Cloudera Streams Messaging Operator 1.0.0

Release Notes

Date published: 2024-06-11
Date modified: 2024-06-11

https://docs.cloudera.com/

Legal Notice

Contents

Release notes

Learn about the new features, improvements, known and fixed issues, limitations, and unsupported features in this release of CSM Operator.

What's New

Learn about the new features and notable changes in this release of CSM Operator.

CSM Operator 1.0.0

This is the initial release of CSM Operator. This release of CSM Operator is based on Strimzi 0.40.0 and Kafka 3.7.0. See the following upstream resources for more information on these versions:

- Strimzi 0.40.0 Release notes
- Kafka 3.7.0 Release notes
- Kafka 3.7.0 Notable changes

To learn more about CSM Operator and its typical deployment architecture, see Overview. For installation instructions, see Installation.

Known Issues

Learn about the known issues in this release of CSM Operator.

CSMDS-334: ZooKeeper pods are running but Kafka pods are not created

Under certain circumstances, ZooKeeper pods might not be able to form a quorum. In a case like this, the creation of the Kafka cluster gets stuck in a state where ZooKeeper pods are running, but Kafka pods are not created.

If you encounter this issue, at least one of the ZooKeeper pods logs a WARN entry similar to the following:

```
2024-02-23 18:45:00,311 WARN Unexpected exception (org.apache.zo
okeeper.server.quorum.QuorumPeer) [QuorumPeer[myid=3](plain=127.
0.0.1:12181)(secure=[0:0:0:0:0:0:0:0]:2181)]
java.lang.InterruptedException: Timeout while waiting for epoch
 from quorum
 at org.apache.zookeeper.server.quorum.Leader.getEpochToPropose
(Leader.java:1443)
 at org.apache.zookeeper.server.quorum.Leader.lead(Leader.java:60
6)
 at org.apache.zookeeper.server.quorum.QuorumPeer.run(QuorumPeer.
java:1552)
```

This is caused by a race condition issue in ZooKeeper. ZooKeeper is unable to recover from this state automatically.

Delete the ZooKeeper pods that are unable to form a quorum.

```
kubectl delete pod [***ZOOKEEPER POD***] -n [***NAMESPACE***]
```

The Strimzi Cluster Operator automatically recreates the ZooKeeper pods that are deleted. The newly created ZooKeeper pods are less likely to encounter the issue.

CSMDS-953: Kafka and ZooKeeper might experience downtime during upgrades

Under certain circumstances, ZooKeeper pods might not be able to form a quorum during an upgrade. This results in both ZooKeeper and Kafka becoming unavailable for several minutes during an upgrade.

After a certain amount of time, failed ZooKeeper pods are automatically recreated by the Strimzi Cluster Operator, and the upgrade succeeds.

If you encounter this issue, at least one of the ZooKeeper pods will log the following error:

```
java.net.BindException: Cannot assign requested address
```

This issue affects ZooKeeper-based deployments only.

CSMDS-644: Kafka CLI commands fail with a TLS handshake error when using kafka_shell.sh

If an mTLS listener is configured for the Kafka cluster targeted by the kafka_shell.sh tool, the tool generates the client configuration incorrectly. As a result, Kafka CLI commands that you run using the tool fail with a TLS handshake error.

```
#...
ERROR org.apache.kafka.common.errors.SslAuthenticationException:
 SSL handshake failed
Caused by: javax.net.ssl.SSLHandshakeException: PKIX path buildin
g failed: sun.security.provider.certpath.SunCertPathBuilderExcep
tion: unable to find valid certification path to requested target
```

Run the following command in the interactive shell that the tool opens.

```
sed -i '8,/name=CONTROLPLANE-9090/d' /tmp/client.properties
```

CSMDS-721: Some alerts are triggered when deploying the prometheus-rules.yaml example

When deploying the prometheus-rules.yaml example, the following alerts are triggered.

- BridgeContainersDown
- EntityOperatorTlsSidecarContainerDown
- KafkaBrokerContainersDown
- MirrorMakerContainerDown

These are false positive results. KafkaBrokerContainersDown has an incorrect expression. The other three alerts are for cluster components that are not supported or are not used in CSM Operator.

CSMDS-805: The kafka_shell.sh and connect_shell.sh tools do not propagate command return code

The kafka_shell.sh and connect_shell.sh tools do not propagate the return code of the last command which ran inside the shell.

None.

Unsupported features

Learn what features are unsupported in this release of CSM Operator.

The following Strimzi features are unsupported in CSM Operator:

- Kafka Connect
- Kafka MirrorMaker
- Kafka MirrorMaker 2
- Kafka Bridge
- Kafka cluster creation without using `KafkaNodePool` resources

Component versions

CSM Operatorcomponents and their versions shipped in this release of CSM Operator.

Table 1: CSM Operatorcomponent versions

Component	Version
Cruise Control	2.5.137.1.0.0-b283
Kafka	3.7.0.1.0.0-b283
Strimzi	0.40.0.1.0.0-b283
ZooKeeper	3.8.1.7.2.18.0-641

Supported Kafka versions

CSM Operator supports the following Kafka versions:

Table 2: Supported Kafka versions

Latest (default)	Other
3.7.0.1.0	None

The latest version is the current and latest supported Kafka version. This version is used by default to deploy clusters if an explicit version is not provided in your `Kafka` resource. The default version is the version recommended by Cloudera. Other supported versions are previous Kafka releases that are also supported.

Notice that the Kafka versions are specific to Cloudera. The version is made up of two parts. The first three digits specify the Apache Kafka version. The digits following the Apache Kafka version specify the major and minor version of CSM Operator. When deploying a cluster, you must use the Cloudera versions for Kafka listed here. Specifying upstream versions is not supported.

System requirements

CSM Operator requires a Kubernetes cluster environment that meets the following system requirements and prerequisites. Ensure that you meet these requirements, otherwise, you will not be able to install and use CSM Operator or its components.

- A Kubernetes 1.23 or later cluster:
  - OpenShift 4.10 or later.
  - RKE2 (Rancher Kubernetes Engine 2) 1.23 or later.

  Note: CSM Operator complies with Cloud Native Computing Foundation (CNCF) standards and is compatible with CNCF-compliant Kubernetes distributions. For supporting your specific Kubernetes distribution, contact Cloudera.
- Administrative rights on the Kubernetes cluster.
- Access to kubectl or oc. These command line tools must be configured to connect to your running cluster.
- Access to helm.
- Log collection is enabled for the Kubernetes cluster. Cloudera requires that the logs of CSM Operator components are stored long term for diagnostic and supportability purposes. Review Log collection.

- A persistent storage class configured on the Kubernetes cluster that satisfies the durability and low-latency requirements for operating Kafka. The ideal storage class configuration can vary per environment and use-case and is determined by the Kubernetes platform where CSM Operator is deployed.

  Additionally, for Kafka brokers, Cloudera recommends a `StorageClass` that has volume expansion enabled (allowvolumeexpansion set to true).
- A Prometheus installation running in the same Kubernetes cluster where you install CSM Operator is recommended. Prometheus is used for collecting and monitoring Kafka and Strimzi metrics.