Cloudera DataFlow for Data Hub 7.3.2

Cloudera DataFlow for Data Hub Release Notes

Date published: 2019-12-16
Date modified: 2024-12-11

https://docs.cloudera.com/

Legal Notice

Contents

What's New in Cloudera DataFlow for Data Hub 7.3.2

Cloudera DataFlow for Data Hub 7.3.2 includes components for Flow Management, Streaming Analytics, and Streams Messaging Data Hub clusters. Learn about the new features and improvements available for these components.

What's new in Flow Management

Learn about the new features available when using NiFi 2 in Flow Management Data Hub clusters running on Cloudera on cloud 7.3.2.

Cloudera 7.3.2 is a platform-level release defining the base environment that can be used to create different Data Hub cluster types including Flow Management. From 7.3.2 onwards, Flow Management Data Hub clusters are only available with Apache NiFi 2.

Cloudera Flow Management 4.12.0 is built on Apache NiFi 2.6.0 and adds several new features not available in previous versions of Apache NiFi. It also includes fixes addressing multiple Common Vulnerabilities and Exposures (CVEs) to enhance stability, reliability, and compliance for enterprise deployments.

The following sections summarize the most important updates in this release.

New NiFi components

- ConsumeKinesis - Enables users to receive messages from an AWS Kinesis source
- PutIcebergRecord - Enables users to write record-based FlowFiles into Iceberg tables
- AzureDevOpsFlowRegistryClient - Enables users to leverage Azure DevOps as a Flow Registry

For a full list of supported NiFi components, see the Support Matrix.

Improvements

Industrial Internet of Things (IIoT) support

ConsumeMQTTIIoT and MQTTIIoTReader allows flows to receive and parse Sparkplug messages, enabling IIoT/edge use cases as an industry standard format.

Kafka OAuth2 authentication support

Support for OAuth2 (OAUTHBEARER) authentication has been added for Kafka components. This enhancement enables secure access to Kafka brokers using an OAuth2 Token Provider controller service to obtain and manage access tokens.

The feature is available for the following Kafka component groups:

- Kafka_2_6 / Kafka2CDP
- Kafka3ConnectionService

New flow analysis rule

RequireServerSSLContext Flow Analysis Rule has been added to ensure that servers (using ListenHttp or ListenTCP) cannot be created without a secure communications layer.

Migration Tool

Cloudera provides a Flow Migration Tool that helps you replace deprecated processors, update configurations, and handle breaking changes automatically in your data flows.

Cloudera Flow Migration Tool 7.0.0 adds support for migrations from Cloudera Flow Management 2.2.9.700 to 4.12.0, and includes several bug fixes.

Note: Version 7.0.0 is not yet available and will be released soon.

For more information, see the Migration Tool documentation documentation.

What's new in Edge Management [Technical Preview]

Learn about the Technical Preview for Light Duty Edge Flow Management cluster definitions available in Cloudera DataFlow for Data Hub 7.3.2 in Cloudera on cloud.

These cluster definitions provide the full functionality of Cloudera Edge Management 2.3.1.0, along with an improved user experience and enhanced capabilities for streamlined management and integration.

Edge Flow Manager integrates with Cloudera User Management, making it easier to manage users and groups. For more information about this integration, see After creating your cluster and Managing user groups using LDAP.

What's New in Streams Messaging

Learn about the new Streams Messaging features in Cloudera DataFlow for Data Hub 7.3.2.

Cloudera DataFlow for Data Hub 7.3.2 introduces new Streams Messaging features and includes all service packs and cumulative hotfixes from Cloudera Runtime 7.3.1.100 through 7.3.1.706. For a comprehensive record of all Streams Messaging updates in Cloudera Runtime 7.3.1.x, see New Features.

What's New in Apache Kafka

New features and functional updates for Kafka are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Rebase on Kafka 3.9

Kafka shipped with this version of Cloudera Runtime is based on Apache Kafka 3.9.1 (previously 3.4.1). For more information, see the following resources:

- Notable changes for releases 3.5.0 through 3.9.1: Upgrading | Apache Kafka.
- The Apache Kafka release notes for the following versions:
  - Kafka 3.5.0
  - Kafka 3.5.1
  - Kafka 3.6.0
  - Kafka 3.6.1
  - Kafka 3.5.2
  - Kafka 3.7.0
  - Kafka 3.6.2
  - Kafka 3.7.1
  - Kafka 3.8.0
  - Kafka 3.8.1
  - Kafka 3.9.0
  - Kafka 3.9.1
- The Apache Kafka release announcements: Release Announcements | Apache Kafka

KRaft is generally available and ZooKeeper is deprecated

KRaft (Kafka Raft) is generally available. KRaft is from now on the recommended metadata management mode for Kafka in Cloudera. Additionally, migrating existing ZooKeeper-based Kafka clusters to use KRaft is now possible.

With the general availability of KRaft, deploying new or using existing Kafka clusters running in ZooKeeper mode is deprecated. Additionally, support for ZooKeeper-based Kafka clusters will be removed in a future release.

Cloudera recommends the following:

- Deploy all new Kafka clusters in KRaft mode.
- Migrate existing ZooKeeper-based clusters to KRaft following an upgrade to Cloudera Runtime 7.3.2.

  This is the only version where migration is possible. Neither previous or future major, minor, and maintenance versions support migration.

  Important: Migrating Kafka from ZooKeeper to KRaft is in Technical Preview for Cloudera Runtime 7.3.2.0 in Cloudera on cloud. The migration creates a three node setup for KRaft, which cannot be scaled.

For additional information, see the following resources:

- Setting up your Streams Messaging cluster
- Kafka KRaft
- Migrating Kafka from ZooKeeper to KRaft overview

Kafka protocol and metadata version is set automatically during upgrades

When upgrading Kafka, Cloudera Manager now automatically sets the inter.broker.protocol.version property for ZooKeeper-based clusters and the metadata.version property for KRaft-based clusters. You no longer need to manually set these properties to the current protocol or metadata version before an upgrade. This feature is only available when upgrading to Cloudera Runtime 7.3.2 or higher.

