Red Hat Integration 2020.Q1 Installing Debezium on RHEL

TECHNOLOGY PREVIEW - For use with Debezium 1.0 on Red Hat Enterprise Linux (RHEL)
Abstract

This guide describes how to install Red Hat Debezium on RHEL with AMQ Streams.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREFACE</td>
<td>3</td>
</tr>
<tr>
<td>CHAPTER 1. DEBEZIUM OVERVIEW</td>
<td>4</td>
</tr>
<tr>
<td>1. DOCUMENT CONVENTIONS</td>
<td>4</td>
</tr>
<tr>
<td>CHAPTER 2. INSTALLING DEBEZIUM CONNECTORS</td>
<td>5</td>
</tr>
<tr>
<td>2.1. PREREQUISITES</td>
<td>5</td>
</tr>
<tr>
<td>2.2. KAFKA TOPIC CREATION RECOMMENDATIONS</td>
<td>5</td>
</tr>
<tr>
<td>2.3. DEPLOYING DEBEZIUM WITH AMQ STREAMS ON RHEL</td>
<td>6</td>
</tr>
<tr>
<td>APPENDIX A. USING YOUR SUBSCRIPTION</td>
<td>8</td>
</tr>
<tr>
<td>Accessing Your Account</td>
<td>8</td>
</tr>
<tr>
<td>Activating a Subscription</td>
<td>8</td>
</tr>
<tr>
<td>Downloading Zip and Tar Files</td>
<td>8</td>
</tr>
</tbody>
</table>
PREFACE

IMPORTANT

Technology Preview features are not supported with Red Hat production service-level agreements (SLAs) and might not be functionally complete; therefore, Red Hat does not recommend implementing any Technology Preview features in production environments. This Technology Preview feature provides early access to upcoming product innovations, enabling you to test functionality and provide feedback during the development process. For more information about support scope, see Technology Preview Features Support Scope.
CHAPTER 1. DEBEZIUM OVERVIEW

Red Hat Debezium is a distributed platform that monitors databases and creates change event streams. Red Hat Debezium is built on Apache Kafka and is deployed and integrated with AMQ Streams.

Debezium captures row-level changes to a database table and passes corresponding change events to AMQ Streams. Applications can read these change event streams and access the change events in the order in which they occurred.

Debezium has multiple uses, including:

- Data replication
- Updating caches and search indexes
- Simplifying monolithic applications
- Data integration
- Enabling streaming queries

Debezium provides connectors (based on Kafka Connect) for the following common databases:

- MySQL
- PostgreSQL
- SQL Server
- MongoDB

NOTE

This guide refers to Debezium documentation. Debezium is the upstream community project for Debezium.

1.1. DOCUMENT CONVENTIONS

Replaceables

In this document, replaceable text is styled in monospace and italics.

For example, in the following code, you will want to replace `my-namespace` with the name of your namespace:

```
sed -i 's/namespace: .*/namespace: my-namespace/' install/cluster-operator/*RoleBinding*.yaml
```
CHAPTER 2. INSTALLING DEBEZIUM CONNECTORS

Install Debezium connectors through AMQ Streams by extending Kafka Connect with connector plugins. Following a deployment of AMQ Streams, you can deploy Debezium as a connector configuration through Kafka Connect.

2.1. PREREQUISITES

A Debezium installation requires the following:

- Red Hat Enterprise Linux version 7.x or 8.x with an x86_64 architecture.
- Administrative privileges (sudo access).
- AMQ Streams 1.4 on Red Hat Enterprise Linux is installed on the host machine.
  - AMQ Streams must be running in one of the supported JVM versions.
- Credentials for the kafka user that was created when AMQ Streams was installed.
- An AMQ Streams cluster is running.
  - For instructions on running a basic, non-production AMQ Streams cluster containing a single ZooKeeper and a single Kafka node, see Running a single node AMQ Streams cluster.

NOTE

If you have an earlier version of AMQ Streams, you need to upgrade to AMQ Streams 1.4. For upgrade instructions, see AMQ Streams and Kafka upgrades.

Additional resources

- For more information about how to install AMQ Streams, see Installing AMQ Streams.

