

Overview

Date published: 2024-06-11

Date modified: 2024-12-02



Legal Notice

© Cloudera Inc. 2025. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

This content is modified and adapted from [Strimzi Documentation](#) by Strimzi Authors, which is licensed under [CC BY 4.0](#).

Contents

Deployment architecture.....4

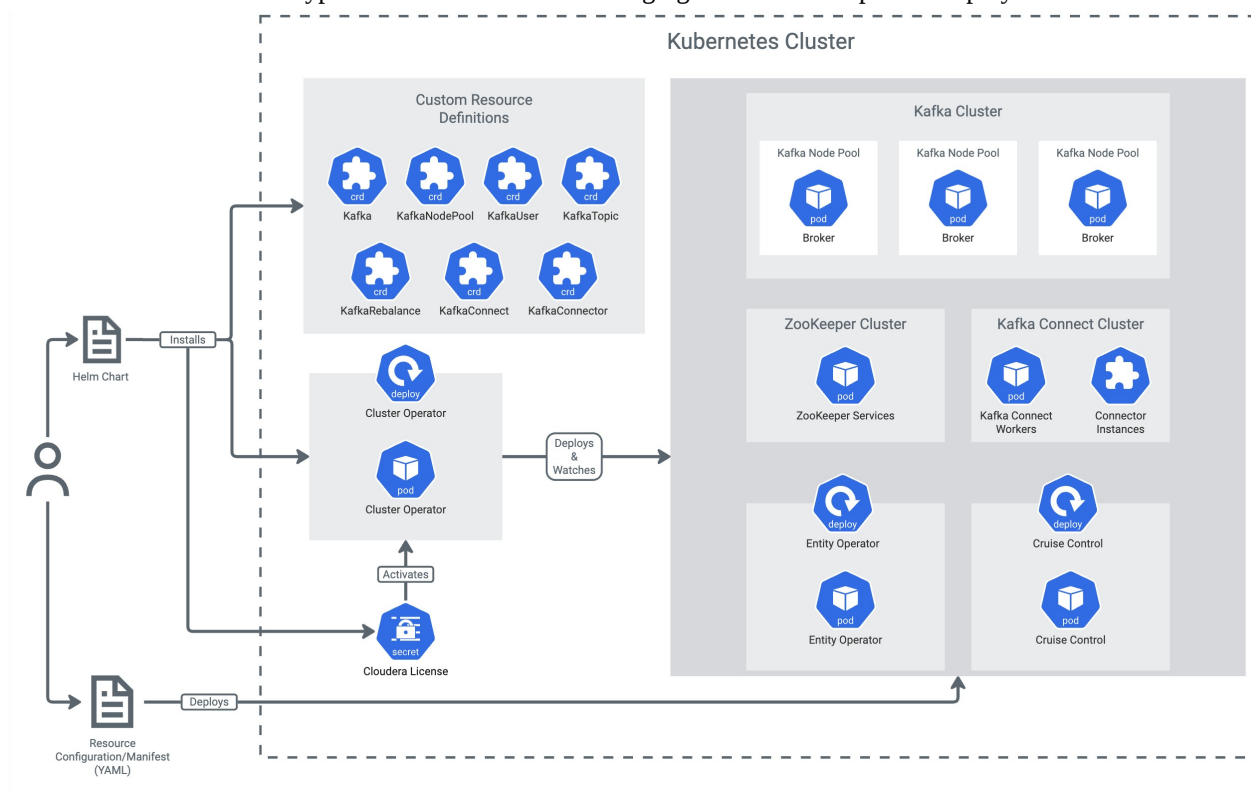
Licensing..... 5

Sizing and performance considerations..... 5

 Recommended minimum setup..... 6

Deployment architecture

Learn the architecture of a typical Cloudera Streams Messaging - Kubernetes Operator deployment.



Cloudera Streams Messaging - Kubernetes Operator deployment starts with first installing a Helm chart.

The Helm chart installs various Custom Resource Definitions (CRDs) to the Kuberentes environment. Resources described by these CRDs are managed by the Kuberentes operator applications. Strimzi defines CRDs such as:

- **Kafka** – represents a Kafka cluster consisting of Kafka brokers, ZooKeeper servers, Cruise Control and Strimzi Entity Operators.
- **KafkaNodePool** – represents a group of Kafka brokers from the cluster that have the same configuration.
- **KafkaTopic** – represents a Kafka topic.
- **KafkaUser** – represents a user that is external to the Kafka cluster.
- **KafkaRebalance** – represents a broker rebalance action for the Kafka cluster.
- **KafkaConnect** – represents a Kafka Connect cluster consisting of one or more Kafka Connect workers.
- **KafkaConnector** – represents a Kafka Connect connector instance.