After the upgrade, clearing these properties remains a manual task. However, in Cloudera Runtime 7.3.2 and higher, both inter.broker.protocol.version and metadata.version are now available for direct configuration in Cloudera Manager Kafka Configuration . The label names of the properties are Kafka Inter-Broker Protocol Version and Kafka Metadata Version. This means you can set or clear these properties directly from the UI, without needing to use advanced configuration snippets.

Connector-level offset flush control

A new connector-level property, cloudera.offset.flush.interval.ms, is added. Use this property to override the Kafka Connect role-level Offset Flush Interval (offset.flush.interval.ms) property. Overriding enables you to control the interval at which connector task offsets are committed on a per-connector basis.

Configure cloudera.offset.flush.interval.ms in connectors that need a different offset flush interval than the role default. This is commonly useful for connectors where the interval controls how often data is flushed to target systems, for example NiFiStatelessSink, HDFSSink, and S3Sink.

IPv6 support for Kafka

Starting with the 7.3.2 release, Kafka supports IPv6 with dual-stack functionality, allowing seamless communication over both IPv4 and IPv6 networks. This capability improves network scalability, future-proofs deployments, and enhances overall platform security.

Offline Log Directories chart

A new default chart, **Offline Log Directories**, is added for Kafka in Cloudera Manager. This chart can help you quickly identify and track storage issues on your brokers. It is available by default for the Kafka service as well as for individual Kafka Broker role instances.

The chart shows offline log directories and their mount paths for Kafka brokers. A non-zero value indicates an active error state for a specific log directory, while a value of 0 means the directory was in an error state during the selected timeframe but is now healthy. The chart only displays log directories that had errors during the selected timeframe.

New actions for collecting Kafka diagnostic data

The following new service-specific actions are available for collecting Kafka diagnostic data in Cloudera Manager:

- Collect Kafka Cluster Diagnostics - gathers detailed cluster-wide data, including topics, configurations, consumer groups, and more.
- Describe Kafka Topics - provides detailed information about all Kafka topics.

These actions are available in the Actions dropdown on the Kafka service and Kafka Broker role instance pages. Diagnostic data is printed to sdtout for immediate access and also saved as a compressed archive on the host where the action runs.

For more information, seeCollecting Kafka diagnostic data using Cloudera Manager actions Connect.

Debezium connectors upgraded from 1.9.8.Final to 3.3.1.Final

This release of Cloudera Runtime ships version 3.3.1.Final of the following Debezium connectors:

- MySQL
- PostgreSQL
- Oracle
- SQL Server
- Db2

Existing connector instances are automatically upgraded to the new version as part of a cluster upgrade. However, you will be required to make configuration updates before you can upgrade your cluster. Critical changes that affect all Debezium connectors are summarized below.

Important: Debezium 3.3.1.Final includes many major breaking changes compared to 1.9.8.Final. Depending on your connector configuration and use case, additional configuration updates not highlighted here might be necessary. For a comprehensive list of changes, review release notes on https://debezium.io/.

- Property renaming (configuration namespace changes)

  New, more consistent namespaces for configuration properties are introduced. The old database.* prefixes have been removed. Connector configuration keys collected in the following table must be updated before an upgrade.

Old Property Prefix (Debezium 1.9)	New Property Prefix (Debezium 3.3)
database.server.name	topic.prefix
database.history.*	schema.history.internal.*
database.* (JDBC pass-through)	driver.*
database.dbname (SQL Server)	database.names

- Database driver version requirements are updated

  The recommended and supported JDBC driver versions used by the majority of connectors has changed. The following table collects the JDBC drivers you will need to deploy on your cluster before an upgrade.

  Note: The recommended driver version for Debezium Db2 remains unchanged and is still 11.5.0.0 but is listed in the following table as a reference.

Component	New Driver Version / Notes
MySQL	9.1.0
PostgreSQL	42.7.7

Component	New Driver Version / Notes
Oracle	21.x, 23.x — use a Java 11+ Oracle JDBC driver (ojdbc11.jar)
SQL Server	12.4.2.jre8
Db2	11.5.0.0

For more information, see Getting started with upgrades for Cloudera on cloud.

What's New in Schema Registry

New features and functional updates for Schema Registry are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

There are no new features in this release.

What's New in Streams Messaging Manager

New features and functional updates for Streams Messaging Manager are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

There are no new features in this release.

What's New in Streams Replication Manager

New features and functional updates for Streams Replication Manager are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Reverse Checkpointing

Streams Replication Manager now supports reverse checkpointing. This feature enables the tracking and replication of consumer offsets from a target cluster back to a source cluster. By tracking offsets in the reverse direction, you ensure that the progress made by consumer groups on a backup cluster is preserved and translated back to the primary cluster during a failback scenario.

Reverse checkpointing minimizes message duplication upon failback by mapping the offsets from the replica topic back to the equivalent offsets in the source topic. To enable this feature, you must configure the following in Cloudera Manager:

- Set the cloudera.reverse.checkpointing.enabled property to true.
- Enable bidirectional replication in the Streams Replication Manager's Replication Configs property.

In addition to service configurations, you must use the srm-control tool to explicitly allowlist topics for reverse checkpointing using the reverse-checkpointed-topics command. Consumer group replication must also be enabled in both directions.

Important: Reverse checkpointing relies on cluster prefixes to identify replica topics and is therefore not compatible with the IdentityReplicationPolicy.

Single REST server for all replication flows

Streams Replication Manager now uses a single REST server with a single port to handle inter-worker communication for all replication flows. Previously, a dedicated REST server was started for each replication flow. The new implementation exposes only the endpoints required for inter-worker coordination and task configuration updates. These endpoints are

restricted to inter-worker communication and cannot be accessed externally. The legacy per-flow REST server implementation is deprecated in 7.3.2 and will be removed in a future release. Cloudera recommends that you migrate your Streams Replication Manager clusters to the new implementation.

