



Red Hat Streams for Apache Kafka 2.5

RHEL 上的 AMQ Streams 2.5 发行注记

Red Hat Enterprise Linux 中的 AMQ Streams 发行版本的主要新功能及变化信息

Red Hat Streams for Apache Kafka 2.5 RHEL 上的 AMQ Streams 2.5 发行 注记

Red Hat Enterprise Linux 中的 AMQ Streams 发行版本的主要新功能及变化信息

法律通告

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

摘要

本发行注记概述了 AMQ Streams 2.5 发行版本中引入的新功能、功能增强和修复。

目录

第 1 章 名称通知会改为 APACHE KAFKA 的 STREAMS	3
第 2 章 AMQ STREAMS 2.5 LONG TERM SUPPORT	4
第 3 章 功能	5
3.1. AMQ STREAMS 2.5.X (LONG TERM SUPPORT)	5
3.2. KAFKA 3.5.X 支持	5
3.3. OPENTELEMETRY 用于分布式追踪	5
第 4 章 功能增强	6
4.1. KAFKA 3.5.X 的改进	6
4.2. 对 KRAFT 模式的 OAUTH 2.0 支持	6
4.3. 用于授予管理的 OAUTH 2.0 配置属性	6
4.4. 在提取用户名时 OAUTH 2.0 支持 JSONPATH 查询	6
4.5. 指标和 OPENAPI 的 KAFKA BRIDGE 增强	7
第 5 章 技术预览	8
5.1. KRAFT 模式	8
5.2. KAFKA 静态配额插件配置	8
第 6 章 已弃用的功能	9
6.1. RHEL 7 在 AMQ STREAMS 2.5.X (LTS)中弃用	9
6.2. AMQ STREAMS 2.4.0 删除了对 JAVA 8 的支持	9
6.3. OPENTRACING	9
6.4. KAFKA MIRRORMAKER 2 身份复制策略	9
6.5. KAFKA MIRRORMAKER 1	9
第 7 章 修复的问题	10
7.1. 修复了 AMQ STREAMS 2.5.2 的问题	10
7.2. 修复了 AMQ STREAMS 2.5.1 的问题	10
7.3. 修复了 AMQ STREAMS 2.5.0 的问题	10
第 8 章 已知问题	12
8.1. 以 FIPS 模式运行时的 JMX 身份验证	12
第 9 章 支持的配置	13
9.1. 支持的平台	13
9.2. 支持的 APACHE KAFKA 生态系统	13
9.3. 其他支持的功能	13
9.4. 存储要求	14
第 10 章 组件详情	15
第 11 章 支持的与红帽产品集成	17
11.1. 红帽单点登录	17

第 1 章 名称通知会改为 APACHE KAFKA 的 STREAMS

AMQ Streams 被重命名为 **Apache Kafka 的流**，作为品牌工作的一部分。这个变化旨在提高红帽产品用于 Apache Kafka 的客户了解。在这个转换期间，您可能会遇到对旧名称 AMQ Streams 的引用。我们积极努力更新文档、资源和媒体，以反映新的名称。

第 2 章 AMQ STREAMS 2.5 LONG TERM SUPPORT

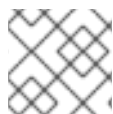
AMQ Streams 2.5 是 AMQ Streams 的 Long Term Support (LTS) 产品。

有关 LTS 术语和日期的详情，请查看 [AMQ Streams LTS 支持策略](#)。

第 3 章 功能

AMQ Streams 2.5 引入了本节所述的功能。

RHEL 上的 AMQ Streams 2.5 基于 Apache Kafka 3.5.0。



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](#)。

3.1. AMQ STREAMS 2.5.X (LONG TERM SUPPORT)

AMQ Streams 2.5.x 是 AMQ Streams 的 Long Term Support (LTS) 产品。

最新的补丁版本是 AMQ Streams 2.5.2。AMQ Streams 二进制文件已改为 2.5.2 版本。虽然支持的 Kafka 版本被列为 3.5.0，但它包含了 Kafka 3.5.2 的更新和改进。

有关 LTS 术语和日期的详情，请查看 [AMQ Streams LTS 支持策略](#)。

3.2. KAFKA 3.5.X 支持

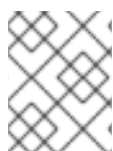
AMQ Streams 支持并使用 Apache Kafka 版本 3.5.0。Kafka 3.5.2 的更新会包括在 2.5.2 补丁版本中。仅支持由红帽构建的 Kafka 发行版本。

