Cloudera Runtime 7.1.1

Securing Streams Replication Manager

Date published: 2019-09-13
Date modified: 2020-05-22

https://docs.cloudera.com/

Legal Notice

Contents

Streams Replication Manager security overview

Configuring SRM security involves enabling and setting security-related features and properties for the SRM service and the srm-control command line tool. This permits SRM to access source and target clusters and replicate data between them. There are multiple methods you can use to configure security, which methods you use will depend on your cluster environment.

Streams Replication Manager (SRM) can replicate data between secured and unsecured environments, as well as between environments that have differing security setups.

You configure security by providing the SRM service (driver and service roles) as well as the srm-control command line tool with the required keys, certificates, and credentials needed to access the clusters that replication is happening between. This can be done by setting security-related configuration properties. There are three methods you can use to configure security properties. These are the following:

- Configure security with the SSL/TLS and Kerberos feature toggles available in Cloudera Manager
- Configure security with the Streams Replication Manager's Replication Configs Cloudera Manager property
- Configure security with environment variables

Which configuration methods you use depends on your cluster environment. The following sections provide an overview of each method. In addition to reviewing the following sections, Cloudera recommends that you also review the security configuration examples to gain a better understanding of how you can set up security for your environment.

Configure security with the SSL/TLS and Kerberos feature toggles available in Cloudera Manager

This method can only be used to configure security for the SRM service (driver and service roles). This method does not configure security for the srm-control tool.

You can use the Enable Kerberos Authentication, Enable TLS/SSL for SRM Driver, and Enable TLS/SSL for SRM Service properties to enable or disable TLS/SSL and Kerberos for SRM. These properties can be configured in Cloudera Manager under Streams Replication Manager Configuration . Setting these properties will give the SRM service access to the clusters and Kafka services that are managed by the same Cloudera Manager as SRM. It will not give access to clusters that are managed by a different instance of Cloudera Manager. As a result, using only this method to configure security for the SRM service can be applied if the following requirements are met:

Note: If the requirements are met, Cloudera recommends that you use this method of configuration.

- The Kafka and SRM services are all managed by the same instance of Cloudera Manager.
- All Kafka services that replication is happening between have identical security configurations.
- Auto-TLS is turned on, or TLS is set up in accordance with Cloudera recommendations.

If the Kafka services have differing security configurations, or are managed by different Cloudera Manager instances, you need to also use the Streams Replication Manager's Replication Configs property to manually set security properties. In other words, if the above requirements are not all met, using only this method of configuration is not sufficient.

Configure security with the Streams Replication Manager's Replication Configs Cloudera Manager property

This method of configuration can be used to configure security for the SRM service (driver and service roles), as well as the srm-control tool.

The Streams Replication Manager's Replication Configs property is used to configure properties that SRM accepts, but are not directly available for configuration in Cloudera Manager. For more information about its usage in general, see Configuring Properties Not Exposed in Cloudera Manager.

You can use Streams Replication Manager's Replication Configs to set security-related properties. Depending on your environment, it is likely that you need to add security properties that are prefixed with each cluster's alias. Prefixing security properties with aliases is needed when the clusters SRM is connecting to require different security configurations. Prefixing properties enables you to specify different security configurations for each cluster. Based on these prefixed properties SRM will know what configuration to use when accessing each cluster.

For example, if you have two clusters with the aliases primary and secondary that both have SSL enabled, you need to add the following to the Streams Replication Manager's Replication Configs property:

```
primary.security.protocol = SSL
primary.ssl.keystore.location = [KEYSTORE_PATH]
primary.ssl.keystore.password = [PASSWORD]
primary.ssl.key.password = [PASSWORD]
primary.ssl.truststore.location = [TRUSTSTORE_PATH]
primary.ssl.truststore.password = [PASSWORD]

secondary.security.protocol = SSL
secondary.ssl.keystore.location = [KEYSTORE_PATH]
secondary.ssl.keystore.password = [PASSWORD]
secondary.ssl.key.password = [PASSWORD]
secondary.ssl.truststore.location = [TRUSTSTORE_PATH]
secondary.ssl.truststore.password = [PASSWORD]
```

It is also possible to add non-prefixed security properties, SRM will accept these as well. However, in a case like this, it will use that configuration for all clusters that it is connecting to.