Suppressing internal metrics topics

You can now configure the Streams Replication Manager Service to suppress the eager creation of srm-metrics topics for all possible replication flows. This prevents the creation of unused topics. To enable this behavior, set the metrics.topic.creation.for.possible.flows.enabled property to false.

Configurable timeout for Streams Application Kafka Connection Health Test

A new SRM Service Streams Application Connection Test Timeout (streams.replication.manager.service.streams.application.connection.test.timeout) Cloudera Manager configuration option is now available for the Streams Replication Manager Service. It sets the timeout, in seconds, for the Streams Application Kafka Connection Health Test, which periodically checks connectivity to the target Kafka cluster. The default is 1 second.

What's New in Cruise Control

New features and functional updates for Cruise Control are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

New configuration parameter for controlling IP stack preference

A new cc.additional.java.options configuration parameter is available on the Cruise Control configuration page in Cloudera Manager. The default value sets the IP protocol to IPv4.

New intra.broker.goals configuration for Cruise Control

Cloudera Manager introduces a new intra.broker.goals configuration for Cruise Control. The default value includes com.linkedin.kafka.cruisecontrol.analyzer.goals.IntraBrokerDiskCapacityGoal and com.linkedin.kafka.cruisecontrol.analyzer.goals.IntraBrokerDiskUsageDistributionGoal.

This has an effect on the existing Default Goals (default.goals) configuration, which must be a subset of Supported Goals and Supported Intra Broker Goals.

Additionally, the intra.broker.goals configuration no longer needs to be defined in an advanced configuration snippet if done previously.

What's New in Cloudera Streaming Analytics

Learn about the new Cloudera Streaming Analytics features in Cloudera Data Flow for Cloudera Data Hub 7.3.2.

The following features and changes apply to Cloudera Streaming Analytics 1.16.0 in Cloudera Data Flow for Cloudera Data Hub 7.3.2:

Cloudera platform support

Cloudera Streaming Analytics 1.16.0 is supported on Cloudera 7.3.1.500 (SP3) with Cloudera Data Hub. Review the 7.3.1.500 (SP3) Release Notes and Support Matrix to understand which operating system, database, and JDK versions are supported for Cloudera Streaming Analytics as well.

Framework and UI upgrades

The underlying frameworks for Cloudera SQL Stream Builder are upgraded to improve platform stability and resolve vulnerabilities.

Enhanced Kafka Data Source configurations

You can specify custom properties for Kafka connections directly in the Kafka Data Source form. Additionally, you can explicitly define the truststore and keystore types (JKS, PKCS12, or BCFKS) when configuring a secured Kafka cluster.

Removal of JavaScript UDFs

All JavaScript User-Defined Function (UDF) features have been removed from Cloudera Streaming Analytics. Cloudera recommends rewriting your existing UDFs in Python before upgrading to Cloudera Streaming Analytics 1.16.0.

For more information about working with Python UDFs, see Adding Python UDFs.

Component support in Cloudera DataFlow for Data Hub 7.3.2

Cloudera DataFlow for Data Hub 7.3.2 includes the following components.

Flow Management clusters

Cloudera 7.3.2 is a platform-level release defining the base environment used to create and run different Data Hub cluster types including Flow Management. From 7.3.2 onwards, Flow Management Data Hub clusters are only available with Apache NiFi 2.

For the Flow Management cluster type, Cloudera provides specific NiFi-based component bundles through Cloudera Flow Management versions. These are aligned with the platform version and include a specific NiFi version used in the cluster:

Flow Management clusters with NiFi 2

Platform version: 7.3.2.000

Cloudera Flow Management version: 4.12.0

Components:

- Apache NiFi 2.6.0.4.12.0
- Apache NiFi Registry 2.6.0.4.12.0
- Schema Registry 0.10.0.7.3.1.0

Streams Messaging clusters

- Apache Kafka 3.4.0
- Schema Registry 0.10.0
- Streams Messaging Manager 2.3.0
- Streams Replication Manager 1.1.0
- Cruise Control 2.5.85

Streaming Analytics clusters

- Apache Flink 1.20.1

Cloudera Flow Management support matrix

Flow Management clusters running Apache NiFi 2 include a broad set of NiFi components, most of which are fully supported by Cloudera. To ensure stability and full support, avoid using unsupported components in production environments.

Supported NiFi processors

Review the list of Apache NiFi processors supported in Cloudera Flow Management 4.12.0.

Apache NiFi provides a wide range of processors for building dataflows. Cloudera Flow Management 4.12.0 is based on Apache NiFi 2.6.0 and includes most of these processors, but not all are supported by Cloudera.

Some processors available in Apache NiFi are developed and maintained by the community and are not supported in Cloudera. These processors may be excluded due to factors such as limited testing, insufficient reliability, lack of production readiness, or misalignment with Cloudera best practices.

To ensure stability, performance, and full support coverage, use only processors that are officially supported by Cloudera in production environments.

Table 1: List of NiFi processors supported in Cloudera Flow Management 4.12.0

AttributeRollingWindow	GetAwsTranslateJobStatus
AttributesToCSV	GetAzureEventHub
AttributesToJSON	GetAzureQueueStorage_v12
CalculateParquetOffsets	GetBoxFileCollaborators
CalculateParquetRowGroupOffsets	GetBoxGroupMembers
CalculateRecordStats	GetCouchbase
CaptureChangeDebeziumDB2	GetDynamoDB
CaptureChangeDebeziumMongoDB	GetElasticsearch
CaptureChangeDebeziumMySQL	GetFile
CaptureChangeDebeziumOracle	GetFileResource
CaptureChangeDebeziumPostgreSQL	GetFTP
CaptureChangeDebeziumSQLServer	GetGcpVisionAnnotateFilesOperationStatus
CaptureChangeMySQL	GetGcpVisionAnnotateImagesOperationStatus
ChunkData	GetHBase
ChunkDocument	GetHDFS
CompressContent	GetHDFSEvents
ConnectWebSocket	GetHDFSFileInfo
ConsumeAMQP	GetHDFSSequenceFile
ConsumeAzureEventHub	GetHubSpot
ConsumeBoxEnterpriseEvents	GetJiraIssue
ConsumeBoxEvents	GetMongo
ConsumeElasticsearch	GetMongoRecord
ConsumeGCPubSub	GetS3ObjectMetadata
ConsumeIMAP	GetS3ObjectTags
ConsumeJMS	GetSFTP
ConsumeKafka	GetShopify
ConsumeKafka_2_6	GetSlackReaction
ConsumeKafka2CDP	GetSmbFile
ConsumeKafka2RecordCDP	GetSNMP
ConsumeKafkaRecord_2_6	GetSnowflakeIngestStatus

