Red Hat AMQ 7.2

AMQ Streams 1.1.0 on OpenShift Container Platform Release Notes

Release Notes for AMQ Streams 1.1.0

Last Updated: 2019-03-18

Red Hat AMQ 7.2 AMQ Streams 1.1.0 on OpenShift Container Platform Release Notes

Release Notes for AMQ Streams 1.1.0

Legal Notice

Abstract

These release notes contain the latest information about new features, enhancements, fixes, and issues contained in the AMQ Streams 1.1.0 release.

Table of Contents

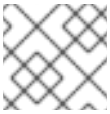
CHAPTER 1. FEATURES

The features added in this release, and that were not in previous releases of the AMQ Streams, are outlined below.

Kafka 2.1.1 support

AMQ Streams now supports Kafka 2.1.1.
You must upgrade the Cluster Operator to 1.1.0 before you can upgrade to Kafka 2.1.1. For instructions, see AMQ Streams and Kafka upgrades.

Refer to the Kafka 2.1.0 and Kafka 2.1.1 Release Notes for additional information.



NOTE

Kafka 2.0.0 is supported only for upgrade purposes.

JBOD storage option

You can now configure AMQ Streams to use JBOD, a data storage configuration in which multiple disks or volumes are used. JBOD is one approach to providing increased data storage for Kafka brokers. It is not supported for Zookeeper.
See JBOD storage overview

Labels and annotations for resources

Additional configuration options have been added for configuring resource metadata. You can add labels and annotations to control how Pods are treated by Istio or other services.
See Labels and Annotations

New Pod template fields

Use new template fields to customize your Pod creation:

- **`terminationGracePeriodSeconds`** defines the period of time, in seconds, by which a Pod must have terminated gracefully.
- **`imagePullSecrets`** defines a list of references to OpenShift Secrets that can be used for pulling container images from private repositories.
- **`securityContext`** sets pod-level security attributes for containers running as part of a given Pod.

See Customizing Pods

Mounting Secrets for Kafka Connect

You can mount your own Secrets or ConfigMaps into a Kafka Connect deployment (including Kafka Connect S2I) by specifying the mount in your Pod deployment configuration as either:

- Environment variables
- Volumes (using property files referenced in the configuration)

This approach applies especially to confidential data, such as usernames, passwords, or certificates.

See Using external configuration and secrets

Network policies

Define network policies for Kafka listeners. Use labels to describe permissible connections when configuring the listeners and the **`networkPolicyPeers`** field to specify the application pods or namespaces that will be allowed to access the Kafka cluster.
See Network policies

Maintenance time window

You can schedule certain rolling updates of your Kafka and Zookeeper clusters to start at a convenient time. Maintenance time windows allow you specify the days and times for planned maintenance using Cron expressions. Actions, such as certificate renewals, are started only during the maintenance window, though actions continue for as much time as required and do not stop when the window has ended.
See Maintenance time windows overview

Watch resources across all namespaces

You can configure the Cluster Operator to watch AMQ Streams custom resources across all OpenShift projects in your OpenShift cluster. When running in this mode, the Cluster Operator automatically manages clusters in any new projects or namespaces that are created.
See Deploying the Cluster Operator to watch all namespaces

Custom ImagePullPolicy

You can customize the default image pull policy for containers in all pods deployed by the Cluster Operator. The image pull policy is configured using the environment variable **`STRIMZI_IMAGE_PULL_POLICY`** in the Cluster Operator deployment.
See Cluster Operator Configuration

Pod Disruption Budgets

It is now possible to configure the Pod Disruption Budgets created by default through the Cluster Operator template. The template is set by default for one Pod to be unavailable at a time, which can be can changed using the **`maxUnavailable`** field.
See Customizing Pod Disruption Budgets

CHAPTER 2. ENHANCEMENTS

The following table lists the enhancements in AMQ Streams 1.1.0.

User authorization

Simple Authorization now supports Transactional IDs when specifying resource types.
See Simple Authorization

TLS sidecar health checks

TLS sidecar, used to encrypt and decrypt all communication between the AMQ Streams components and Zookeeper, now supports the following configuration options:

- **`readinessProbe`** defines when a container can start accepting traffic
- **`livenessProbe`** defines when to restart a container

See Healthcheck configurations

Enriched configuration options for off-cluster access

An external listener may be used to connect to a Kafka cluster outside an OpenShift environment. AMQ Streams supports three types of external listeners:

1. **`route`**
2. **`loadbalancer`**
3. **`nodeport`**

With external listeners, you can:

- Override host names and ports
- Add DNS names for the bootstrap service
- Specify target OpenShift routes
- Configure NodePort numbers

See External listener

Sample Prometheus alerts

Example Prometheus alerting rules are provided for Kafka and Zookeeper metrics. The examples may be used when configuring Prometheus Alertmanager.

Annotations refactoring

A new **`Annotations`** class holds a constant for the **`strimzi.io`** domain. Current uses of annotations have been refactored to use the new standardized class. Old annotations are deprecated, but still functional.

- **`cluster.operator.strimzi.io/delete-claim`** → **`strimzi.io/delete-claim`**
- **`operator.strimzi.io/manual-rolling-update`** → **`strimzi.io/manual-rolling-update`**

- **`operator.strimzi.io/delete-pod-and-pvc` → `strimzi.io/delete-pod-and-pvc`**
- **`operator.strimzi.io/generation` → `strimzi.io/generation`**

Kafka and Zookeeper shutdown handling

TLS sidecar now shuts down after the main pods so that connections to Zookeeper are maintained for a clean shutdown.

Certificate validity

Changes to the number of days a security certificate should be valid through **`renewalDays`** are now propagated to **`UserOperator`** to override the (365 day) default validity.

CHAPTER 3. FIXED ISSUES

The following table lists the issues fixed in AMQ Streams 1.1.0.

Issue Number	Description
ENTMQST-639	Topic Operator scalability
ENTMQST-709	Cluster Operator fails to start — can't parse certificate

CHAPTER 4. KNOWN ISSUES

There are no known issues for AMQ Streams 1.1.0.

CHAPTER 5. IMPORTANT LINKS

- Red Hat AMQ 7 Supported Configurations
- Red Hat AMQ 7 Component Details

Revised on 2019-03-18 17:41:33 UTC