Lastly, it is also worth highlighting that out of the three security configuration methods, this is the only one that configures both the SRM service and srm-control tool.

Configure security with environment variables

This method is used to configure the srm-control tool.

This method of configuration is used in scenarios when you do not want to use the Streams Replication Manager's Replication Configs property to configure security properties for the srm-control tool. In a scenario like this, you can use the following environment variables to specify security properties.

Note: Cloudera recommends that you store security-related environment variables in a protected file and directory, and source the file when the security properties need to be set.

Environment Variable	Corresponding Property
security_protocol	security.protocol
ssl_keystore_location	ssl.keystore.location
ssl_keystore_password	ssl.keystore.password
ssl_key_password	ssl.key.password
ssl_truststore_location	ssl.truststore.location
ssl_truststore_location	ssl.truststore.location
ssl_truststore_password	ssl.truststore.password
SRM_KERBEROS_OPTS	Used to set the JAAS configuration file that should be used. In addition, it can also be used to set other Java system properties related to Kerberos. For example: SRM_KERBEROS_OPTS="-Djava.security.auth.login.config=/opt/streams-replication-manager/conf/srm-jaas.conf"

Related Information

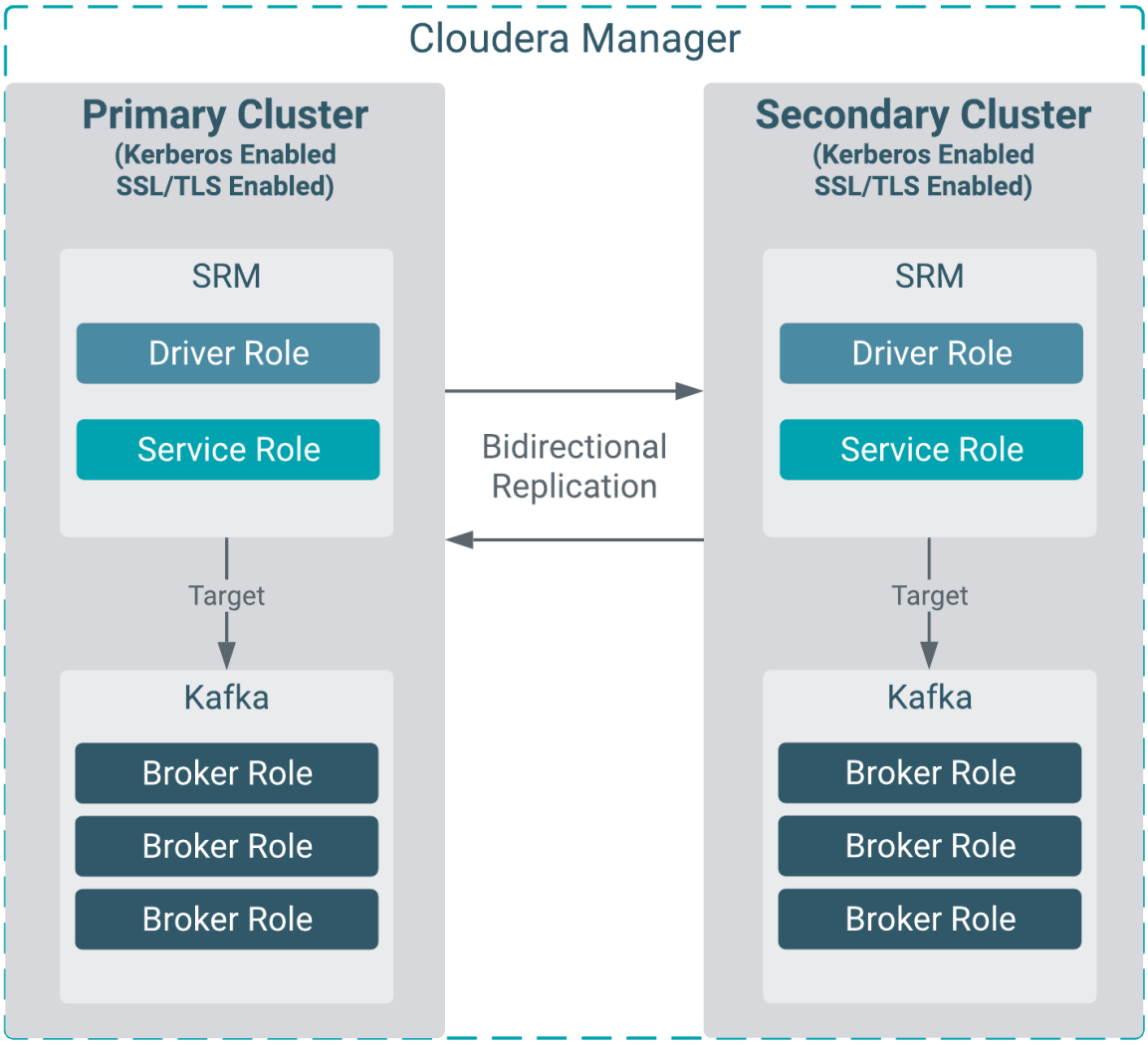
Configuring Properties Not Exposed in Cloudera Manager