ConsumeKinesisStream	GetSolr
ConsumeMQTT	GetSplunk
ConsumeMQTTIIoT	GetSQS
ConsumePLC	GetTCP
ConsumePOP3	GetWorkdayReport
ConsumeSlack	GetZendesk
ConsumeTwitter	HandleHttpRequest
ConsumeWindowsEventLog	HandleHttpResponse
ControlRate	IdentifyMimeType
ConvertAvroToParquet	InsertToMilvus
ConvertCharacterSet	InvokeGRPC
ConvertProtobuf	InvokeHTTP
ConvertRecord	InvokeScriptedProcessor
CopyAzureBlobStorage_v12	ISPEnrichIP
CopyS3Object	JoinEnrichment
CountText	JoltTransformJSON
CreateBoxFileMetadataInstance	JoltTransformRecord
CreateBoxMetadataTemplate	JSLTTransformJSON
CreateHadoopSequenceFile	JsonQueryElasticsearch
CryptographicHashContent	LexicalQueryMilvus
DebugFlow	ListAzureBlobStorage_v12
DecryptContentAge	ListAzureDataLakeStorage
DecryptContentPGP	ListBoxFile
DeduplicateRecord	ListBoxFileInfo
DeleteAzureBlobStorage_v12	ListBoxFileMetadataInstances
DeleteAzureDataLakeStorage	ListBoxFileMetadataTemplates
DeleteBoxFileMetadataInstance	ListCDPObjectStore
DeleteByQueryElasticsearch	ListDatabaseTables
DeleteCDPObjectStore	ListDropbox
DeleteDynamoDB	ListenBeats
DeleteFile	ListenFTP
DeleteGCSObject	ListenGRPC
DeleteGridFS	ListenHTTP
DeleteHBaseCells	ListenNetFlow
DeleteHBaseRow	ListenOTLP
DeleteHDFS	ListenSlack
DeleteMongo	ListenSyslog
DeleteS3Object	ListenTCP
DeleteSFTP	ListenTrapSNMP
DeleteSQS	ListenUDP

DetectDuplicate	ListenUDPRecord
DistributeLoad	ListenWebSocket
DuplicateFlowFile	ListFile
EmbedData	ListFTP
EncodeContent	ListGCSBucket
EncryptContentAge	ListGoogleDrive
EncryptContentPGP	ListHBaseRegions
EnforceOrder	ListHDFS
EvaluateJsonPath	ListS3
EvaluateXPath	ListSFTP
EvaluateXQuery	ListSmb
ExecuteGraphQuery	LogAttribute
ExecuteGraphQueryRecord	LogMessage
ExecuteGroovyScript	LookupAttribute
ExecuteProcess	LookupRecord
ExecuteScript	MergeContent
ExecuteSparkInteractive	MergeRecord
ExecuteSQL	ModifyBytes
ExecuteSQLRecord	ModifyCompression
ExecuteStreamCommand	MonitorActivity
ExtractAvroMetadata	MoveAzureDataLakeStorage
ExtractDocumentText	MoveHDFS
ExtractEmailAttachments	Notify
ExtractEmailHeaders	PackageFlowFile
ExtractGrok	PaginatedJsonQueryElasticsearch
ExtractHL7Attributes	ParseDocument
ExtractImageMetadata	ParseEvtx
ExtractMediaMetadata	ParseNetflowv5
ExtractRecordSchema	ParseSyslog
ExtractStructuredBoxFileMetadata	ParseSyslog5424
ExtractText	PartitionCsv
FetchAzureBlobStorage_v12	PartitionDocx
FetchAzureDataLakeStorage	PartitionHtml
FetchBoxFile	PartitionPdf
FetchBoxFileInfo	PartitionRecord
FetchBoxFileMetadataInstance	PartitionText
FetchBoxFileRepresentation	PromptAzureOpenAI
FetchCDPObjectStore	PromptBedrock
FetchDistributedMapCache	PromptChatGPT
FetchDropbox	PromptClaude

FetchFile	PromptOpenAI
FetchFTP	PublishAMQP
FetchGCSObject	PublishGCPubSub
FetchGoogleDrive	PublishJMS
FetchGridFS	PublishKafka
FetchHBaseRow	PublishKafka_2_6
FetchHDFS	PublishKafka2CDP
FetchParquet	PublishKafka2RecordCDP
FetchPLC	PublishKafkaRecord_2_6
FetchS3Object	PublishMQTT
FetchSFTP	PublishSlack
FetchSmb	PutAccumuloRecord
FilterAttribute	PutAzureBlobStorage_v12
FlattenJson	PutAzureCosmosDBRecord
ForkEnrichment	PutAzureDataExplorer
ForkRecord	PutAzureDataLakeStorage
GenerateFlowFile	PutAzureEventHub
GenerateRecord	PutAzureQueueStorage_v12
GenerateTableFetch	PutBigQuery
GeoEnrichIP	PutBoxFile
GeoEnrichIPRecord	PutCassandraQL
GeohashRecord	PutCassandraRecord
GetAsanaObject	PutCDPObjectStore
GetAwsPollyJobStatus	PutChroma
GetAwsTextractJobStatus	PutClouderaHiveQL
GetAwsTranscribeJobStatus	PutClouderaHiveStreaming

Supported NiFi controller services

Review the list of Apache NiFi controller services supported in Cloudera Flow Management 4.12.0.

