6.1 修复了 2.6.0 版本中的诸多问题，改进了一些功能，新增了对 OAuth2 的支持，覆盖 Broker、Pulsar SQL、Pulsar Functions、Go Function、Java Client 和 C++ Client，进一步丰富了 Pulsar 作为云原生流数据平台的功能。

在 Pulsar 2.6.1 版本中，来自社区的 commit 有 102 个，越来越多的小伙伴开始参与到 Pulsar 社区建设中，成为 Contributor 的一员。下面一起来看看 2.6.1 版本有哪些更新吧。

Broker 相关改进

将批处理大小限制为 “maxNumberOfMessages” 和 “maxSizeOfMessages” 的最小值

在 2.6.0 版本之前，BatchReceive 策略中 maxNumberOfMessages 和 maxSizeOfMessages 的最小值不会影响批处理大小。当批量大小大于 consumer 中设置的 receiveQueue 大小时（假设使用的批量大小为 3000，receiveQ 为 500），会出现以下问题：

在 consumer 中使用多主题模式，client 被阻塞，导致不接收任何消息；即使用户在批处理中设置超时策略，client 也不会恢复。

在 2.6.1 版本中，我们把批处理大小设置为 “maxNumberOfMessages” 和 “maxSizeOfMessages” 中的最小值，修复了该问题。

更多详情查看 PR-6865：<https://github.com/apache/pul...>

解决 Key_Shared 中使用粘性 hash range 导致的哈希范围冲突问题

在以前的版本中，当用户在 Key_Shared 订阅模型中使用 “stickyHashRange” 时，consumer 指定的 hash 范围不允许重叠。例如，consumer-1 的哈希范围为：[[0,99], [400,65535]]，consumer-2 的哈希范围为：[[100,399]]。

这是因为在 broker 端，没有对 stick hash range 中的 start 和 end 位置进行检查。正常情况下不允许 start 大于 end 的位置。在 2.6.1 版本中，我们加入了相应的 check 机制，来避免出现 hash range 冲突的问题。

更多详情查看 PR-7231：<https://github.com/apache/pul...>

修复获取 lookup 权限的错误

当前，当 Pulsar AuthorizationService 检查 lookup 权限时，拥有 canProducer 或 canConsumer 角色应该具备可以 canLookup 的能力，但实际上并没有该能力。代码如下：

```
try {  
    return canLookupAsync(topicName, role, authenticationData)  
        .get(conf.getZookeeperOperationTimeoutSeconds(), SECONDS);  
}
```

如果 canProduce 或 canConsume 方法抛出异常，canLookup 只会抛出该异常，不检查其他权限。

在 2.6.1 版本中，使用 canLookupAsync 代替原来的行为，更多详情查看 PR-7234：<https://github.com/apache/pul...>

修复创建 non-durable cursor 时无法删除 topic 的错误

当非持久游标创建失败时，会返回 NPE。因为程序发生 NPE 后，仍在继续创建订阅实例：

```
try {  
    cursor = ledger.newNonDurableCursor(startPosition, subscriptionName);  
} catch (ManagedLedgerException e) {  
    subscriptionFuture.completeExceptionally(e);  
}  
return new PersistentSubscription(this, subscriptionName, cursor, false);
```

将导致该 topic 的引用计数加一。当用户想要删除这个 topic 时，由于引用计数没有清零，所以即使使用 --force 强制删除，也无法删除 topic。在 2.6.1 版本中，我们解决了无法删除 topic 的问题。

更多详情查看 PR-7355：<https://github.com/apache/pul...>

避免在 ManagedLedgerImpl.isOffloadedNeedsDelete 方法中发生 NPE

在 2.6.1 版本之前，offload-deletion-lag 的默认值为 null，导致了 NPE 问题。在 2.6.1 版本中，我们在 ManagedLedgerImpl.isOffloadedNeedsDelete 方法中添加对 null 值的检查，避免出现该问题。

