



Red Hat AMQ Broker 7.12

Red Hat AMQ Broker 7.12 发行注记

AMQ Broker 发行注记

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

本发行注记包含有关 AMQ Broker 7.12 发行版本中包含的新功能、增强功能、修复和问题的最新信息。

目录

使开源包含更多	3
第 1 章 AMQ BROKER 7.12 的长期支持	4
第 2 章 支持的配置	5
第 3 章 新的和更改的功能	6
第 4 章 已弃用的功能	9
第 5 章 修复的问题	10
第 6 章 修复了常见漏洞和风险	11
第 7 章 已知问题	13
第 8 章 重要链接	15

使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。详情请查看 [CTO Chris Wright 的信息](#)。

第 1 章 AMQ BROKER 7.12 的长期支持

AMQ Broker 7.12 被指定为 Long Term Support (LTS) 发行版本。有关 LTS 版本条款的详情，请参阅 [AMQ LTS 版本的支持时间？](#)

支持 Red Hat Enterprise Linux 和 OpenShift Container Platform

AMQ Broker 7.12 LTS 版本支持：

- Red Hat Enterprise Linux 7、8 和 9
- OpenShift Container Platform 4.12, 4.13, 4.14 或 4.15

红帽努力确保 AMQ Broker 保持与将来的 OpenShift Container Platform 版本兼容，但无法保证这种兼容性。为每个新的 OpenShift Container Platform 版本执行互操作性测试。如果没有发现兼容性问题，新的 OpenShift Container Platform 版本将添加到 [Red Hat AMQ Broker 7 支持的配置](#) 中。

第 2 章 支持的配置

有关支持的配置的详情，请参考 [Red Hat AMQ Broker 7 支持的配置](#)。

最低 Java 版本

至少，AMQ Broker 7.12 需要 Java 版本 11 才能运行。

OpenWire 支持

自 2017 年发布以来，AMQ 7 Broker 提供了对 Openwire 协议的支持，作为将客户端应用程序迁移到 AMQ 7 的方法。2021 年发布 AMQ Broker 7.9.0 后，Openwire 协议已被弃用，并鼓励客户将现有 Openwire 客户端应用程序迁移到 AMQ 7 的完全支持协议之一(CORE、AMQP、MQTT 或 STOMP)。从 AMQ Broker 8.0 发行版本开始，Openwire 协议将从 AMQ Broker 中删除。

第 3 章 新的和更改的功能

本节论述了 AMQ Broker 7.12 中突出显示的一组增强功能和更改的功能。

AMQ Broker 与 Openshift 的 cert-manager Operator 集成以进行证书管理

在 OpenShift 上的 AMQ Broker 7.12 中，您可以使用 cert-manager Operator for Openshift 创建和管理在 AMQ Broker 上配置 TLS 所需的证书。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [为 Openshift 使用 cert-manager Operator](#)。

AMQ Broker Openshift 服务提供 TLS 证书

如果要保护同一 Openshift 集群中的代理和客户端之间的内部连接，您可以在 acceptor 服务中添加注解来请求 Openshift 生成服务用 TLS 证书。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [使用 Openshift 服务服务证书](#)。

支持 Privacy Enhanced Mail (PEM) 证书

在 AMQ Broker 7.12 中，增加了对 PEM 格式的 TLS 证书的支持。

支持在具有受限策略的 Openshift 命名空间中部署

默认情况下，您可以在具有受限 OpenShift 安全上下文约束的命名空间中在 OpenShift 上部署 AMQ Broker 7.12。除了 pod 安全选项外，如果您要在不同 OpenShift 安全上下文中运行代理，您可以自定义 CR 中的容器安全选项。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [自定义资源配置参考](#)。

隔离 brokerProperties 配置

如果您的 OpenShift 上 AMQ Broker 7.12 的自定义资源(CR)包含 **brokerProperties** 部分，并且 CR 处于最大大小限制 1MB，您可以将 **brokerProperties** 配置隔离到一个或多个 Java 属性文件中，然后在 CR 中引用。您可能还想在单独的文件中隔离 **brokerProperties** 配置，以逻辑方式对 **brokerProperties** 项进行分组，以便更轻松地维护。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [访问 brokerProperties 配置](#)。