Apache NiFi provides a wide range of controller services for building dataflows. Cloudera Flow Management 4.12.0 is based on Apache NiFi 2.6.0 and includes most of these controller services, but not all are supported by Cloudera.

Some controller services available in Apache NiFi are developed and maintained by the community and are not supported in Cloudera. These controller services may be excluded due to factors such as limited testing, insufficient reliability, lack of production readiness, or misalignment with Cloudera best practices.

To ensure stability, performance, and full support coverage, use only controller services that are officially supported by Cloudera in production environments.

Table 2: List of NiFi controller services supported in Cloudera Flow Management 4.12.0

AccumuloService	JsonRecordSetWriter
ActiveMQJMSConnectionFactoryProvider	JsonTreeReader
ADLSCredentialsControllerService	JWTBearerOAuth2AccessTokenProvider

ADLSCredentialsControllerServiceLookup	Kafka3ConnectionService
ADLSIDBrokerCloudCredentialsProviderControllerService	KafkaRecordSink_2_6
AmazonGlueEncodedSchemaReferenceReader	KafkaRecordSinkCDP
AmazonGlueSchemaRegistry	KerberosKeytabUserService
AmazonMSKConnectionService	KerberosPasswordUserService
ApicurioSchemaRegistry	KerberosTicketCacheUserService
AvroReader	KuduLookupService
AvroRecordSetWriter	LivySessionController
AvroSchemaRegistry	LoggingRecordSink
AWSCredentialsProviderControllerService	MapCacheClientService
AWSIDBrokerCloudCredentialsProviderControllerService	MapCacheServer
AzureBlobIDBrokerCloudCredentialsProviderControllerService	MongoDBControllerService
AzureBlobStorageFileResourceService	MongoDBLookupService
AzureCosmosDBClientService	MQTTIIoTReader
AzureDataLakeStorageFileResourceService	Neo4JCypherClientService
AzureEventHubRecordSink	ParquetReader
AzureServiceBusJMSConnectionFactoryProvider	ParquetRecordSetWriter
AzureStorageCredentialsControllerService_v12	PEMEncodedSSLContextProvider
AzureStorageCredentialsControllerServiceLookup_v12	PhoenixThickConnectionPool
CassandraDistributedMapCache	PhoenixThinConnectionPool
CassandraSessionProvider	PostgreSQLConnectionPool
CdpCredentialsProviderControllerService	PropertiesFileLookupService
CdpOauth2AccessTokenProviderControllerService	ProtobufReader
CEFReader	ProxyPLC4XConnectionPool
CiscoEmblemSyslogMessageReader	RabbitMQJMSConnectionFactoryProvider
ClouderaAttributeSchemaReferenceReader	ReaderLookup
ClouderaAttributeSchemaReferenceWriter	RecordSetWriterLookup
ClouderaEncodedSchemaReferenceReader	RecordSinkServiceLookup
ClouderaEncodedSchemaReferenceWriter	RedisConnectionPoolService
ClouderaHiveConnectionPool	RedisDistributedMapCacheClientService
ClouderaHiveConnectionPoolLookup	RedshiftConnectionPool
ClouderaSchemaRegistry	RESTCatalogService
CMLLookupService	RestLookupService
ConfluentEncodedSchemaReferenceReader	S3FileResourceService
ConfluentEncodedSchemaReferenceWriter	ScriptedLookupService
ConfluentProtobufMessageNameResolver	ScriptedReader
ConfluentSchemaRegistry	ScriptedRecordSetWriter
CouchbaseKeyValueLookupService	ScriptedRecordSink
CouchbaseMapCacheClient	SetCacheClientService
CouchbaseRecordLookupService	SetCacheServer

CSVReader	SimpleCsvFileLookupService
CSVRecordLookupService	SimpleDatabaseLookupService
CSVRecordSetWriter	SimpleKeyValueLookupService
DatabaseRecordLookupService	SimpleRedisDistributedMapCacheClientServic
DatabaseRecordSink	SimpleScriptedLookupService
DatabaseTableSchemaRegistry	SiteToSiteReportingRecordSink
DBCPConnectionPool	SlackRecordSink
DBCPConnectionPoolLookup	SmbjClientProviderService
DeveloperBoxClientService	SnowflakeComputingConnectionPool
DistributedMapCacheLookupService	StandardAsanaClientProviderService
EBCDICRecordReader	StandardAzureCredentialsControllerService
ElasticSearchClientServiceImpl	StandardCouchbaseConnectionService
ElasticSearchLookupService	StandardDatabaseDialectService
ElasticSearchStringLookupService	StandardDropboxCredentialService
EmailRecordSink	StandardFileResourceService
EmbeddedHazelcastCacheManager	StandardHashiCorpVaultClientService
ExcelReader	StandardHttpContextMap
ExternalHazelcastCacheManager	StandardJiraCredentialService
FreeFormTextRecordSetWriter	StandardJsonSchemaRegistry
GCPCredentialsControllerService	StandardKustoIngestService
GCSFileResourceService	StandardKustoQueryService
GenericPLC4XConnectionPool	StandardOauth2AccessTokenProvider
GrokReader	StandardPGPPrivateKeyService
HadoopCatalogService	StandardPGPPublicKeyService
HadoopDBCPConnectionPool	StandardPLC4XConnectionPool
HazelcastMapCacheClient	StandardPrivateKeyService
HBase_2_ClientMapCacheService	StandardProtobufReader
HBase_2_ClientService	StandardProxyConfigurationService
HBase_2_RecordLookupService	StandardRestrictedSSLContextService
HikariCPConnectionPool	StandardS3EncryptionService
HiveCatalogService	StandardSnowflakeIngestManagerProviderServ
HttpRecordSink	StandardSSLContextService
ImpalaConnectionPool	StandardWebClientServiceProvider
IPFIXReader	Syslog5424Reader
IPLookupService	SyslogReader
JASN1Reader	TinkerpopClientService
JdbcCatalogService	UDPEventRecordSink
JettyWebSocketClient	VolatileSchemaCache
JettyWebSocketServer	WindowsEventLogReader
JiraRecordSink	XMLFileLookupService

JMSConnectionFactoryProvider	XMLReader
JndiJmsConnectionFactoryProvider	XMLRecordSetWriter
JsonConfigBasedBoxClientService	YamlTreeReader
JsonPathReader	ZendeskRecordSink

Supported NiFi reporting tasks

Review the list of Apache NiFi reporting tasks supported in Cloudera Flow Management 4.12.0.