SRM security example for a cluster environment managed by a single Cloudera Manager instance

To run SRM in a secure environment with two clusters that have identical security configuration and are managed by the same Cloudera Manager, you must configure security for the SRM service and the srm-control command line tool on both clusters.

About this task

Consider the following replication environment.



There are two clusters, each with a Kafka and a Streams Replication Manager (SRM) service deployed on it. Both clusters are managed by the same Cloudera Manager. The clusters have an identical security setup. TLS/SSL, Kerberos, and Auto-TLS are enabled on both clusters. In addition, the same Kerberos KDC and trusted CA is used

by both clusters. Data replication is bidirectional. All SRM driver and service roles target the same cluster and Kafka service they are deployed in.

The following example walks you through the steps to configure security for this replication environment. There are two steps you need to complete. You have to configure security for the SRM service, as well as the srm-control command line tool on both clusters. In this scenario, the SRM service is configured with Cloudera Manager. The srm-control tool can be configured either with Cloudera Manager or environment variables.

Note: The following steps cover security configuration only, other configuration, such as setting up cluster aliases or replications is not covered.

Procedure

1. Configure security for the SRM service.

   Because data replication is bidirectional, this step needs to be completed for both primary and secondary clusters.

   a) In Cloudera Manager, select the Streams Replication Manager service.
   b) Go to Configuration.
   c) Find and enable the following properties:

   - Enable Kerberos Authentication
   - Enable TLS/SSL for SRM Driver
   - Enable TLS/SSL for SRM Service

   d) Click Save Changes.
   e) Restart Streams Replication Manager.

2. Configure security for the srm-control tool.

   In addition to configuring the SRM service, you also need to configure security for the srm-control tool. Without configuration, the tool will not be able to run and you will not be able to kick off replication. There are two methods you can use to configure security for the tool, either with Cloudera Manager or with environment variables. Choose one of the following methods:

   a) Using Cloudera Manager

      Complete this step for all SRM services.

      1. In Cloudera Manager select Streams Replication Manager.
      2. Go to Configuration.
      3. Find the Streams Replication Manager's Replication Configs property.
      4. Add and configure the required security properties.

         In the case of this example, properties related to both SSL/TLS and Kerberos need to be added.

         Warning: Passwords entered into the following properties are stored in plaintext.

         ```
         security.protocol = SASL_SSL
         ssl.truststore.location = /path/to/truststore.jks
         ssl.truststore.password = test1234
         ssl.keystore.location = /path/to/keystore.jks
         ssl.keystore.password = test1234
         ssl.key.password = test1234
         sasl.kerberos.service.name = kafka
         sasl.mechanism = GSSAPI
         sasl.jaas.config = com.sun.security.auth.module.Krb5LoginModule requ
         ired useKeyTab=true keyTab="/path/to/keytab file" storeKey=true useT
         icketCache=false principal="streamsrepmgr@STREAMANALYTICS.COM";
         ```

      5. Click Save Changes.
      6. Restart Streams Replication Manager.

b) Using environment variables

Complete this step for all SRM driver hosts.

1. Log in to the SRM driver host.
2. Configure security properties with environment variables.

   In the case of this example, properties related to both SSL/TLS and Kerberos need to be added. As a result, all available environment variables need to be set. These are the following:

   Note: Instead of running the export commands individually, Cloudera recommends that you store them in a protected file and directory, and source the file when the security properties need to be set.