除了 Openshift 上的路由外，还支持 ingresses

在 OpenShift 上的 AMQ Broker 7.12 中，除了路由外，您还可以使用入口功能向 Openshift 集群以外的客户端公开接收器、连接器和代理控制台。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [配置接收器](#)。

支持为第三方 JAR 文件挂载共享卷

在 OpenShift 上的 AMQ Broker 7.12 中，您可以将 Operator 配置为在集群中的每个代理 pod 上挂载共享卷。在每个 pod 上挂载共享卷的用例是存储第三方 JAR 文件，如用于 JDBC 数据库的 JAR 文件（代理需要）。在 RHEL 和 Openshift 平台上，您可以扩展 Java 类路径，以便在运行时为代理提供额外的 JAR 文件。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [添加第三方 JAR 文件](#)。

自定义 Operator 创建的 Openshift 资源

在 OpenShift 上的 AMQ Broker 7.12 中，您可以自定义 Openshift 资源，如部署、Pod 和由 Operator 创建和管理的服务。如果要执行某些任务，则自定义这些资源很有用，例如：

- 添加自定义注解，以控制资源如何被其他服务处理。
- 修改没有在代理 CR 中公开的属性。
如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [自定义 Operator 创建的 Openshift 资源](#)。

支持在 Openshift 上的 AMQ Broker 中添加插件

在 OpenShift 上的 AMQ Broker 7.12 中，您可以通过在 CR 中注册插件来扩展 AMQ Broker 的功能。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [使用 AMQ Broker 注册插件](#)。

安全集群连接支持

在 Openshift 上的 AMQ Broker 7.12 中，您可以通过为内部接收器和连接器启用 SSL 来保护集群连接。如需更多信息，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [保护集群连接](#)。

自动重新载入 SSL 工件

在 OpenShift 和 RHEL 上的 AMQ Broker 7.12 中，您可以将 AMQ Broker 配置为重新加载更新的 TLS 证书和其他更改密钥存储或信任存储配置，而无需重启代理。要配置自动重新载入，请为 acceptor 设置 `sslAutoReload` 属性。有关如何在 Openshift 上配置 SSL 工件自动重新载入的示例，请参阅在 [Openshift 上部署 AMQ Broker](#) 中的 [为 Openshift 使用 cert-manager Operator](#)。

集群代理的健康检查

在 AMQ Broker 7.12 中，您可以使用 `artemis check cluster` 命令行工具来验证集群中代理节点的拓扑。如需更多信息，请参阅 [配置 AMQ Broker](#) 中的 [检查代理、队列和集群的健康状态](#)。

使用 AMQP 代理连接的联邦支持

在 AMQ Broker 7.12 中，您可以通过出站 AMQP 代理连接配置地址和队列的联邦。与使用 Core 协议相比，使用 AMQP 协议提供以下优点：

- 如果客户端使用 AMQP 协议进行消息传递，请使用 AMQP 协议进行联邦，以消除 AMQP 和 Core 之间的消息转换。
- AMQP 联邦支持通过单一传出连接进行双向联邦。双向支持消除了远程代理重新连接到本地代理的需求，这是当您为联邦使用 Core 协议时的要求，这些协议可能会被网络策略阻止。
- AMQP 联邦提供对代理间消息移动的更好的控制，以避免消息在代理间移动。

如需更多信息，请参阅 [配置 AMQ Broker](#) 中的 [使用 AMQP 协议配置联邦](#)。

使用命令行界面使用自定义 shell

在 AMQ Broker 7.12 中，您可以使用 AMQ Broker 命令行界面中的自定义 `artemis` shell 与代理交互。自定义 shell 具有命令和命令参数的内置自动完成功能。如需更多信息，请参阅 [管理 AMQ Broker](#) 中的 [在 artemis shell 中使用 CLI](#)。

包含通配符的地址的字面匹配

在 AMQ Broker 7.12 中，您可以配置字面匹配，将通配符字符视为包含通配符的匹配地址的字面字符。如需更多信息，请参阅 [配置 AMQ Broker](#) 中的 [配置字面匹配](#)。

对 JMX 管理操作进行基于角色的访问控制

在 AMQ Broker 7.12 中，您可以使用两个新权限来 [查看和编辑](#)，为 JMX 管理操作配置基于角色的访问控制，而无需代理重启。有关在 RHEL 上配置基于角色的访问控制的详情，请参考 [配置 AMQ Broker](#) 中的 [在 broker.xml 文件中配置基于角色的访问控制](#)。有关在 Openshift 中配置基于角色的访问控制的详情，请参考在 [Openshift 上部署 AMQ Broker](#) 中的 [为管理操作配置基于角色的访问控制](#)。