Cloudera Flow Management is based on Apache NiFi and includes a set of reporting tasks, most of which are supported by Cloudera. To ensure optimal performance and reliable support, it is crucial to use only supported reporting tasks and avoid deploying unsupported ones in production environments.

- AzureLogAnalyticsProvenanceReportingTask
- AzureLogAnalyticsReportingTask
- ControllerStatusReportingTask
- MonitorDiskUsage
- MonitorMemory
- QueryNiFiReportingTask
- ReportLineageToAtlas
- ScriptedReportingTask
- SiteToSiteBulletinReportingTask
- SiteToSiteMetricsReportingTask
- SiteToSiteProvenanceReportingTask
- SiteToSiteStatusReportingTask

Additional reporting tasks are developed and tested by the community but are not officially supported by Cloudera. Reporting tasks may be excluded for various reasons, including insufficient reliability, incomplete test coverage, community declaration of non-production readiness, or deviations from Cloudera best practices.

Supported NiFi parameter providers

Review the list of Apache NiFi parameter providers supported in Cloudera Flow Management 4.12.0.

Cloudera Flow Management is shipped with Apache NiFi and includes a set of parameter providers, most of which are supported by Cloudera. To ensure optimal performance and reliable support, it is crucial to use only supported parameter providers and avoid deploying unsupported ones in production environments.

- AwsSecretsManagerParameterProvider
- AzureKeyVaultSecretsParameterProvider
- CyberArkConjurParameterProvider
- DatabaseParameterProvider
- EnvironmentVariableParameterProvider
- GcpSecretManagerParameterProvider
- HashiCorpVaultParameterProvider
- KubernetesSecretParameterProvider
- OnePasswordParameterProvider
- PropertiesFileParameterProvider

Additional parameter providers are developed and tested by the community but are not officially supported by Cloudera. Parameter providers may be excluded for various reasons, including insufficient reliability, incomplete test coverage, community declaration of non-production readiness, or deviations from Cloudera best practices.

Supported NiFi flow analysis rules

Review the list of Apache NiFi processors supported in Cloudera Flow Management 4.12.0.

Flow Analysis Rules allow analyzing components or parts of a flow to help maintain optimal design. Rules can be set as Recommendations (informational only) or Policies (enforceable). Recommendation violations are logged but do not affect functionality, while Policy violations invalidate components until resolved.

- DisallowComponentType
- DisallowConsecutiveConnectionsWithRoundRobinLB
- DisallowDeadEnd
- DisallowDeprecatedProcessor
- DisallowExtractTextForFullContent
- RecommendRecordProcessor
- RequireHandleHttpResponseAfterHandleHttpRequest
- RequireMergeBeforePutIceberg
- RequireServerSSLContextService
- RestrictBackpressureSettings
- RestrictComponentNaming
- RestrictConcurrentTasksVsThreadPoolSizeInProcessors
- RestrictFlowFileExpiration
- RestrictProcessorConcurrency
- RestrictSchedulingForListProcessors
- RestrictThreadPoolSize
- RestrictYieldDurationForConsumeKafkaProcessors

Supported NiFi flow registry clients

Review the list of Apache NiFi flow registry clients supported in Cloudera Flow Management 4.12.0.

NiFi Flow Registry clients are components in Apache NiFi that allow it to connect to external Flow Registries, services used to store and manage versioned dataflows.

- AzureDevOpsFlowRegistryClient
- BitbucketFlowRegistryClient
- ClouderaDataFlowRegistryClient
- ClouderaFlowLibraryFlowRegistryClient
- GitHubFlowRegistryClient
- GitLabFlowRegistryClient
- NifiRegistryFlowRegistryClient

Supported Cloudera exclusive components

Review the list of Cloudera exclusive components supported in Cloudera Flow Management 4.12.0.

Cloudera Flow Management provides a set of NiFi components available only to Cloudera customers. These components provide additional functionality and are tailored to enhance the Cloudera NiFi experience. The list of these components is provided below.

Processors

- CaptureChangeDebeziumDB2
- CaptureChangeDebeziumMongoDB

- CaptureChangeDebeziumMySQL
- CaptureChangeDebeziumOracle
- CaptureChangeDebeziumPostgreSQL
- CaptureChangeDebeziumSQLServer
- ConsumeKafka2CDP
- ConsumeKafka2RecordCDP
- ConsumeMQTTIIoT
- ConsumePLC
- ConvertProtobuf
- DeleteCDPObjectStore
- FetchCDPObjectStore
- FetchPLC
- GetJiraIssue
- GetSlackReaction
- GetTCP
- InvokeGRPC
- ListCDPObjectStore
- ListenGRPC
- ListenNetFlow
- PublishKafka2CDP
- PublishKafka2RecordCDP
- PutCDPObjectStore
- PutClouderaHiveQL
- PutClouderaHiveStreaming
- PutClouderaORC
- PutIcebergCDC
- PutJiraIssue
- PutPLC
- SawmillTransformJSON
- SawmillTransformRecord
- SelectClouderaHiveQL
- TriggerClouderaHiveMetaStoreEvent
- UpdateClouderaHiveTable
- UpdateDeltaLakeTable
- UpdateJiraIssue