更多详情查看 PR-7389：<https://github.com/apache/pul...>

修复创建新 ledger 时引发 NPE 导致生产者卡死的问题

由于无法解析网络地址，在创建 ledger 时会引发 NPE。如果在添加超时任务之前引发了 NPE，则超时机制不起作用。无法解析的网络地址在 Kubernetes 环境中很常见。当 bookie pod 或工作程序节点重新启动时，可能会发生这种情况。

在 2.6.1 版本中，可通过以下操作来修复该问题：

- 在创建一个新的 ledger 时，捕获这个 NPE；
- 触发超时任务时，始终执行回调。因为回调只能触发一次；
- 添加机制检测“CreatingLedger”状态是否发生变化。

更多详情查看 PR-7401：<https://github.com/apache/pul...>

修复使用 advertisedListeners 产生的 NPE 问题

当使用带有外部 listener 名称的 advertisedListeners = internal : pulsar : // node1 : 6650 , external : pulsar : //node1.external : 6650 时，broker 无法获取名称空间包的所有权。如果未启用 TLS，我们需要更改 BrokerServiceUrIs。

更多详情查看 PR-7620：<https://github.com/apache/pul...>

获取最后一条 entry 时，client 错误地读取 -1 这条 entry

在 2.6.1 版本之前，getLargestBatchIndexWhenPossible() 函数没有 return 语句，当 entry 为 -1 时，client 会对把相应的 MessageData 设置为当前位置的值，并将该值发送到 client，当 client 尝试读取该 entry，会出现如下问题：

```
16:34:25.779 [pulsar-io-54-7:org.apache.bookkeeper.client.LedgerHandle@748] ERROR org.apache.bookkeeper.client.LedgerHandle - IncorrectParameterException on ledgerId:0 firstEntry:-1 lastEntry:-1
```

```
16:34:25.779 [pulsar-client-io-82-1:org.apache.pulsar.client.impl.ConsumerImpl@1986] INFO
org.apache.pulsar.client.impl.ConsumerImpl - [persistent://external-repl-prop/pulsar-function
-admin/assignment][c-use-fw-localhost-0-function-assignment-initialize-reader-
b21f7607c9] Successfully getLastMessageId 0:-1
16:34:25.779 [pulsar-client-io-82-1:org.apache.pulsar.client.impl.ClientCnx@602] WARN org.a
pache.pulsar.client.impl.ClientCnx - [id: 0xc78f4a0e, L:/127.0.0.1:55657 - R:localhost/127.0.0.1:
55615] Received error from server: Failed to get batch size for entry org.apache.bookkeeper.m
ledger.ManagedLedgerException: Incorrect parameter input
16:34:25.779 [pulsar-client-io-82-1:org.apache.pulsar.client.impl.ClientCnx@612] WARN org.a
pache.pulsar.client.impl.ClientCnx - [id: 0xc78f4a0e, L:/127.0.0.1:55657 - R:localhost/127.0.0.1:
55615] Received unknown request id from server: 10
```

PR-7495 在代码中增加了 return 语句，GetLastEntry() 会读取最后一条 entry，而不是 -1。

更多详情查看 PR-7495：<https://github.com/apache/pul...>

ZooKeeper 相关改进

使用主机名进行 Bookie 机架感知映射

PR-5607 中添加了 useHostName() 和 return false。这意味着机架式策略会尝试将 Bookie 主机名解析为 IP 地址，然后使用该 IP 地址来确定 Bookie 属于哪个机架。

这会导致如下两个问题：

- IP 地址与在 / bookies z-节点中记录的主机名不匹配；
- 如果在解析 bookie 主机名时发生错误（例如：瞬态 DNS 错误），会触发 NPE 异常；对 BookKeeper 客户端来说，该 bookie 在集群中一直不可用。

例如，在下面代码中的第 77 行会抛出 NPE，因为 getAddress() 给出了一个 null，而该地址没有解析：

```
74     if (dnsResolver.useHostName()) {
75         names.add(addr.getHostName());
76     } else {
77         names.add(addr.getAddress().getHostAddress());
78     }
```

