MarkLogic

MarkLogic Kafka Connector

Changes for version 1.2.2

Biju George
7-15-2020

Contents

Summary of this document

This document describes the changes made to the connector version 1.2.2. Below are the changes / capabilities added

New Capabilities

1. Mutual Authentication with MarkLogic server

Examples of Sink Configurations

JsonConverter without SSL

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = 192.168.0.29
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = false
ml.connection.ssl = false
ml.connection.certFile = \
ml.connection.certPassword = \
ml.connection.externalName = \
ml.connection.hostNameVerifier = \
ml.connection.tlsVersion = \
ml.connection.mutualAuth = false
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-nossl
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/nossl/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

JsonConverter with Simple SSL

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = 192.168.0.29
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = true
ml.connection.ssl = false
ml.connection.certFile = \
ml.connection.certPassword = \
ml.connection.externalName = \
ml.connection.hostNameVerifier = \
ml.connection.tlsVersion = \
ml.connection.mutualAuth = false
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-simplessl
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/simplessl/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

JsonConverter with 1way Mutual Authentication and STRICT HostName verification

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = myvm-host1
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = false
ml.connection.ssl = true
ml.connection.certFile = \
ml.connection.certPassword = \
ml.connection.externalName = \
ml.connection.hostNameVerifier = STRICT
ml.connection.tlsVersion = TLSv1.2
ml.connection.mutualAuth = false
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-1wayauth-strict
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/1wayauth/strict/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

JsonConverter with 2way Mutual Authentication and STRICT HostName verification

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = myvm-host1
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = false
ml.connection.ssl = true
ml.connection.certFile = /tmp/srportal.p12
ml.connection.certPassword = abc
ml.connection.externalName = \
ml.connection.hostNameVerifier = STRICT
ml.connection.tlsVersion = TLSv1.2
ml.connection.mutualAuth = true
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-mutualauth-strict
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/mutualauth/strict/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

JsonConverter with 2way Mutual Authentication and ANY HostName verification

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = 192.168.0.29
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = false
ml.connection.ssl = true
ml.connection.certFile = /tmp/srportal.p12
ml.connection.certPassword = abc
ml.connection.externalName = \
ml.connection.hostNameVerifier = ANY
ml.connection.tlsVersion = TLSv1.2
ml.connection.mutualAuth = true
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-mutualauth-any
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/mutualauth/any/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

JsonConverter with Certificate Authentication and ANY HostName verification

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = myvm-host1
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = CERTIFICATE
ml.connection.type = DIRECT
ml.connection.simpleSsl = false
ml.connection.ssl = true
ml.connection.certFile = /tmp/srportal.p12
ml.connection.certPassword = abc
ml.connection.externalName = \
ml.connection.hostNameVerifier = ANY
ml.connection.tlsVersion = TLSv1.2
ml.connection.mutualAuth = false
ml.connection.username = \
ml.connection.password = \
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-certauth
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/certauth/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

JsonConverter with Certificate Authentication and STRICT HostName Verification

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.json.JsonConverter
value.converter = org.apache.kafka.connect.json.JsonConverter
topics = perf-queue
ml.connection.host = 192.168.0.29
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = CERTIFICATE
ml.connection.type = DIRECT
ml.connection.simpleSsl = false
ml.connection.ssl = true
ml.connection.certFile = /tmp/srportal.p12
ml.connection.certPassword = abc
ml.connection.externalName = \
ml.connection.hostNameVerifier = STRICT
ml.connection.tlsVersion = TLSv1.2
ml.connection.mutualAuth = false
ml.connection.username = \
ml.connection.password = \
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-certauth
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/certauth/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

String Converter with simpleSSL

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = org.apache.kafka.connect.storage.StringConverter
value.converter = org.apache.kafka.connect.storage.StringConverter
topics = perf-queue
ml.connection.host = myvm-host1
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = true
ml.connection.ssl = false
ml.connection.certFile = \
ml.connection.certPassword = \
ml.connection.externalName = \
ml.connection.hostNameVerifier = \
ml.connection.tlsVersion = \
ml.connection.mutualAuth = false
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-simplessl-string-conv
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/simplessl/string/
ml.document.uriSuffix = .json
key.converter.schemas.enable = false
value.converter.schemas.enable = false
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