Controller services

- ActiveMQJMSConnectionFactoryProvider
- ADLSIDBrokerCloudCredentialsProviderControllerService
- AWSIDBrokerCloudCredentialsProviderControllerService
- AzureBlobIDBrokerCloudCredentialsProviderControllerService
- AzureServiceBusJMSConnectionFactoryProvider
- CdpCredentialsProviderControllerService
- CdpOauth2AccessTokenProviderControllerService
- CiscoEmblemSyslogMessageReader
- ClouderaAttributeSchemaReferenceReader
- ClouderaAttributeSchemaReferenceWriter
- ClouderaEncodedSchemaReferenceReader
- ClouderaEncodedSchemaReferenceWriter
- ClouderaHiveConnectionPool

- ClouderaHiveConnectionPoolLookup
- ClouderaSchemaRegistry
- CMLLookupService
- EBCDICRecordReader
- GenericPLC4XConnectionPool
- ImpalaConnectionPool
- IPFIXReader
- JiraRecordSink
- KafkaRecordSinkCDP
- MQTTIIoTReader
- PhoenixThickConnectionPool
- PhoenixThinConnectionPool
- PostgreSQLConnectionPool
- ProxyPLC4XConnectionPool
- RabbitMQJMSConnectionFactoryProvider
- RedshiftConnectionPool
- StandardJiraCredentialService
- StandardPLC4XConnectionPool

Parameter provider

- PropertiesFileParameterProvider

Flow analysis rules

- DisallowConsecutiveConnectionsWithRoundRobinLB
- DisallowDeadEnd
- DisallowDeprecatedProcessor
- DisallowExtractTextForFullContent
- RecommendRecordProcessor
- RequireHandleHttpResponseAfterHandleHttpRequest
- RequireMergeBeforePutIceberg
- RestrictComponentNaming
- RestrictConcurrentTasksVsThreadPoolSizeInProcessors
- RestrictProcessorConcurrency
- RestrictSchedulingForListProcessors
- RestrictThreadPoolSize
- RestrictYieldDurationForConsumeKafkaProcessors

Flow registry clients

- ClouderaDataFlowRegistryClient
- ClouderaFlowLibraryFlowRegistryClient

Components supported by partners

Learn about the components built, maintained, and supported by Cloudera partners, and available in Flow Management Data Hub clusters in Cloudera DataFlow for Data Hub 7.3.2.

Although Cloudera's Quality Engineering teams have added test coverage for these components, they are not officially supported by Cloudera. For assistance, contact the respective partners directly.

NiFi 1.28 processors supported by partners

- ConsumePulsar (1.18.0)
- ConsumePulsarRecord (1.18.0)
- PublishPulsar (1.18.0)
- PublishPulsarRecord (1.18.0)

NiFi 1.28 controller services supported by partners

- PulsarClientAthenzAuthenticationService (1.18.0)
- PulsarClientJwtAuthenticationService (1.18.0)
- PulsarClientOauthAuthenticationService (1.18.0)
- PulsarClientTlsAuthenticationService (1.18.0)
- StandardPulsarClientService (1.18.0)

These components can be used to push data into Apache Pulsar as well as getting data out of it. In case you have issues or questions while using these components, Cloudera recommends you to reach out to your StreamNative representative team.

Note: In Flow Management Data Hub clusters using NiFi 2, the Pulsar components are not included. You can manually download the components from a Maven repository and add them into your cluster.

Unsupported features in Cloudera DataFlow for Data Hub 7.3.2

Some features exist within Cloudera DataFlow for Data Hub 7.3.2 components, but are not supported by Cloudera.

Unsupported Flow Management features

Review the list of features that are not supported in Cloudera DataFlow for Data Hub.

The following features are developed and tested by the Cloudera community but are not officially supported by Cloudera. These features are excluded for a variety of reasons, including insufficient reliability or incomplete test case coverage, declaration of non-production readiness by the community at large, and feature deviation from Cloudera best practices. Do not use these features in your production environments.

Unsupported NiFi components

NiFi 1 custom NARs cannot be successfully loaded into NiFi 2. If your NiFi setup includes custom NARs, it is a requirement to update your dependencies to align with NiFi 2. This entails making the necessary adjustments and rebuilding your NARs using Java 21. The below components are not supported and should not be used anymore.

Processors

- AttributeRollingWindow
- ConsumeIMAP
- ConsumePOP3
- DebugFlow
- DeleteMongo
- ExecuteSparkInteractive
- ExtractDocumentText
- ExtractEmailAttachments
- ExtractEmailHeaders
- ExtractMediaMetadata
- ExtractTNEFAttachments
- GetDynamoDB

- GetHDFSEvents
- GetMongo
- GetSmbFile
- ISPEnrichIP
- ListenSMTP
- ModifyBytes
- ParseNetflowv5
- ParseSyslog5424
- PutIcebergRecord
- PutIoTDBRecord
- PutMongo
- QueryAzureDataExplorer
- QueryDNS
- QueryIoTDBRecord
- RunMongoAggregation

Controller services

- HikariCPConnectionPool
- JettyWebSocketClient
- JettyWebSocketServer
- LivySessionController
- SimpleCsvFileLookupService
- StandardKustoQueryService
- XMLFileLookupService

Reporting tasks

- AzureLogAnalyticsProvenanceReportingTask
- AzureLogAnalyticsReportingTask
- DataDogReportingTask
- StandardGangliaReporte

Related Information

Cloudera Community Forum

Unsupported Edge Management features [Technical Preview]

See the unsupported features listed in the Cloudera Edge Management documentation.

Unsupported Streams Messaging features

Some Streams Messaging features exist in Cloudera DataFlow for Data Hub 7.3.2, but are not supported by Cloudera.