2.2. KAFKA TOPIC CREATION RECOMMENDATIONS

Debezium uses multiple Kafka topics for storing data. The topics have to be either created by an administrator, or by Kafka itself by enabling auto-creation for topics using the auto.create.topics.enable broker configuration.

The following list describes limitations and recommendations to consider when creating topics:

Database history topics (for MySQL and SQL Server connectors)

- Infinite (or very long retention).
- Replication factor of at least 3 in production.
- Single partition.

Other topics

- Optionally, log compaction enabled (if you wish to only keep the last change event for a given record).
  In this case, the min.compaction.lag.ms and delete.retention.ms topic-level settings in
Apache Kafka should be configured so that consumers have enough time to receive all events and delete markers. Specifically, these values should be larger than the maximum downtime you anticipate for the sink connectors (for example, when you update them).

- Replicated in production.
- Single partition.
  
  You can relax the single partition rule, but your application must handle out-of-order events for different rows in the database (events for a single row are still totally ordered). If multiple partitions are used, Kafka will determine the partition by hashing the key by default. Other partition strategies require using Simple Message Transforms (SMTs) to set the partition number for each record.

## 2.3. DEPLOYING DEBEZIUM WITH AMQ STREAMS ON RHEL

This procedure describes how to set up connectors for Debezium on Red Hat Enterprise Linux. Connectors are deployed to an AMQ Streams cluster using Kafka Connect, a framework for streaming data between Apache Kafka and external systems. Kafka Connect must be run in distributed mode rather than standalone mode.

This procedure assumes that AMQ Streams is installed and ZooKeeper and Kafka are running.

### Procedure

1. Visit the [Red Hat Integration download site](https://subscription.services.redhat.com) on the Red Hat Customer Portal and download the Debezium connector or connectors that you want to use. For example, download the Debezium 1.0.0 MySQL Connector to use Debezium with a MySQL database.

2. In `/opt/kafka`, create the `connector-plugins` directory if not already created for other Kafka Connect plugins:
   
   ```shell
   $ sudo mkdir /opt/kafka/connector-plugins
   
   $ sudo unzip debezium-connector-mysql-1.0.0-plugin.zip -d /opt/kafka/connector-plugins
   
   $ su - kafka
   $ Password:
   $ jcmd | grep ConnectDistributed
   18514 org.apache.kafka.connect.cli.ConnectDistributed /opt/kafka/config/connect-distributed.properties
   $ kill 18514
   ```
7. Edit the `connect-distributed.properties` file in `/opt/kafka/config/` and specify the location of the Debezium connector:

```
plugin.path=/opt/kafka/connector-plugins
```

8. Run Kafka Connect in distributed mode:

```
$ /opt/kafka/bin/connect-distributed.sh /opt/kafka/config/connect-distributed.properties
```

Kafka Connect runs. During startup, Debezium connectors are loaded from the `connector-plugins` directory.

9. Repeat steps 6–8 for each Kafka Connect worker node.

Additional resources

- Kafka Connect in distributed mode
- Adding connector plugins

Updating Kafka Connect

If you need to update your deployment, amend the Debezium connector JAR files in the `/opt/kafka/connector-plugins` directory, and then restart Kafka Connect.

Next Steps

The Debezium User Guide describes how to configure each connector and its source database for change data capture. Once configured, a connector will connect to the source database and produce events for each inserted, updated, and deleted row or document.
APPENDIX A. USING YOUR SUBSCRIPTION

Debezium is provided through a software subscription. To manage your subscriptions, access your account at the Red Hat Customer Portal.

**Accessing Your Account**

1. Go to access.redhat.com.
2. If you do not already have an account, create one.
3. Log in to your account.

**Activating a Subscription**

1. Go to access.redhat.com.
2. Navigate to My Subscriptions.
3. Navigate to Activate a subscription and enter your 16-digit activation number.

**Downloading Zip and Tar Files**

To access zip or tar files, use the customer portal to find the relevant files for download. If you are using RPM packages, this step is not required.

1. Open a browser and log in to the Red Hat Customer Portal Product Downloads page at access.redhat.com/downloads.
2. Scroll down to INTEGRATION AND AUTOMATION.
3. Click Red Hat Integration to display the Red Hat Integration downloads page.
4. Click the Download link for your component.

*Revised on 2021-02-19 08:56:17 UTC*