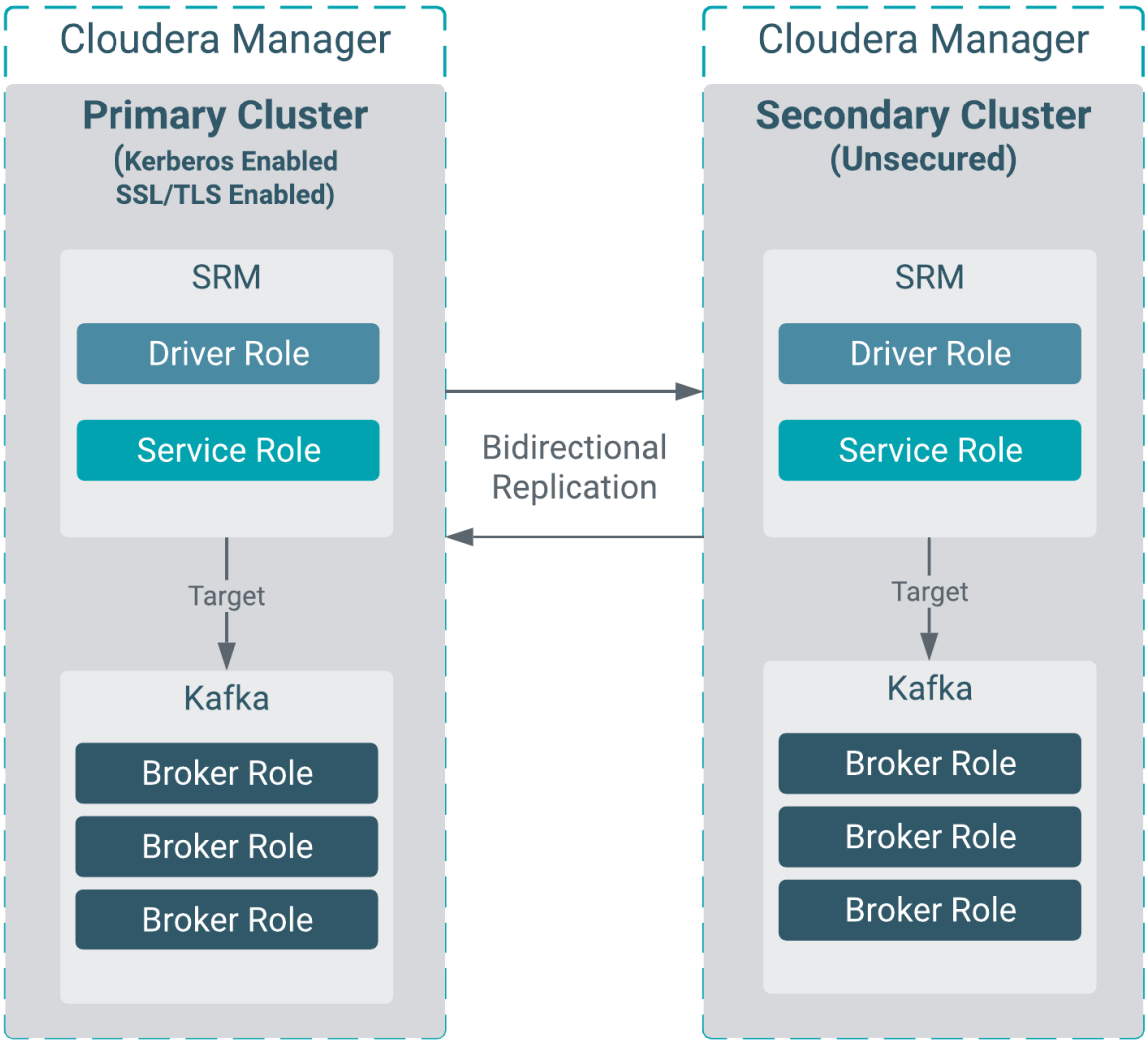
```
export SRM_KERBEROS_OPTS="-Djavax.security.auth.useSubjectCredsOnly=
false -Djava.security.auth.login.config=/path/to/jaas.conf"
export security_protocol=SASL_SSL
export ssl_truststore_location=/path/to/truststore.jks
export ssl_truststore_password=password123
export ssl_keystore_location=/path/to/keystore.jks
export ssl_keystore_password=password123
export ssl_key_password=password123
```

SRM security example for a cluster environment managed by multiple Cloudera Manager instances

To run SRM in a secure environment with two clusters that have differing security setups and are managed by different instances of Cloudera Manager, you must configure security properties using Cloudera Manager in both clusters manually.