Note: Strimzi includes CRDs not highlighted here. The use of these CRDs are either not supported in Cloudera Streams Messaging - Kubernetes Operator, or are for internal use by the Strimzi Cluster Operator.

When installing the Helm chart, a Strimzi Cluster Operator Kubernetes deployment is created with a Strimzi Cluster Operator pod running within the deployment. This application is responsible for monitoring cluster components and reconciling these components when their configuration changes. During installation you are required to register a license, which activates the Strimzi Cluster Operator.

Following installation, you deploy various custom resources in the cluster like **Kafka** and **KafkaNodePool** resources. Based on the configuration in the resource, the Strimzi Cluster Operator deploys clusters of the components described by the resources.

Specifically, `Kafka` and `KafkaNodePool` resources will deploy Kafka and ZooKeeper clusters. Optionally, if configured, the `Kafka` resource also deploys the Strimzi Entity Operator and Cruise Control. The Strimzi Entity Operator is responsible for managing other resources inside the particular Kafka cluster (topics, users, and so on), Cruise Control is used for rebalancing Kafka.

`KafkaConnect` and `KafkaConnector` resources are used to deploy Kafka Connect clusters and instances of Kafka Connect connectors.



Warning: Strimzi allows creating Kafka brokers by creating only a single `Kafka` resource. However, Cloudera Streams Messaging - Kubernetes Operator only supports creating Kafka brokers by creating `KafkaNodePool` resources. Node pools allow for more flexible deployments with easier scaling options. Moreover, certain features like rack awareness and scaling are limited without node pools. Broker creation using the `Kafka` resource only is deprecated, and results in unnecessary effort of migrating the deployment to use node pools.

Related Information

[Overview](#) | [Strimzi](#)

Licensing

Cloudera Streams Messaging - Kubernetes Operator requires a valid license to function. Licenses are made available to you together with your Cloudera credentials as part of your license and subscription agreement with Cloudera.

Licenses are registered during Cloudera Streams Messaging - Kubernetes Operator installation. Specifically, during the installation of Strimzi. The license activates the Strimzi Cluster Operator. Licenses are stored in a Kubernetes secret. Licenses can be updated at any time.

Licenses are valid for a set period of time. Once the license expires, the cluster resources you deployed will continue to run. However, reconciliation of resources will be blocked. For example, failed pods will not be restarted, scaling your clusters will not be possible. In general, the control mechanisms in place that keep resources healthy will be blocked. This leads to deployed resources breaking down over time.

Cloudera Streams Messaging - Kubernetes Operator publishes various log entries and Kubernetes events related to your licenses.

For example, if your license expires or becomes invalid due to any reason, appropriate logs and events are published notifying you that there are issues with your license.

These logs and events are published for the Strimzi Cluster Operator deployment. You can check these logs and events with the following commands.

```
kubectl events deployments/strimzi-cluster-operator --namesp  
ace [***NAMESPACE***]
```

```
kubectl logs deployment/strimzi-cluster-operator --names  
pace [***NAMESPACE***]
```

Related Information

[Updating a license](#)

Sizing and performance considerations

Learn about ways you can size your deployment for optimal performance.

Kafka broker performance primarily depends on the IO bandwidth of the nodes and disks. Because of this, Cloudera recommends using SSDs with high IOPS and throughput for large workloads. JBOD can also lead to throughput improvements when the node IO bandwidth can support multiple disks.

When optimizing for large workloads, using HDDs and storage replication services such as Longhorn might add a significant performance overhead.

Depending on the characteristics of the workload, brokers might require a large memory pool to be able to serve fetch requests from the cache. Brokers might also require an increased CPU allocation to support compressed messages

ZooKeeper requires a small resource pool in most workloads, and scaling the cluster to more than three nodes usually provides no benefit.

Recommended minimum setup

Cloudera recommends the following cluster sizing as a baseline for small and medium workloads.

Container	Count	CPU (m) per Pod	Memory (MiB) per Pod	Notes
Strimzi Cluster Operator	2	1000	384	Required.
Kafka Broker	3	8000	20480	Required for Kafka workloads.
ZooKeeper	3	4000	4096	Required for Kafka workloads.
Cruise Control	1	4000	4096	Required for rebalance operations.
Topic Operator	1	500	256	Required if you want to manage topics with <code>KafkaTopic</code> resources.
User Operator	1	500	256	Required if you want to manage Kafka users with <code>KafkaUser</code> resources.
Kafka Exporter	1	500	256	Required if you want to have additional broker and client metrics available.
Kafka Connect	3	4000	4096	Required if you want to use Kafka Connect and related functionality.