Red Hat AMQ 2021.Q3

RHEL 上的 AMQ Streams 1.8 发行注记

用于 Red Hat Enterprise Linux 上的 AMQ Streams

Last Updated: 2023-02-23

Red Hat AMQ 2021.Q3 RHEL 上的 AMQ Streams 1.8 发行注记

用于 Red Hat Enterprise Linux 上的 AMQ Streams

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

法律通告

摘要

这些发行注记包含 AMQ Streams 1.8 发行版中包含的新功能、增强、修复和问题的最新信息。

目录

使开源包含更多

红帽承诺替换我们的代码、文档和网页属性中存在问题的语言。我们从这四个术语开始： master、slave、blacklist 和 whitelist。这些更改将在即将发行的几个发行本中逐渐实施。如需了解更多详细信息，请参阅 CTO Chris Wright 信息。

第 1 章 功能

本发行版本中添加的功能，以及之前 AMQ Streams 版本中没有的功能，如下所述。

注意

要查看此版本中已解决的所有增强和错误，请查看 AMQ Streams Jira 项目 。

1.1. KAFKA 2.8.0 支持

AMQ Streams 现在支持 Apache Kafka 版本 2.8.0。

AMQ Streams 使用 Kafka 2.8.0。仅支持由红帽构建的 Kafka 发行版本。

有关升级说明，请参阅 AMQ Streams 和 Kafka 升级 。

如需更多信息，请参阅 Kafka 2.7.0 和 Kafka 2.8.0 发行注记。

注意

仅支持将 Kafka 2.7.x 升级到 AMQ Streams 1.8。

有关支持版本的更多信息，请参阅红帽知识库文章 Red Hat AMQ 7 组件详情页 。

Kafka 2.8.0 需要 ZooKeeper 版本 3.5.9。因此，您需要在从 AMQ Streams 1.7 升级到 AMQ Streams 1.8 时升级 ZooKeeper，如升级文档所述。

警告

Kafka 2.8.0 提供了对 *自我管理模式* 的早期访问，其中 Kafka 使用 Raft 协议在没有 ZooKeeper 的情况下运行。**请注意，AMQ Streams 不支持自助管理模式。**

第 2 章 增强

下面概述了本发行版本中添加的增强功能。

2.1. KAFKA 2.8.0 增强

有关 Kafka 2.8.0 引入的增强概述，请参阅 Kafka 2.8.0 发行注记。

2.2. OAUTH 2.0 身份验证增强

配置使用者和范围

现在，您可以配置 **oauth.audience** 和 **oauth.scope** 属性，并在获取令牌时将其值作为参数传递。这两个属性都在 OAuth 2.0 身份验证侦听器配置中配置。

在以下情况下使用这些属性：

- 获取用于内部代理身份验证的访问令牌时
- 在基于 PLAIN 客户端身份验证的 OAuth 2.0 客户端名称中，使用 **clientId** 和 **secret**

这些属性会影响客户端是否可以获取令牌和令牌的内容。它们不会影响侦听器实施的令牌验证规则。

oauth.audience 和 oauth.scope 属性的配置示例

```
listener.name.client.oauthbearer.sasl.jaas.config=org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required \
  # ...
  oauth.token.endpoint.uri="https://AUTH-SERVER-ADDRESS/auth/realms/REALM-NAME/protocol/openid-connect/token" \
  oauth.scope=""SCOPE"" \
  oauth.audience="AUDIENCE" \
  oauth.check.audience="true" \
  # ...
```

您的授权服务器可能会在 JWT 访问令牌中提供 **提示（严重）声明。当通过设置 oauth.check.audience="true"** 启用用户检查时，Kafka 代理将拒绝在其 **aud** 声明中不包含代理 **客户端 Id** 的令牌。默认情况下禁用受众检查。

请参阅为 Kafka 代理配置 OAuth 2.0 支持 https://access.redhat.com/documentation/en-us/red_hat_amq/2021.q3/html-single/using_amq_streams_on_rhel/index#proc-oauth-broker-config-str

OAuth 2.0 over PLAIN 不需要令牌端点

使用 OAuth 2.0 的 "client ID 和 secret" 方法通过 PLAIN 身份验证时，不再需要 **oauth.token.endpoint.uri** 参数。