默认情况下，DnsResolver.useHostName() 返回 true。

更多详情参考 PR-7361：<https://github.com/apache/pul...>

Java Client 相关改进

修复了无法重命名 Athenz 身份验证中使用的 HTTP header 的问题

Athenz 的身份验证插件允许用户更改 HTTP header 的名称，并通过 `roleHeader` 参数将身份验证令牌发送到代理服务器。更改 HTTP header 名称会保留 "AuthenticationAthenz" 侧的 "roleHeader" 参数的值，并将其直接用作标头名称。

更多详情参考 PR-7311：<https://github.com/apache/pul...>

修复多次回收 batch ack 的集合

多次回收 batch ack 的根本原因是批量 Ack 刷新和累积确认中存在竞争条件。因此，为该 ackset 添加回收状态检查，避免多次回收 batch ack。

更多详情参考 PR-7409：<https://github.com/apache/pul...>

添加支持 OAuth2 身份验证的客户端

Pulsar 支持使用 OAuth 2.0 访问令牌验证客户端身份。可以使用令牌来标识 Pulsar 客户端，并将令牌关联到允许执行某些操作（例如：发布到主题或从主题消费）的某些 "principal"（或 "role"）。

该模块直接支持 OAuth 2.0 的 Pulsar 客户端身份验证插件。客户端与 OAuth 2.0 服务器进行通信后，将从 OAuth 2.0 服务器获取 "访问令牌"，并将该 "访问令牌" 传递给 Pulsar broker 进行身份验证。

因此，代理方仍然可以使用 "org.apache.pulsar.broker.authentication.AuthenticationProviderToken"，用户也可以添加自己的 AuthenticationProvider 来使用此模块。

更多详情参考 PR-7420：<https://github.com/apache/pul...>

在 consumer 关闭之后，不再订阅这个 topic

当 consumer 重新连接到 broker 时，将竞争条件固定在 consumer 中。

在 consumer 重新连接到代理时会发生竞争条件，消费者重新连接到代理时连接设置为 null。如果此时关闭 consumer，客户端不再向代理发送关闭 consumer 的命令。因此，如果 consumer 重新连接到 broker，consumer 将再次发送订阅命令。

在 2.6.1 版本中，当 consumer 的连接打开时，consumer

会添加状态检查。如果使用者状态为关闭或正在关闭，则无需发送订阅命令。

更多详情参考 PR-7589：<https://github.com/apache/pul...>

OAuth2 身份验证插件使用 AsyncHttpClient

在之前的版本中，OAuth2 客户端 auth 插件使用 Apache HTTP 客户端库发出请求，Apache HTTP 客户端仅用于主机名验证。如 PR-7612 所述，为了摆脱对 Apache HTTP 客户端库的依赖，在 2.6.1 版本中使用 AsyncHttpClient。AsyncHttpClient 在客户端和 broker 中的其他地方都有使用。

更多详情参考 PR-7615：<https://github.com/apache/pul...>

CPP Client 相关改进

在 CPP 客户端中支持 OAuth2 的认证方式

Pulsar 支持使用 OAuth 2.0 访问令牌对客户端进行身份验证。可以使用令牌来标识 Pulsar 客户端，并将其与允许执行某些操作（例如：发布到主题或从主题消费）的某些“principal”（或“role”）关联。

在 2.6.1 版本中，允许用户在 CPP 客户端中使用 OAuth2 的认证方式。

更多详情参考 PR-7467：<https://github.com/apache/pul...>

修复在关闭 callback 中 partition 索引的错误

在分区生产者/消费者中关闭 callback 时，分区索引始终为 0。我们需要将 ProducerImpl / ConsumerImpl 的内部 partition 索引字段传递给 PartitionedProducerImpl / PartitionedConsumerImpl 的 close 回调。

更多详情参考 PR-7282：<https://github.com/apache/pul...>

修复了 C++ 客户端中计时器的竞争状况导致的段崩溃

在 2.6.1 版本之前，竞争条件下会发生段崩溃：

- 关闭操作，称为“keepAliveTimer_.reset()”；
- 同时，在 startConsumerStatsTimer 和 handleKeepAliveTimeout 方法中访问计时器。