AvroConverter with simpleSSL

```
name = MarkLogicSink
connector.class = com.marklogic.kafka.connect.sink.MarkLogicSinkConnector
tasks.max = 1
key.converter = io.confluent.connect.avro.AvroConverter
value.converter = io.confluent.connect.avro.AvroConverter
topics = perf-queue
ml.connection.host = myvm-host1
ml.connection.port = 8071
ml.connection.database = CertServer
ml.connection.securityContextType = BASIC
ml.connection.username = srportal
ml.connection.password = abc
ml.connection.type = DIRECT
ml.connection.simpleSsl = true
ml.connection.ssl = false
ml.connection.certFile = \
ml.connection.certPassword = \
ml.connection.externalName = \
ml.connection.hostNameVerifier = \
ml.connection.tlsVersion = \
ml.connection.mutualAuth = false
ml.dmsdk.batchSize = 1
ml.dmsdk.threadCount = 1
ml.dmsdk.transform = \
ml.dmsdk.transformParams = \
ml.document.collections = coll-simplessl-avro-conv
ml.document.format = JSON
ml.document.mimeType = \
ml.document.permissions = rest-reader,read,rest-writer,update
ml.document.uriPrefix = /kafka-data/simplessl/avro/
ml.document.uriSuffix = .json
key.converter.schemas.enable = true
key.converter.schema.registry.url = http://localhost:8081
value.converter.schemas.enable = true
value.converter.schema.registry.url = http://localhost:8081
errors.tolerance = all
errors.log.enable = false
errors.deadletterqueue.topic.name = test-dlq
errors.deadletterqueue.context.headers.enable = true
```

Customising Security options

1. Security options: The sink connector supports BASIC/DIGEST/CERTIFICATE authentication with both 1waySSL and 2waySSL. The Host verification supported are ANY, STRICT and COMMON. If a custom verifier or any other security related changes are required, then `buildDatabaseClientConfig` method of the interface `com.marklogic.kafka.connect.DatabaseClientConfigBuilder` should be changed or newly implemented. The connector can be considered as any other Java client for implementing the security changes. Below are the security related sink properties and how to update those properties. The properties newly introduced in Sink version 1.2.2 is highlighted in **bold**

Property	Description
`ml.connection.security ContextType`	The security context of the MarkLogic target App Server. Possible values are BASIC, DIGEST, CERTIFICATE
`ml.connection.username`	The username to connect to MarkLogic.
`ml.connection.password`	The password to connect to MarkLogic. This does not have effect with CERTIFICATE authentication
`ml.connection.type`	Configure as GATEWAY if the endpoint is a load balancer. Otherwise configure as DIRECT
`ml.connection.simpleSs l`	Configure as 'true' if a simple SSL connection is used. No Host Verification is done when configured like this. This might be enough for development, but not for production.
`ml.connection.ssl`	If stricter SSL is what is configured in the App server, configure as 'true'. SimpleSsl should be 'false' if ml.connection.ssl is configured as 'true. Refer the example sink configurations.
`ml.connection.certFile` `ml.connection.certPass word` `ml.connection.external Name`	Applicable for CERTIFICATE authentication only. The path to the user certificate in PKCS12 format, certificate password and external name to be provided. For details of certificate based authentication please see below documentation. https://docs.marklogic.com/guide/security/security-based-auth
`ml.connection.hostName Verifier`	Possible values are ANY, STRICT or COMMON. This is to configure the strictness of HostName verification. Refer below documentation for additional details. Default behavior is of 'ANY', if not configured. http://docs.marklogic.com/javadoc/client/index.html?com/marklogic/client/DatabaseClientFactory.SSL HostnameVerifier.HostnameVerifierAdapter.html
`ml.connection.tlsVersi on`	The TLS version. TLSv1.2 would be most appropriate value. However, these values can be configured if you have a lower TLS configuration. TLSv1, TLSv1.1, TLSv1.2
`ml.connection.mutualAu th`	If a 2waySSL is required for BASIC and DIGEST authentication, configure this as true. If true, the client certificate should be supplied. Hence, ml.connection.certFile & ml.connection.certPassword also need to be configured.

===