指定了令牌端点 URI 的 OAuth 2.0 over PLAIN 侦听器配置示例

```
listener.name.client.plain.sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule required \
  oauth.valid.issuer.uri="https://__AUTH-SERVER-ADDRESS__" \
  oauth.jwks.endpoint.uri="https://__AUTH-SERVER-ADDRESS__/jwks" \
```

```
  oauth.username.claim="preferred_username" \
  oauth.token.endpoint.uri="http://__AUTH_SERVER__/auth/realms/__REALM__/protocol/openid-
connect/token" ;
```

如果没有指定 **oauth.token.endpoint.uri**，侦听器将处理：

- **用户名** 参数作为帐户名称
- 作为原始访问令牌 **的密码** 参数，传递给授权服务器进行验证（与 OAUTHBEARER 身份验证的作用相同）

OAuth 2.0 与 PLAIN 身份验证相比的"长生命访问令牌"方法的行为没有改变。在使用此方法时，不需要 **oauth.token.endpoint.uri**。

请参阅 OAuth 2.0 Kafka 代理配置

第 3 章 技术预览

重要

技术预览功能不被红帽产品服务级别协议(SLA)支持，且可能无法完成。因此，红帽不推荐在生产环境中实施任何技术预览功能。此技术预览功能为您提供对即将推出的产品创新的早期访问，允许您在开发过程中测试并提供反馈。如需有关支持范围的更多信息，请参阅技术预览功能支持范围。

3.1. KAFKA 静态配额插件配置

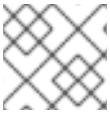
使用 *Kafka Static Quota* 插件，为 Kafka 集群中的代理设置吞吐量和存储限制。您可以设置字节阈值和存储配额，对与代理交互的客户端设定限制。

Kafka 静态配额插件配置示例

```
client.quota.callback.class= io.strimzi.kafka.quotas.StaticQuotaCallback
client.quota.callback.static.produce= 1000000
client.quota.callback.static.fetch= 1000000
client.quota.callback.static.storage.soft= 400000000000
client.quota.callback.static.storage.hard= 500000000000
client.quota.callback.static.storage.check-interval= 5
```

请参阅使用 Kafka 静态配额插件设置代理限制

3.2. 用于集群重新平衡的精简控制



注意

固态控制仍为技术预览，但有一些新的 增强功能。

您可以安装 Cruise Control，并使用它来在 CPU、磁盘、网络负载等中使用 *优化的 target SAS-* SAS 定义约束来重新平衡 Kafka 集群。在均衡 Kafka 集群中，工作负载更加均匀地分布在代理 pod 中。

cruise Control 有助于减少运行高效、均衡 Kafka 集群所需的时间和工作量。

可以从 客户门户网站 下载 Cruise Control 的压缩分发包。要安装 Cruise Control，您需要将每个 Kafka 代理配置为使用提供的 Metrics Reporter。然后，您设置 Cruise Control 属性，包括优化目标，并使用提供的脚本启动 Cruise 控制。

Cruise Control 服务器托管在整个 Kafka 集群的单个机器上。

当 Cruise Control 运行时，您可以使用 REST API:

- 从多个优化目标生成 *空运行* 优化调整
- 启动优化建议以重新平衡 Kafka 集群

目前不支持其他 Cruise 控制功能，包括异常检测、通知、写入目标以及更改主题复制因素。

请参阅 用于集群重新平衡的 Cruise Control

3.2.1. 技术预览的改进

Straliation Control 版本 2.5.59 显著提升了性能，包括加快 10% 的性能计算。

最新版本的压缩发行版可从红帽客户门户下载。

请参阅 客户门户

第 4 章 已弃用的功能

本发行版本中弃用的功能，以及之前的 AMQ Streams 版本中所支持的功能如下所示。

4.1. 弃用和删除 KAFKA 功能

本节提前通知 Apache Kafka 项目中的重要弃用和删除。

4.1.1. 计划在 Kafka 版本 3.0 中删除

Kafka 版本 3.0 将随 AMQ Streams 下一个主发行版本一起提供。

下表显示了在 Kafka 2.x 或更早版本中弃用且将在 Kafka 3.0 **中删除** 的方法和组件。这份清单并非详尽。

表 4.1. 弃用了将在 Kafka 3.0 中删除的 API 方法和组件

API 或组件	问题链接	描述
管理 API	KAFKA-12581	删除已弃用的 Admin.electPreferredLeaders
管理 API	KAFKA-6987	使用 CompletableFuture 重新实施 KafkaFuture（弃用 KafkaFuture.Function）
管理客户端	KAFKA-12577	删除已弃用的 **ConfigEntry** 构造器
所有客户端	KAFKA-12579	从客户端的 3.0 中删除各种弃用方法
所有客户端	KAFKA-12600	删除客户端配置客户端. **dns.lookup**的已弃用配置值 **默认值**
所有客户端	KAFKA-12578	删除已弃用的安全类/methods
broker	KAFKA-12591	删除已弃用的 **quota.producer.default** 和 **quota.consumer.default** 配置
broker	KAFKA-12592	删除已弃用的 LogConfig.Compact
broker	KAFKA-12590	Remove deprecated SimpleAclAuthorizer
broker	KAFKA-5905	删除 PrincipalBuilder 和 DefaultPrincipalBuilder

API 或组件	问题链接	描述
common	KAFKA-12573	删除了 deprecated **Metric#value**
使用者 API	KAFKA-12637	删除已弃用的 PartitionAssignor 接口
连接 API	KAFKA-12482	删除已弃用的 rest.host.name 和 rest.port Connect worker 配置
连接 API	KAFKA-12945	删除 3.0 中的端口、host.name 和相关配置
连接 API	KAFKA-12717	删除内部转换器配置属性
Streams API	KAFKA-12574	弃用 eos-alpha
Streams API	KAFKA-12808	删除 StreamsMetrics 下已弃用的方法
Streams API	KAFKA-7606	从 StreamsResetter 中删除已弃用的选项
Streams API	KAFKA-12796	在 stream **-scala 下删除**已弃用的类
Streams API	KAFKA-12419	删除 3.0 中已弃用的 Kafka Streams API
Streams API	KAFKA-10434	删除 WindowStore 上已弃用的方法
Streams API	KAFKA-12449	删除已弃用的 WindowStore#put
Streams API	KAFKA-12813	删除 ProcessorContext 中已弃用的调度方法
Streams API	KAFKA-12809	删除 Stores 中已弃用的方法
Streams API	KAFKA-12814	删除已弃用的方法 StreamsConfig#getConsumerConfig
Streams API	KAFKA-12313	弃用 default.windowed.serde.inner.class 配置

API 或组件	问题链接	描述
Streams API	KAFKA-8372	删除已弃用的 RocksDB#compactRange API
Streams API	KAFKA-12584	删除已弃用的 **Sum** 和 **Total** 类
Streams API	KAFKA-12683	删除已弃用的"UsePreviousTimeOnInvalidTimeStamp"
Streams API	KAFKA-12810	Remove deprecated TopologyDescription.Source#topics
Streams API	KAFKA-12630	Remove deprecated KafkaClientSupplier#getAdminClient
Streams API	KAFKA-10046	弃用的 PartitionGrouper 配置会被忽略
Streams API	KAFKA-12633	Remove deprecated "TopologyTestDriver#pipeInput / readOutput"
Streams API	KAFKA-12441	删除已弃用的方法 StreamsBuilder#addGlobalStore
Streams API	KAFKA-12452	为 ProcessorContext#forward 删除已弃用的过载
Streams API	KAFKA-12450	从 ReadOnlyWindowStore 中删除已弃用的方法
Streams API	KAFKA-12880	删除 3.0 中已弃用的 Count 和 SampledTotal
Streams API	KAFKA-12451	删除 WindowStore 中基于长期读取操作的弃用注解
Streams API	KAFKA-12568	删除已弃用的 "KStream#groupBy/join", "Joined#named" overloads
Streams API	KAFKA-12849	将 TaskMetadata 迁移到与内部实现的接口