有关升级说明，请参阅 [AMQ Streams](#) 和 [Kafka 升级](#)。

如需了解更多信息，请参阅 [Kafka 3.5.0](#)、[Kafka 3.5.1](#) 和 [Kafka 3.5.2](#) 发行注记。

Kafka 3.4.x 仅支持升级到 AMQ Streams 2.5。

Kafka 3.5.x 使用 ZooKeeper 版本 3.6.4，它是 Kafka 3.4.x 的不同版本。建议您执行滚动更新以使用新的二进制文件。



注意

Kafka 3.5.x 提供对 KRaft 模式的访问，其中 Kafka 在使用 Raft 协议的情况下运行没有 ZooKeeper。KRaft 模式作为 [技术预览](#) 提供。

3.3. OPENTELEMETRY 用于分布式追踪

OpenTelemetry 用于分布式追踪已变为 GA。您可以在指定的追踪系统中使用 OpenTelemetry。OpenTelemetry 已经替换了 OpenTracing 以进行分布式追踪。[对 OpenTracing 的支持已弃用](#)。

默认情况下，OpenTelemetry 使用 OTLP (OpenTelemetry 协议) exporter 进行追踪。带有 OpenTelemetry 的 AMQ Streams 与 Jaeger exporter 一起使用，但您可以指定 OpenTelemetry 支持的其他追踪系统。AMQ Streams 计划默认迁移到使用 OTLP exporter 的 OpenTelemetry，它支持 Jaeger exporter。

请参阅 [分布式追踪简介](#)。

第 4 章 功能增强

AMQ Streams 2.5 添加了很多改进。

4.1. KAFKA 3.5.X 的改进

AMQ Streams 2.5.x 发行版本支持 Kafka 3.5.0。升级到 2.5.2 补丁版本包含来自 Kafka 3.5.2 的更新和改进。

有关 Kafka 3.5.x 中引入的增强功能概述，请参阅 [Kafka 3.5.0](#)、[Kafka 3.5.1](#) 和 [Kafka 3.5.2](#) 发行注记。

4.2. 对 KRAFT 模式的 OAUTH 2.0 支持

由 AMQ Streams 提供的 Red Hat Single Sign-On 授权器的 **KeycloakRBACAuthorizer** 已被 **KeycloakAuthorizer** 替代。新的授权器与 ZooKeeper 集群管理或 KRaft 模式使用 AMQ Streams 兼容。与前面的授权器一样，您可以使用 Red Hat Single Sign-On 提供的 Authorization Services 的 Red Hat Single Sign-On REST 端点，您可以在 Kafka 代理上配置 **KeycloakAuthorizer**。使用带有 ZooKeeper 集群管理的 AMQ Streams 时仍可使用 **KeycloakRBACAuthorizer**，但您应该迁移到新的授权器。

4.3. 用于授予管理的 OAUTH 2.0 配置属性

现在，您可以使用其他配置从授权服务器管理 OAuth 2.0 授予。

如果您使用 Red Hat Single Sign-On 进行 OAuth 2.0 授权，您可以在 Kafka 代理的授权配置中添加以下属性：

- **strimzi.authorization.grants.max.idle.time.seconds** 指定缓存中闲置授权可以被驱除的时间（以秒为单位）。默认值为 300。
- **strimzi.authorization.grants.gc.period.seconds** 指定在始终运行从缓存中清理过时的作业之间的时间（以秒为单位）。默认值为 300。
- **strimzi.authorization.reuse.grants** 控制是否为新会话获取最新的授权。禁用后，从 Red Hat Single Sign-On 检索并缓存该用户的权限。默认值为 **true**。

使用 OAuth 2.0 授权的 Kafka 配置

```
strimzi.authorization.grants.max.idle.time.seconds="300"  
strimzi.authorization.grants.gc.period.seconds="300"  
strimzi.authorization.reuse.grants="false"
```

请参阅[配置 OAuth 2.0 授权支持](#)。

4.4. 在提取用户名时 OAUTH 2.0 支持 JSONPATH 查询

要在 Kafka 集群中使用 OAuth 2.0 身份验证，您可以使用 OAUTH 身份验证机制指定监听程序配置。在配置侦听器属性时，现在可以使用 JsonPath 查询从正在使用的授权服务器中提取用户名。您可以使用 JsonPath 查询来为 **oauth.username.claim** 和 **oauth.fallback.username.claim** 属性在侦听器中指定 username extraction 选项。这可让您通过访问嵌套数据结构中的特定值从令牌中提取用户名。例如，您可能有一个包含在 JSON 令牌数据结构中的 *user info* 数据结构的用户名。