更改到 queue stat 命令的输出格式

`queue stat` 命令的输出格式已从 7.11 和早期版本的 AMQ Broker 中的输出更改，这可能会影响部署中运行的自动化进程。

在 MQTT acceptor 中可配置的新参数，以自动删除 MQTT 订阅队列

在 AMQ Broker 7.12 中，您可以在 MQTT acceptor 中配置 `defaultMqttSessionExpiryInterval` 参数，以自动删除在相应客户端会话过期时不会被删除的 MQTT 订阅队列。new 参数代表在代理删除会话状态和订阅队列前必须经过的秒数。在 7.12 之前，需要在 `address-setting` 中配置 `auto-delete` 114 参数，以删除在客户端会话过期时不会被删除的队列。

Operator 频道

AMQ Broker Operator、[Red Hat Integration - AMQ Broker for RHEL 8 \(Multiarch\)](#) 有以下频道：

- **7.12.x** - 此频道仅为版本 7.12 提供更新，并且是 Long Term Support (LTS) 频道。

- **7.11.x** - 此频道仅为版本 7.11 提供更新，并且是 Long Term Support (LTS) 频道。
- **7.10.x** - 此频道仅为版本 7.10 提供更新，并且是 Long Term Support (LTS) 频道。



注意

无法通过切换频道来升级 Operator。您必须卸载现有 Operator，并从适当的频道安装 Operator 的新版本。

要确定需要选择哪个 Operator，请参阅 [Red Hat Enterprise Linux Container Compatibility Matrix](#) 。

第 4 章 已弃用的功能

本节论述了支持但已从 AMQ Broker 中弃用的功能。

ActiveMQArtemisAddress CRD

从 7.12 开始，**ActiveMQArtemisAddress** CRD 已被弃用。使用 **ActiveMQArtemis** CR 中的 **spec.brokerProperties** 属性为您的部署创建地址和队列。

ActiveMQArtemisSecurity CRD

从 7.12 开始，**ActiveMQArtemisSecurity** CRD 已被弃用。使用 **ActiveMQArtemis** CR 中的 **spec.brokerProperties** 属性来配置部署的安全性。

ActiveMQArtemisScaledown CRD

从 7.12 开始，**ActiveMQArtemisScaledown** CRD 已被弃用。**ActiveMQArtemisScaledown** CRD 由代理在内部使用，因此此更改对 AMQ Broker 管理员是透明的。

LDAP 查询的连接池

从 7.12 开始，**connectionPool** 参数（为 LDAP 查询启用连接池）已弃用。内置的授权和身份验证缓存提供了优化 LDAP 查询性能的替代方法。有关自定义内置缓存的详情，[请参考配置身份验证和授权缓存](#)。

自定义资源中的升级 属性

从 7.11 开始，**upgrade** 属性和关联的 **enabled** 和 **minor** 属性已被弃用，因为它们无法像最初的设计一样工作。使用 **image** 或 **version** 属性来部署特定的代理容器镜像。

queues 配置元素

从 7.10 开始，`<queues>` 配置元素已弃用。您可以使用 `<addresses>` 配置元素来创建地址和相关队列。以后的发行版本中会删除 `<queues>` 配置元素。

getAddressesSettings 方法

从 7.10 开始，`getAddressesSettings` 方法（包含在 `org.apache.activemq.artemis.core.config.Configuration` 接口）已被弃用。使用 `getAddressSettings` 方法，以编程方式为代理程序配置地址和队列。

OpenWire 协议

从 7.9 开始，OpenWire 协议是一个已弃用的功能。如果要创建基于 AMQ Broker 的新的系统，请使用其他支持的协议之一。从 8.0 发行版本开始，Openwire 协议将从 AMQ Broker 中删除。

当代理实例没有运行时添加用户

从 7.8 开始，当 AMQ Broker 实例没有运行时，从 CLI 界面中添加用户到代理的功能会被删除。

网络 pinger

从 7.5 开始，网络 ping 是一个已弃用的功能。网络 ping 无法保护代理集群免受网络隔离问题，这可能会导致无法恢复的消息丢失。这个功能将在以后的发行版本中被删除。红帽继续支持使用网络 ping 的现有 AMQ Broker 部署。但是，红帽不推荐在新部署中使用网络 ping。有关配置代理集群以实现高可用性的指导，并避免网络隔离问题，[请参阅配置 AMQ Broker 中的 实施高可用性](#)。