API 或组件	问题链接	描述
Streams API	KAFKA-7785	删除 PartitionGrouper 接口及其配置，并将 DefaultPartitionGrouper 移到内部软件包
Streams API	KAFKA-7106	从窗口定义中删除片段/segmentInterval
Streams API	KAFKA-8897	增加 RocksDB 版本
Streams API	KAFKA-12909	允许用户选择-inbeious left/outer stream-stream 加入改进
工具	KAFKA-8405	删除已弃用的 **kafka-preferred-replica-election** 命令
工具	KAFKA-12588	删除 shell 命令中已弃用的 --zookeeper

4.1.2. mirror Maker 1.0 计划删除 Kafka 版本 4.0

Kafka 版本 4.0 将在 AMQ Streams 未来的主发行版本中提供。

下表显示了将在 Kafka 3.0 中弃用并在 Kafka 4.0 **中删除** 的功能。

表 4.2. 将在 Kafka 3.0 中弃用并在 Kafka 4.0 中删除的组件

组件	发布链接	概述
镜像 Maker 1.0	KAFKA-12436	deprecate MirrorMaker v1

第 5 章 修复的问题

下表显示了在 RHEL 上的 AMQ Streams 1.8 中修复的问题。有关在 Kafka 2.8.0 中修复的问题的详情，请参考 Kafka 2.8.0 发行注记。

表 5.1. 修复的问题

问题号	描述
ENTMQST-2453	**kafka-exporter** pod 不会因任何原因重启。
ENTMQST-2459	运行 Kafka Exporter 会导致 CPU 使用率高。
ENTMQST-2511	微调健康检查，以在滚动更新期间停止 Kafka Exporter 重启。
ENTMQST-1529	如果有大型文件，文件源连接器将停止。

表 5.2. 修复了常见漏洞和风险(CVE)

问题号	标题	描述
ENTMQST-3023	CVE-2021-34428 jetty-server: jetty: SessionListener 可以防止会话被破坏注销无效。	jetty-server 中发现了一个缺陷，如果从 SessionListener#sessionDestroyed（）方法抛出异常，则会话 ID 不会在会话 ID 管理器中无效。在具有集群会话和多个上下文的部署中，可能会导致会话无效，并且共享计算机应用被保留登录。来自此漏洞的最大威胁是数据保密性和完整性。
ENTMQST-2980	CVE-2021-28169 jetty-server: jetty: 对 ConcatServlet 和 WelcomeFilter 的请求可以访问 WEB-INF 目录中受保护的资源。	-
ENTMQST-2711	CVE-2021-21409 netty：通过内容长度标头请求调用.	Netty 中发现了一个漏洞。如果请求使用单个 Http2HeaderFrame 并将 endstream 设置为 true，则存在一个问题，即 content-length 标头不会被正确验证。如果请求代理到远程同级并转换为 HTTP/1.1，则会导致请求生效。这种漏洞的最大威胁是完整性。

问题号	标题	描述
ENTMQST-2663	CVE-2021-27568 json-smart：未发生异常可能会导致崩溃或信息泄露。	json-smart 中发现了一个漏洞。当异常从函数抛出但未被捕获时，使用库的程序可能会崩溃或公开敏感信息。来自此漏洞的最大威胁是数据机密和系统可用性。 在 OpenShift Container Platform(OCP)中，组成 OCP Metering 堆栈的 Hive/Presto/Hadoop 组件会提供存在安全漏洞的 json-smart 软件包版本。自 OCP 4.6 发布以来，Metering 产品已弃用 [1]，因此受影响的组件被标记为 wintfix。以后可能会解决这个问题。 [1] https://docs.openshift.com/container-platform/4.6/release_notes/ocp-4-6-release-notes.html#ocp-4-6-metering-operator-deprecated
ENTMQST-2647	CVE-2021-21295 netty：在 HTTP/2 中因为缺少验证而可能的请求。	在 4.1.60.Final 以前的 Netty(io.netty:netty-codec-http2)中，存在一个启用请求的漏洞。如果原始 HTTP/2 请求中存在 Content-Length 标头，则该字段不会因为已传播而被 **Http2MultiplexHandler** 验证。只要请求没有作为 HTTP/1.1 代理通过，这都是正常的。如果请求以 HTTP/2 流的形式传入，则通过**Http 2StreamFrameToHttpObject Codec 转换为 HTTP/1.1 域对象（ HttpRequest、Http Content 等）**，然后发送至子频道的管道并通过远程 peer(HTTP/1.1)进行代理，这可能会导致请求退出。
ENTMQST-2617	CVE-2021-21290 netty：通过本地系统临时目录披露信息。	在 Netty 中，类似 Unix 的系统中存在一个涉及不安全临时文件的漏洞。当使用 netty 的多部分解码器时，如果启用了在磁盘上临时存储上传，则可以通过本地系统临时目录进行本地信息披露。在与 unix 类似的系统上，临时目录在所有用户之间共享。因此，使用未明确设置文件/目录权限的 API 写入该目录可能会导致信息泄露。