About this task

Consider the following replication environment.



There are two clusters, primary and secondary. Each managed by a unique instance of Cloudera Manager. Both clusters have a Kafka and a Streams Replication Manager (SRM) service deployed on it. The clusters and Kafka services have differing security setups. In the primary cluster TLS/SSL and Kerberos are enabled. The secondary cluster is unsecured. All SRM driver and service roles target the same cluster and Kafka service they are deployed in. Data replication is bidirectional.

The following example walks you through the steps to configure security for this replication environment. In this scenario, configuration of both the primary and secondary clusters is required. Both clusters are configured using Cloudera Manager.

Note: The following steps cover security configuration only, other configuration, such as setting up cluster aliases or replications is not covered.

Procedure

1. Configure the primary cluster.

   In the primary cluster you need to enable security feature toggles and configure additional security properties with the Streams Replication Manager's Replication Configs property.

   a) In the primary cluster's Cloudera Manager, select the Streams Replication Manager service.
   b) Go to Configuration.
   c) Enable TLS/SSL and Kerberos.

      In environments where the clusters have differing security setups, the configuration of the SSL/TLS and Kerberos feature toggles should match the security of the cluster that the SRM driver is targeting. In this example, the SRM driver in the primary cluster is targeting the primary cluster. Because the primary cluster has both SSL/TLS and Kerberos enabled, all feature toggles have to be enabled. You can do this by enabling the following properties:

      - Enable Kerberos Authentication
      - Enable TLS/SSL for SRM Driver
      - Enable TLS/SSL for SRM Service

   d) Find the Streams Replication Manager's Replication Configs property.
   e) Add the required prefixed security properties.

      Warning: Passwords entered into the following properties are stored in plaintext.

      ```
      primary.security.protocol = SASL_SSL
      primary.ssl.truststore.location = /path/to/truststore.jks
      primary.ssl.truststore.password = test1234
      primary.ssl.keystore.location = /path/to/keystore.jks
      primary.ssl.keystore.password = test1234
      primary.ssl.key.password = test1234
      primary.sasl.kerberos.service.name = kafka
      primary.sasl.mechanism = GSSAPI
      primary.sasl.jaas.config = com.sun.security.auth.module.Krb5LoginModule
      required useKeyTab=true keyTab="/path/to/keytab file" storeKey=true useT
      icketCache=false principal="streamsrepmgr@STREAMANALYTICS.COM";
      secondary.security.protocol = PLAINTEXT
      ```

   f) Click Save Changes.
   g) Restart Streams Replication Manager.

2. Configure the secondary cluster.

   Although the secondary cluster is unsecured, configuration is still required. The security properties of the primary cluster need to be specified. Otherwise, the SRM instance running in this cluster will not be able to connect to the primary cluster. Additionally, the security protocol for the secondary cluster needs to be set. Otherwise, you will not be able to initiate the srm-control tool on any of the secondary cluster hosts.

   a) In the secondary cluster's Cloudera Manager, select the Streams Replication Manager service.
   b) Go to Configuration.
   c) Find the Streams Replication Manager's Replication Configs property.
   d) Add the required prefixed security properties.

      Note: The truststore, keystore, and keytab files you specify must contain the necessary keys and certificates that allow access to the primary cluster.

      Warning: Passwords entered into the following properties are stored in plaintext.

      ```
      primary.security.protocol = SASL_SSL
      primary.ssl.truststore.location = /path/to/truststore.jks
      ```

```
primary.ssl.truststore.password = test1234
primary.ssl.keystore.location = /path/to/keystore.jks
primary.ssl.keystore.password = test1234
primary.ssl.key.password = test1234
primary.sasl.kerberos.service.name = kafka
primary.sasl.mechanism = GSSAPI
primary.sasl.jaas.config = com.sun.security.auth.module.Krb5LoginModule
required useKeyTab=true keyTab="/path/to/keytab file" storeKey=true useT
icketCache=false principal="streamsrepmgr@STREAMANALYTICS.COM";
secondary.security.protocol = PLAINTEXT
```

e) Click Save Changes.

f) Restart Streams Replication Manager.