下例演示了如何在使用内省端点配置令牌验证时为属性指定 JsonPath 查询。

使用内省端点配置令牌验证

```
# ...
listener.name.client.oauthbearer.sasl.jaas.config=org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required ;
# ...
oauth.username.claim=["user.info"].['user.id'] \ ❶
oauth.fallback.username.claim=["client.info"].['client.id'] \ ❷
# ...
```

- ❶ 在令牌中包含实际用户名的令牌声明（或密钥）。用户名是用于识别用户的 *主体*。**userNameClaim** 值取决于使用的授权服务器。
- ❷ 授权服务器可能无法提供单个属性来识别常规用户和客户端。当客户端以自己的名称进行身份验证时，服务器可能会提供 *客户端 ID*。当用户使用用户名和密码进行身份验证时，若要获取刷新令牌或访问令牌，除了客户端 ID 外，服务器可能会提供一个 *username* 属性。如果主用户 ID 属性不可用，则使用这个 *fallback* 选项指定要使用的用户名声明(attribute)。

请参阅 [为 Kafka 代理配置 OAuth 2.0 支持](#)。

4.5. 指标和 OPENAPI 的 KAFKA BRIDGE 增强

Kafka Bridge 的最新版本包括以下更改：

- 从 HTTP 服务器相关的指标中删除 **remote** 和 **local** 标签，以防止时间序列示例增长。
- 消除对 **/metrics** 端点上的请求核算的 HTTP 服务器指标。
- 通过 OpenAPI 规格公开 **/metrics** 端点，为指标访问和管理提供标准化的接口。
- 修复 **OffsetRecordSentList** 组件模式，以返回记录偏移或错误。
- 修复 **ConsumerRecord** 组件模式，将键和值返回为对象，而不仅仅是(JSON)字符串。
- 更正 **/ready** 和 **/healthy** 端点返回的 HTTP 状态代码：
 - 将成功响应代码从 **200** 改为 **204**，指示成功响应中没有内容。
 - 在故障问题单的规格中添加 **500** 状态代码，指示错误响应中没有内容。

请参阅[使用 AMQ Streams Kafka Bridge](#)。

第 5 章 技术预览

AMQ Streams 2.5 中包含的技术预览功能。



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](#)。

5.1. KRAFT 模式

Apache Kafka 处于对 ZooKeeper 的需求。现在，您可以在没有 ZooKeeper 的情况下，以 KRaft (Kafka Raft metadata) 模式部署 Kafka 集群（技术预览）。

小心

此模式仅适用于开发和测试，且不得在生产环境中启用。

目前，AMQ Streams 中的 KRaft 模式有以下主要限制：

- 不支持从 ZooKeeper 移动到 KRaft 集群或其他方法。
- 不支持 Apache Kafka 版本的升级和降级。
- 不支持带有多个磁盘的 JBOD 存储。
- 许多配置选项仍在开发中。

请参阅[在 KRaft 模式中运行 Kafka](#)。

5.2. KAFKA 静态配额插件配置

使用 *Kafka Static Quota* 插件的技术预览，在 Kafka 集群中的代理上设置吞吐量和存储限制。您可以设置字节阈值和存储配额，以在与代理交互的客户端上放置限制。

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设置限制](#)。

第 6 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。

6.1. RHEL 7 在 AMQ STREAMS 2.5.X (LTS)中弃用

AMQ Streams 2.5.x 中弃用了对 RHEL 7 的支持。AMQ Streams 2.5.x (LTS)是支持 RHEL 7 的最后一个 LTS 版本。

6.2. AMQ STREAMS 2.4.0 删除了对 JAVA 8 的支持

Kafka 3.0.0 和 AMQ Streams 2.0 中已弃用对 Java 8 的支持。AMQ Streams 2.4.0 中删除了对 Java 8 的支持。这适用于所有 AMQ Streams 组件，包括客户端。

AMQ Streams 支持 Java 11 和 Java 17。在开发新应用程序时使用 Java 11 或 17。计划将当前使用 Java 8 的任何应用程序迁移到 Java 11 或 17。