Hawtio 分配控制台插件

从 7.3 开始，AMQ Broker 不再附带 Hawtio 分配控制台插件 `assign-hawtio-console.war`。在以前的版本中，分配控制台用于管理 AMQ Interconnect。但是，AMQ Interconnect 现在使用自己的独立 web 控制台。

第 5 章 修复的问题

有关版本中修复的问题的完整列表，请参阅 [AMQ Broker 7.12.0 修复的问题](#)

第 6 章 修复了常见漏洞和风险

本节详细介绍了 AMQ Broker 7.12 发行版本中修复的通用漏洞和风险(CVE)。

- [ENTMQBR-8644](#) - TRIAGE CVE-2023-6717 keycloak: XSS via SAML POST-binding 流中的断言消费者服务 URL [amq-7]
- [ENTMQBR-8976](#) - TRIAGE CVE-2024-29025 netty-codec-http: Allocation of resources Without Limits 或 Throttling [amq-7]
- [ENTMQBR-8927](#) - CVE-2024-22259 springframework: URL Parsing with Host Validation [amq-7]
- [ENTMQBR-8740](#) - CVE-2024-1132 keycloak: 重定向验证中的路径转换器 [amq-7]
- [ENTMQBR-8758](#) - CVE-2024-1249 keycloak: org.keycloak.protocol.oidc: 在 checkLoginIframe 中未验证的跨原始消息会导致 DDoS [amq-7]
- [ENTMQBR-8626](#) - CVE-2023-6378 logback: serialization vulnerability in logback receiver [amq-7]
- [ENTMQBR-8627](#) - CVE-2023-6481 日志转发 : 日志接收器中的序列化漏洞 [amq-7]
- [ENTMQBR-8953](#) - CVE-2024-29131 CVE-2024-29133 commons-configuration2: various flaws [amq-7]
- [ENTMQBR-8702](#) - CVE-2023-44981 zookeeper: Authorization Bypass in Apache ZooKeeper [amq-7]
- [ENTMQBR-8611](#) - CVE-2022-41678 activemq: Apache ActiveMQ: Deserialization vulnerability on Jolokia, 它允许经过身份验证的用户执行 RCE [amq-7]
- [ENTMQBR-8225](#) - CVE-2023-24540 amq-broker-rhel8-operator-container: golang: html/template:不正确的处理 JavaScript 空格 [amq-7]
- [ENTMQBR-8227](#) - CVE-2022-21698 amq-broker-rhel8-operator-container: prometheus/client_golang: Denial of service using InstrumentHandlerCounter [amq-7]
- [ENTMQBR-8238](#) - CVE-2022-21698 CVE-2023-24534 amq-broker-rhel8-operator-container: golang: net/http, net/textproto: 拒绝服务不受过度内存分配 [amq-7]
- [ENTMQBR-8239](#) - CVE-2023-29400 amq-broker-rhel8-operator-container: golang: html/template:不正确的处理空 HTML 属性 [amq-7]
- [ENTMQBR-8240](#) - CVE-2023-24539 amq-broker-rhel8-operator-container: golang: html/template: incorrect sanitization of CSS 值 [amq-7]
- [ENTMQBR-8228](#) - CVE-2021-43565 amq-broker-rhel8-operator-container: golang.org/x/crypto: 空明文数据包会导致 panic [amq-7]
- [ENTMQBR-8230](#) - CVE-2022-41723 amq-broker-rhel8-operator-container: net/http, golang.org/x/net/http2: 避免 HPACK 解码 [amq-7] 中的数量复杂性。
- [ENTMQBR-8236](#) - CVE-2023-24536 amq-broker-rhel8-operator-container: golang: net/http, net/textproto, mime/multipart: 拒绝服务被过度消耗 [amq-7]