在 2.6.1 版本中，我们修复了此问题，竞争条件下不再发生段崩溃。

更多详情参考 PR-7572：<https://github.com/apache/pul...>

支持从文件读取凭据

支持从文件读取凭据，使其与 Java 客户端保持一致。

更多详情参考 PR-7606：<https://github.com/apache/pul...>

修复在连接出错时多 topic consumer 的段错误

当创建 consumer 出现错误时，多主题 consumer 将触发段错误。这是使用 null 回调关闭部分使用者的调用所致。

在 2.6.1 版本中，我们修复了此问题。

更多详情参考 PR-7588：<https://github.com/apache/pul...>

Functions 相关改进

使用标准主机名作为 worker 的默认值

Java 8 和 Java 11 获取主机名的方法不同。在 Java 8 中，使用 `InetAddress.getLocalHost()` 参数，`getHostName()` 返回完全限定的主机名。在 Java 11 中，则是返回简单主机名。使用 `getCanonicalHostName()` 参数后，在 Java 8 和 Java 11 中都能返回完全限定的主机名。

更多详情参考 PR-7360

<https://github.com/apache/pul...>

修复 2.6.0 引入的向后兼容问题

PR-5985 破坏了向后兼容性。如果分开运行 Function Worker 与 Broker，Function Worker 和 broker 从 2.5 版本单独更新到 2.6 版本时会发生以下错误：

```
java.lang.NullPointerException: null
  at java.net.URI$Parser.parse(URI.java:3104) ~[?:?]
  at java.net.URI.<init>(URI.java:600) ~[?:?]
  at java.net.URI.create(URI.java:881) ~[?:?]
  at org.apache.pulsar.functions.worker.WorkerUtils.initializeDlogNamespace(WorkerUtils.java:160) ~[org.apache.pulsar-pulsar-functions-worker-2.7.0-SNAPSHOT.jar:2.7.0-SNAPSHOT]
  at org.apache.pulsar.functions.worker.Worker.initialize(Worker.java:155) ~[org.apache.pulsar-pulsar-functions-worker-2.7.0-SNAPSHOT.jar:2.7.0-SNAPSHOT]
  at org.apache.pulsar.functions.worker.Worker.start(Worker.java:69) ~[org.apache.pulsar-pulsar-functions-worker-2.7.0-SNAPSHOT.jar:2.7.0-SNAPSHOT]
  at org.apache.pulsar.functions.worker.FunctionWorkerStarter.main(FunctionWorkerStarter.java:67) [org.apache.pulsar-pulsar-functions-worker-2.7.0-SNAPSHOT.jar:2.7.0-SNAPSHOT]
```

错误原因：2.5 版本中 broker 会对包含 bookkeeperMetadataServiceUri 字段的请求做出响应，管理客户端将返回该字段为 null，从而导致 NPE。

在 2.6.1 版本中，当初初始化 function worker 时，对 BookkeeperMetadataServiceUri 的 value 进行检查，判断其是否为 null。

更多详情参考 PR-7528：<https://github.com/apache/pul...>

Pulsar Perf 相关改进

在 pulsar-perf 的 producer/consumer/reader 中支持 tlsAllowInsecureConnection

在命令行工具 pulsar-perf 中支持 tlsAllowInsecureConnection 配置，以支持对不安全的 TLS 连接的集群进行 producer/consumer/reader 的性能测试。

更多详情参考 PR-7300：<https://github.com/apache/pul...>

参考信息

- Apache Pulsar 2.6.1 官网下载地址：<https://pulsar.apache.org/en/...>
- 更多关于 Apache Pulsar 2.6.1 的信息，可以参考下方：<https://pulsar.apache.org/rel...>
- Pulsar 2.6.1 PR 列表：<https://github.com/apache/pul...>

本文原文：[Apache Pulsar 2.6.1 版本正式发布：2.6.0 功能增强版，新增 OAuth2 支持](#)

本博客文章除特别声明，全部都是原创！
转载本文请加上：转载自过往记忆 (<https://www.iteblog.com/>)
本文链接: **【】** ()