如果您要继续使用 Java 8 一段时间，AMQ Streams 2.2 提供了 Long Term Support (LTS)。有关 LTS 术语和日期的详情，请查看 [AMQ Streams LTS 支持策略](#)。

6.3. OPENTRACING

对 OpenTracing 的支持已弃用。

Jaeger 客户端现已停用，OpenTracing 项目存档。因此，我们不能保证其对将来的 Kafka 版本的支持。我们基于 OpenTelemetry 项目推出新的追踪实施。

6.4. KAFKA MIRRORMAKER 2 身份复制策略

身份复制策略是 MirrorMaker 2 的一个功能，用于覆盖远程主题的自动重命名。主题不会使用源集群的名称添加名称，而是保留其原始名称。此设置对主动/被动备份和数据迁移场景特别有用。

要实现身份复制策略，您必须在 MirrorMaker 2 配置中指定复制策略类(**replication.policy.class**)。在以前的版本中，您可以指定 AMQ Streams **mirror-maker-2-extensions** 组件中包含的 **io.strimzi.kafka.connect.mirror.IdentityReplicationPolicy** 类。但是，这个组件现已弃用，并将在以后的版本中删除。因此，建议您更新您的实现以使用 Kafka 自己的复制策略类 (**org.apache.kafka.connect.mirror.IdentityReplicationPolicy**)。

请参阅[使用带有 MirrorMaker 2 的 AMQ Streams](#)。

6.5. KAFKA MIRRORMAKER 1

Kafka MirrorMaker 在两个或更多活跃 Kafka 集群之间复制数据，并在数据中心之间复制数据。Kafka MirrorMaker 1 在 Kafka 3.0.0 中已弃用，并将在 Kafka 4.0.0 中删除。MirrorMaker 2 将是唯一可用的版本。MirrorMaker 2 基于 Kafka Connect 框架，连接器管理集群之间的数据传输。

因此，在 AMQ Streams 中也弃用了 MirrorMaker 1。如果您使用 MirrorMaker 1（称为 AMQ Streams 文档中的 *MirrorMaker*），请将 MirrorMaker 2 与 **IdentityReplicationPolicy** 类搭配使用。MirrorMaker 2 将复制的主题重命名为目标集群。**IdentityReplicationPolicy** 配置会覆盖自动重命名。使用它生成与 MirrorMaker 1 相同的主动/被动单向复制。

请参阅[使用带有 MirrorMaker 2 的 AMQ Streams](#)。

第 7 章 修复的问题

以下小节列出了 AMQ Streams 2.5.x 中修复的问题。红帽建议您升级到最新的补丁版本。

AMQ Streams 2.5.x 发行版本支持 Kafka 3.5.0。有关 Kafka 3.5.0 中修复的问题的详情，请参考 [Kafka 3.5.0 发行注记](#)。

7.1. 修复了 AMQ STREAMS 2.5.2 的问题

AMQ Streams 2.5.2 (Long Term Support) 是最新的补丁版本。补丁版本包含 Kafka 3.5.2 更新。

有关 Kafka 3.5.1 和 3.5.2 中修复的问题的详情，请参考 [Kafka 3.5.1](#) 和 [Kafka 3.5.2](#) 发行注记。

有关 AMQ Streams 2.5.2 中解决问题的更多详细信息，请参阅 [AMQ Streams 2.5.x 解决问题](#)。

7.2. 修复了 AMQ STREAMS 2.5.1 的问题

KAFKA-15353

2.5.1 补丁版本包含 KAFKA-15353 的修复，它是 Kafka 3.5.2 发行版本中包含的问题。请注意，补丁版本针对这个特定问题引入了一个修复，而不是为 Kafka 3.5.2 修复的所有问题。

有关此问题的更多信息，请参阅 [Kafka 3.5.2](#) 发行注记。

HTTP/2 DoS 漏洞(CVE-2023-44487)

该发行版本解决了 CVE-2023-44487，这是 HTTP/2 协议中的关键拒绝服务(DoS)漏洞。此漏洞源自不当处理多路复用流，从而使恶意客户端能够重复请求新流，并使用 **RST_STREAM** 帧立即取消它们。通过这样做，攻击者可以强制服务器扩展资源设置和停止流，而无需每个连接到达活动流的服务器端限制。有关此漏洞的更多信息，请参阅 [CVE-2023-44487](#) 页面了解描述。