问题号	标题	描述
ENTMQST-2613	CVE-2020-13949 libthrift：处理不受信任的载荷时潜在的 DoS。	libthrift 中发现了一个漏洞。使用 Thrift 的应用不会在接收消息时显示错误，声明大小大于载荷的容器。这导致恶意的 RPC 客户端能够发送短消息，进而导致大量内存分配，从而可能拒绝服务。此漏洞的最大威胁在于系统可用性。
ENTMQST-1934	CVE-2020-9488 log4j：对 SMTP 附加程序中主机不匹配的证书验证不正确。	-
ENTMQST-2910	CVE-2021-28163 jetty-server: jetty: Symlink 目录公开 webapp 目录内容。	如果 **${jetty.base}** 目录或 **${jetty.base}/webapps** 目录是符号链接，则 **${jetty.base}/webapps** 目录的内容可以部署为静态 Web 应用，公开要下载的目录内容。来自此漏洞的最大威胁是数据机密性。
ENTMQST-2909	CVE-2021-28164 jetty-server：jetty：嵌入式路径可以访问 WEB-INF。	在 Jetty 中，默认合规模式允许带有包含 **%2e 或 %2e %2e 片段** 的 URI 的请求，以访问 WEB-INF 目录中受保护的资源。攻击者可以利用此漏洞来披露有关 Web 应用实施的敏感信息。
ENTMQST-2908	CVE-2021-28165 jetty-server: jetty: 在接收无效大型 TLS 帧时资源耗尽。	当在 Jetty 中使用 SSL/TLS 时（HTTP/1.1、HTTP/2 或 WebSocket），服务器可能会收到一个无效的大（大于 17408） TLS 帧，该帧被错误处理，从而导致 CPU 资源利用率高。此漏洞的最大威胁在于服务可用性。
ENTMQST-2867	CVE-2021-29425 commons-io：apache-commons-io：Apache Commons IO 2.2 到 2.6 中的有限路径遍历.	-
ENTMQST-2821	CVE-2021-28168 jersey-common：jersey：本地信息通过系统临时目录披露。	-

第 6 章 已知问题

本节列出了 AMQ Streams 1.8 的已知问题。

6.1. LOG4J 的 SMTP 附加程序

AMQ Streams 附带一个潜在的存在安全漏洞的 log4j 版本(**log4j-1.2.17.redhat-3**)。漏洞在于 SMTP 附加程序功能，其默认配置中 AMQ Streams 不使用该功能。

表 6.1. CVE 问题

问题号	描述
ENTMQST-1934	CVE-2020-9488 log4j：SMTP 附加程序 [amq-st-1] 中存在主机不匹配的证书验证不正确。

临时解决方案

如果您使用 SMTP 附加程序，请确保 **mail.smtp.ssl.checkserveridentity** 设置为 **true**。

第 7 章 支持的集成产品

AMQ Streams 1.8 支持与以下红帽产品集成：

Red Hat Single Sign-On 7.4 及更新的版本

提供 OAuth 2.0 身份验证和 OAuth 2.0 授权。

有关这些产品可引入到您的 AMQ Streams 部署的功能信息，请参阅 AMQ Streams 1.8 文档。

其它资源

- Red Hat Single Sign-On 支持的配置

第 8 章 重要链接

- Red Hat AMQ 7 支持的配置
- Red Hat AMQ 7 组件详情

2021-12-18 13:43:08 +1000 修订