- [ENTMQBR-8237](#) - CVE-2023-24537 amq-broker-rhel8-operator-container: golang: go/parser: Infinite loop in resolve [amq-7]
- [ENTMQBR-8231](#) - CVE-2022-2879 amq-broker-rhel8-operator-container: golang: archive/tar: unbounded 内存消耗 (在读取标头 [amq-7])
- [ENTMQBR-8229](#) - CVE-2022-27664 amq-broker-rhel8-operator-container: golang: net/http: handle server error after send GOAWAY [amq-7]
- [ENTMQBR-8226](#) - CVE-2022-32189 amq-broker-rhel8-operator-container: golang: math/big: decoding big.Float 和 big.Rat 类型, 如果编码的消息太短, 可能允许拒绝服务 [amq-7]
- [ENTMQBR-8232](#) - CVE-2022-41715 amq-broker-rhel8-operator-container: golang: regexp/syntax: 限制由解析 regexps [amq-7] 使用的内存
- [ENTMQBR-8241](#) - CVE-2023-24538 amq-broker-rhel8-operator-container: golang: html/template: backticks not treated as as string delimiters [amq-7]
- [ENTMQBR-8233](#) - CVE-2022-2880 amq-broker-rhel8-operator-container: golang: net/http/httputil: ReverseProxy 不应转发不可解析的查询参数 [amq-7]
- [ENTMQBR-8234](#) - CVE-2022-41724 amq-broker-rhel8-operator-container: golang: crypto/tls: large handshake 记录可能会导致 panics [amq-7]
- [ENTMQBR-8608](#) - CVE-2022-41678 activemq-broker-operator: Apache ActiveMQ: Deserialization vulnerability on Jolokia, 它允许经过身份验证的用户执行 RCE [amq-7]
- [ENTMQBR-8235](#) - CVE-2022-41725 amq-broker-rhel8-operator-container: golang: net/http, mime/multipart: 拒绝服务不受过度资源消耗 [amq-7]
- [ENTMQBR-8671](#) - CVE-2023-51074 json-path: stack-based buffer overflow in Criteria.parse method [amq-7]

第 7 章 已知问题

本节论述了 AMQ Broker 7.12 中已知的问题。

- ENTMQBR-9103** - 在关闭多个使用 AMQP 的线程时，`NullPointerException`

当为 AMQP 信息运行多线程消费者时，代理有时会生成 WARN 级别日志消息，如下所示：

```
2024-05-13 18:11:46,048 WARN [io.netty.util.concurrent.AbstractEventExecutor] 一个任务引发异常。任务：
org.apache.activemq.artemis.protocol.amqp.proton.AMQPLargeMessageWriter$$Lambda$643/0
java.lang.NullPointerException: null
```

当客户端完成使用消息且有时由堆栈追踪附带时，会生成消息。

没有信息会丢失，消息可以被忽略。
- ENTMQBR-8106** - 在更改 CR 中的 `MessageMigration` 后 AMQ Broker Drainer pod 无法正常工作

您无法更改正在运行的代理部署中 `messageMigration` 属性的值。要临时解决这个问题，您必须新的 **ActiveMQ Artemis** CR 中为 `messageMigration` 属性设置所需的值，并创建新的代理部署。
- ENTMQBR-8166** - 带有 `UseClientAuth=true` 的自签名证书可防止与 Jolokia 的 Operator 通信

如果在 **ActiveMQ Artemis** CR 的 `console` 部分中将 `useClientAuth` 属性设置为 `true`，Operator 无法在代理上配置某些功能，如创建地址。在 Operator 日志中，您会看到以 `remote error: tls: bad certificate` 结尾的错误消息。
- ENTMQBR-7359** - 更改为当前使用 7.10.0 Operator 的凭证 secret 处理

Operator 存储管理员用户名和密码，用于在 secret 中连接到代理。默认 secret 名称格式为 `<custom-resource-name>-credentials-secret`。您可以手动创建 secret，或允许 Operator 创建 secret。

如果在 7.10.0 之前的自定义资源中配置了 `adminUser` 和 `adminPassword` 属性，Operator 会使用这些属性的值更新手动创建的 secret。从 7.10.0 开始，Operator 不再更新手动创建的 secret。因此，如果您更改了 CR 中的 `adminUser` 和 `adminPassword` 属性的值，则必须：

 - 使用新用户名和密码更新 secret
 - 删除 secret 并允许 Operator 创建 secret。当 Operator 创建 secret 时，如果在 CR 中指定它们，它会添加 `adminUser` 和 `adminPassword` 属性的值。如果这些属性不在 CR 中，Operator 会为 secret 生成随机凭证。
- ENTMQBR-7111** - 7.10 版本的 Operator 通常会在升级过程中删除 StatefulSet