有关 AMQ Streams 2.5.1 中解决问题的详情，请参阅 [AMQ Streams 2.5.x 解决问题](#)。

7.3. 修复了 AMQ STREAMS 2.5.0 的问题

表 7.1. 修复的问题

问题号	Description
ENTMQST-3757	[KAFKA] Mirror Maker 2 negative lag
ENTMQST-4496	[bridge] Logged HTTP 响应状态代码可能与返回到客户端的实际不同
ENTMQST-4707	使连接器任务 backoff 在 Kafka Connect 中进行配置

表 7.2. 修复的常见漏洞和风险(CVE)

问题号	Description
ENTMQST-4484	snakeyaml: Constructor Deserialization Remote Code Execution

问题号	Description
ENTMQST-4995	TRIAGE-CVE-2023-34454 snappy-java-repolib: snappy-java: Integer overflow in compress result to DoS
ENTMQST-4996	TRIAGE-CVE-2023-34454 snappy-java-debuginfo: snappy-java: Integer overflow in compress result to DoS
ENTMQST-4997	TRIAGE-CVE-2023-34454 snappy-java: Integer overflow in compress leads to DoS
ENTMQST-4998	TRIAGE-CVE-2023-34455 snappy-java: Unchecked 块长度会导致 DoS
ENTMQST-5120	在导航 TLS 握手时, CVE-2023-34462 Flaw in Netty 的 SniHandler。
ENTMQST-5121	CVE-2023-0482 REStEasy : 创建不安全的临时文件
ENTMQST-5122	CVE-2022-24823 netty: 全局可读临时文件, 包含敏感数据
ENTMQST-5123	CVE-2021-37137 netty-codec: SnappyFrameDecoder 不限制块长度, 并可能会以不必要的方式缓冲可跳过的块
ENTMQST-5124	CVE-2021-37136 netty-codec: Bzip2Decoder 不允许为解压缩的数据设置大小限制
ENTMQST-5125	在处理精心设计的 GZIP 归档时, OOio 客户端的 CVE-2023-3635 DoS
ENTMQST-5126	带有多部分支持的 CVE-2023-26048 Jetty servlets 可能会导致客户端请求造成 OOM 错误
ENTMQST-5127	Jetty 中的 CVE-2023-26049 非标准 Cookie 解析可能会允许攻击者其他 Cookie 中的 smuggle cookies
ENTMQST-5128	CVE-2022-36944 scala: deserialization gadget 链
ENTMQST-5134	TRIAGE-CVE-2023-3635 okio: GzipSource 类不正确的异常处理
ENTMQST-5178	CVE-2023-26048 jetty-server: OutOfMemoryError for large multipart without filename via request.getParameter ()
ENTMQST-5179	CVE-2023-26049 jetty-server: Cookie 解析带引号的值可能会从其他 Cookie 中破坏值

第 8 章 已知问题

本节列出了 RHEL 上 AMQ Streams 2.5 的已知问题。

8.1. 以 FIPS 模式运行时的 JMX 身份验证

当在启用了 JMX 身份验证的 FIPS 模式下运行 AMQ Streams 时，客户端可能会失败。要临时解决这个问题，在以 FIPS 模式运行时不要启用 JMX 身份验证。我们正在调查此问题，并在以后进行解决。

第 9 章 支持的配置

AMQ Streams 2.5 发行版本支持的配置。

9.1. 支持的平台

以下平台针对在 Red Hat Enterprise Linux (RHEL)发布的版本中使用 Kafka 运行的 AMQ Streams 2.5 测试。

操作系统	架构	JVM
RHEL 7	x86, amd64	Java 11
RHEL 8 和 9	x86, amd64, ppc64le (IBM Power), s390x (IBM Z 和 IBM® LinuxONE), aarch64 (64-bit ARM)	Java 11 和 Java 17

平台使用 Open JDK 11 和 17 测试。IBM JDK 被支持，但不在每个发行版本中定期测试。Open JDK 8、Oracle JDK 8 和 11 和 IBM JDK 8 不被支持。



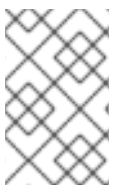
注意

仅在运行 Kafka 3.5.0 时，对 aarch64 (64 位 ARM)的支持适用于 AMQ Streams 2.5。

9.2. 支持的 APACHE KAFKA 生态系统

在 AMQ Streams 中，只支持从 Apache Software Foundation 直接发布的以下组件：

- Apache Kafka Broker
- Apache Kafka Connect
- Apache MirrorMaker
- Apache MirrorMaker 2
- Apache Kafka Java Producer, Consumer, Management client, 和 Kafka Streams
- Apache ZooKeeper



注意

Apache ZooKeeper 仅作为 Apache Kafka 的实现详情支持，不应出于其他目的进行修改。另外，分配给 ZooKeeper 节点的内核数或 vCPU 不包含在订阅合规计算中。换句话说，ZooKeeper 节点不会计算客户的订阅。

9.3. 其他支持的功能

- Kafka Bridge
- Drain Cleaner

- Sything Control
- 分布式追踪

另请参阅 [第 11 章 支持的与红帽产品集成](#)。

9.4. 存储要求

Kafka 需要块存储；NFS 等文件存储选项不兼容。

其他资源

有关 AMQ Streams 2.2 LTS 发行版本支持的配置的详情，[请查看客户门户网站中的 AMQ Streams 支持的配置](#) 文章。

第 10 章 组件详情

下表显示了每个 AMQ Streams 发行版本的组件版本。

AMQ Streams	Apache Kafka	Strimzi Operators	Kafka Bridge	Oauth	Sything Control
2.5.2	3.5.0 (+ 3.5.2)	0.36.0	0.26	0.13.0	2.5.123
2.5.1	3.5.0	0.36.0	0.26	0.13.0	2.5.123
2.5.0	3.5.0	0.36.0	0.26	0.13.0	2.5.123
2.4.0	3.4.0	0.34.0	0.25.0	0.12.0	2.5.112
2.3.0	3.3.1	0.32.0	0.22.3	0.11.0	2.5.103
2.2.2	3.2.3	0.29.0	0.21.5	0.10.0	2.5.103
2.2.1	3.2.3	0.29.0	0.21.5	0.10.0	2.5.103
2.2.0	3.2.3	0.29.0	0.21.5	0.10.0	2.5.89
2.1.0	3.1.0	0.28.0	0.21.4	0.10.0	2.5.82
2.0.1	3.0.0	0.26.0	0.20.3	0.9.0	2.5.73
2.0.0	3.0.0	0.26.0	0.20.3	0.9.0	2.5.73
1.8.4	2.8.0	0.24.0	0.20.1	0.8.1	2.5.59
1.8.0	2.8.0	0.24.0	0.20.1	0.8.1	2.5.59
1.7.0	2.7.0	0.22.1	0.19.0	0.7.1	2.5.37
1.6.7	2.6.3	0.20.1	0.19.0	0.6.1	2.5.11
1.6.6	2.6.3	0.20.1	0.19.0	0.6.1	2.5.11
1.6.5	2.6.2	0.20.1	0.19.0	0.6.1	2.5.11
1.6.4	2.6.2	0.20.1	0.19.0	0.6.1	2.5.11
1.6.0	2.6.0	0.20.0	0.19.0	0.6.1	2.5.11
1.5.0	2.5.0	0.18.0	0.16.0	0.5.0	-

AMQ Streams	Apache Kafka	Strimzi Operators	Kafka Bridge	Oauth	Sything Control
1.4.1	2.4.0	0.17.0	0.15.2	0.3.0	-
1.4.0	2.4.0	0.17.0	0.15.2	0.3.0	-
1.3.0	2.3.0	0.14.0	0.14.0	0.1.0	-
1.2.0	2.2.1	0.12.1	0.12.2	-	-
1.1.1	2.1.1	0.11.4	-	-	-
1.1.0	2.1.1	0.11.1	-	-	-
1.0	2.0.0	0.8.1	-	-	-



注意

Strimzi 0.26.0 包括一个 Log4j 漏洞。产品中包含的版本已更新，以取决于不包含漏洞的版本。

第 11 章 支持的与红帽产品集成

AMQ Streams 2.5 支持与以下红帽产品集成：

红帽单点登录

提供 OAuth 2.0 身份验证和 OAuth 2.0 授权。

有关这些产品可引入您的 AMQ Streams 部署的功能信息，请参阅产品文档。

11.1. 红帽单点登录

AMQ Streams 支持通过 Red Hat Single Sign-On [Authorization Services](#) 使用基于 OAuth 2.0 令牌的授权，它允许您集中管理安全策略和权限。

其他资源

- [Red Hat Single Sign-On 支持的配置](#)

更新于 2024-09-10