如果您要升级到 AMQ Broker Operator 7.10.0，新的 Operator 会在协调过程中自动删除每个部署的现有 StatefulSet。当 Operator 删除 StatefulSet 时，现有代理 pod 会被删除，这会导致临时代理中断。

您可以运行以下命令来手动删除 StatefulSet，并在 Operator 被删除 StatefulSet 前删除正在运行的 pod：

```
oc delete statefulset <statefulset-name> --cascade=orphan
oc delete statefulset <statefulset-name> --cascade=orphan
```

在升级过程中手动删除 StatefulSet，新 Operator 可以在不删除正在运行的 pod 的情况下协调 StatefulSet。如需更多信息，请参阅在 [OpenShift 上部署 AMQ Broker 中的使用 OperatorHub 升级 Operator](#)。
- ENTMQBR-5749** - 删除 OperatorHub 中可见的不受支持的 operator

仅支持从 OperatorHub 部署 Operator 中提到的 Operator 和 Operator 频道。出于与 Operator 出版物相关的技术原因，其他 Operator 和频道在 OperatorHub 中可见，应该会被忽略。为了便于参考，以下列表显示了哪些 Operator 可见，但不支持：

- Red Hat Integration - AMQ Broker LTS - 所有频道
- Red Hat Integration - AMQ Broker - alpha、current 和 current-76
- **ENTMQBR-4140 - AMQ Broker Operator**：如果 **storage.size** 被不正确的指定，则安装将不可用

如果您配置自定义资源(CR)实例的 **storage.size** 属性，以指定部署持久性存储中的代理所需的持久性卷声明(PVC)大小，如果没有正确指定这个值，Operator 安装将不可用。例如，假设您将 **storage.size** 的值设置为 **1** (即，没有指定单元)。在这种情况下，Operator 无法使用 CR 创建代理部署。另外，即使您删除 CR 并使用正确指定 **storage.size** 部署新版本，Operator 仍无法使用此 CR 创建部署。

要临时解决这个问题，首先停止 Operator。在 OpenShift Container Platform Web 控制台中，单击 **Deployments**。对于与 AMQ Broker Operator 对应的 Pod，点 **More options** 菜单（三个垂直点）。单击 **Edit Pod Count**，并将值设为 **0**。当 Operator Pod 停止后，使用 **storage.size** 创建 CR 的新版本。然后，要重启 Operator，请再次点 **Edit Pod Count**，并将值设为 **1**。

- **ENTMQBR-4141 - AMQ Broker Operator**:增加持久性卷大小需要在重新创建状态集后手动参与
如果您试图增加部署持久性存储中的代理所需的持久性卷声明(PVC)的大小，则更改不会生效，而无需进一步手动步骤。例如，假设您配置自定义资源(CR)实例的 **storage.size** 属性，以指定 PVC 的初始大小。如果您修改了 CR 来指定 **storage.size** 的不同值，现有代理将继续使用原始 PVC 大小。即使将部署缩减为零个代理，也会发生这种情况，然后备份到原始数字。但是，如果您扩展部署的大小以添加额外的代理，则新代理将使用新的 PVC 大小。

要临时解决这个问题，并确保部署中的所有代理都使用相同的 PVC 大小，使用 OpenShift Container Platform Web 控制台扩展部署使用的 PVC 大小。在控制台中，单击 **Storage → Persistent Volume Claims**，点您的部署。在右上角的 **Actions** 下拉菜单中，选择 **Expand PVC** 并输入新值。

第 8 章 重要链接

- [Red Hat AMQ Broker 7.11 发行注记](#)
- [Red Hat AMQ Broker 7.10 发行注记](#)
- [Red Hat AMQ Broker 7.9 发行注记](#)
- [Red Hat AMQ Broker 7.8 发行注记](#)
- [Red Hat AMQ Broker 7.7 发行注记](#)
- [Red Hat AMQ Broker 7.6 发行注记](#)
- [Red Hat AMQ Broker 7.1 到 7.5 发行注记\(aggregated\)](#)
- [Red Hat AMQ 7 支持的配置](#)
- [Red Hat AMQ 7 组件详情](#)

更新于 2024-05-22