Kafka

The following Kafka features are not supported in Cloudera:

- Only Java and .Net based clients are supported. Clients developed with C, C++, Python, and other languages are currently not supported.
- The Kafka default authorizer is not supported. This includes setting ACLs and all related APIs, broker functionality, and command-line tools.
- SASL/SCRAM is only supported for delegation token based authentication. It is not supported as a standalone authentication mechanism.

Schema Registry

There are no updates for this release.

Streams Messaging Manager

There are no updates for this release.

Streams Replication Manager

There are no updates for this release.

Cruise Control

There are no updates for this release.

Related Information

Cloudera Community Forum
Setting up your Streams Messaging cluster

Unsupported Cloudera Streaming Analytics features

Some Cloudera Streaming Analytics features exist in Cloudera Data Flow for Cloudera Data Hub 7.3.2, but are not supported by Cloudera.

The following features are not ready for production deployment. Cloudera encourages you to explore these features in non-production environments and provide feedback on your experiences through the *Cloudera Community Forums*.

Cloudera SQL Stream Builder

- Virtual environments for Python are not supported.

Flink

- Apache Flink batch (DataSet) API
- GPU Resource Plugin
- SQL Client
- RAZ-enabled GCP environment
- The following features are not supported in SQL and Table API:
  - HBase Table Connector
  - Old Planner
  - Non-windowed (unbounded) joins, distinct

Related Information

Cloudera Community Forum

Known Issues In Cloudera DataFlow for Data Hub 7.3.2

You must be aware of the known issues and limitations, the areas of impact, and workaround in Cloudera DataFlow for Data Hub 7.3.2.

Known issues in Cloudera Flow Management

Review the list of known issues and limitations in Cloudera DataFlow for Data Hub 4.12.0.

Kafka OAuth authentication not supported across mixed Kafka component groups

Kafka OAuth2 (OAUTHBEARER) authentication cannot be used simultaneously by components from Kafka_2_6 / Kafka2CDP and Kafka3ConnectionService. If OAuth is enabled for one group, components in the other group will fail with authentication errors.

Use OAuth authentication with only one Kafka component group in a deployment, or restart before switching between them..

Known issues in Edge Management [Technical Preview]

Learn about the known issues in Edge Management clusters, the impact or changes to the functionality, and any available workaround.

For Edge Management known issues, see the Cloudera Edge Management documentation.

Known Issues in Streams Messaging

Learn about the known issues in Streams Messaging clusters, the impact or changes to the functionality, and the workaround.

Known Issues in Apache Kafka

Known issues and technical limitations for Kafka are addressed in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

Known issues identified in 7.3.2

There are no new known issues identified in this release.

Known issues identified before 7.3.2

OPSAPS-59553: Streams Messaging Manager bootstrap server config should be updated based on Kafka's listeners

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

Streams Messaging Manager does not show any metrics for Kafka or Kafka Connect when multiple listeners are set in Kafka.

Streams Messaging Manager cannot identify multiple listeners and still points to bootstrap server using the default broker port (9093 for SASL_SSL). You need to override the bootstrap server URL by performing the following steps:

1. In Cloudera Manager, go to Streams Messaging Manager Configuration Streams Messaging Manager Rest Admin Server Advanced Configuration Snippet (Safety Valve)
2. Override bootstrap server URL (hostname:port as set in the listeners for broker) for streams-messaging-manager.yaml.
3. Save your changes.
4. Restart SMM.

KAFKA-2561: Performance degradation when SSL Is enabled

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

In some configuration scenarios, significant performance degradation can occur when SSL is enabled. The impact varies depending on your CPU, JVM version, Kafka configuration, and message size. Consumers are typically more affected than producers.

Configure brokers and clients with ssl.secure.random.implementation = SHA1PRNG. It often reduces this degradation drastically, but its effect is CPU and JVM dependent.

RANGER-3809: Idempotent Kafka producer fails to initialize due to an authorization failure

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

Kafka producers that have idempotence enabled require the Idempotent Write permission to be set on the cluster resource in Ranger. If permission is not given, the client fails to initialize and an error similar to the following is thrown:

```
org.apache.kafka.common.KafkaException: Cannot execute transacti
onal method because we are in an error state
              at org.apache.kafka.clients.producer.internals.Tr
ansactionManager.maybeFailWithError(TransactionManager.java:1125)
              at org.apache.kafka.clients.producer.internals.T
ransactionManager.maybeAddPartition(TransactionManager.java:442)
              at org.apache.kafka.clients.producer.KafkaProduce
r.doSend(KafkaProducer.java:1000)
              at org.apache.kafka.clients.producer.KafkaProduc
er.send(KafkaProducer.java:914)
              at org.apache.kafka.clients.producer.KafkaProducer
.send(KafkaProducer.java:800)
              .
              .
              .
              Caused by: org.apache.kafka.common.errors.Cluste
rAuthorizationException: Cluster authorization failed.
```

Idempotence is enabled by default for clients in Kafka 3.0.1, 3.1.1, and any version after 3.1.1. This means that any client updated to 3.0.1, 3.1.1, or any version after 3.1.1 is affected by this issue.

This issue has two workarounds, do either of the following:

- Explicitly disable idempotence for the producers. This can be done by setting enable.idempoten ce to false.
- Update your policies in Ranger and ensure that producers have Idempotent Write permission on the cluster resource.

CDPD-49304: AvroConverter does not support composite default values

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

AvroConverter cannot handle schemas containing a STRUCT type default value.

None.

CFM-3532: The Stateless NiFi Source, Stateless NiFi Sink, and HDFS Stateless Sink connectors cannot use Snappy compression

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

This issue only affects Stateless NiFi Source and Sink connectors if the connector is running a dataflow that uses a processor that uses Hadoop libraries and is configured to use Snappy compression. The HDFS Stateless Sink connector is only affected if the Compression Codec or Compression Codec for Parquet properties are set to SNAPPY.

If you are affected by this issue, errors similar to the following will be present in the logs.

```
Failed to write to HDFS due to java.lang.UnsatisfiedLinkError: o
rg.apache.hadoop.util.NativeCodeLoader.buildSupportsSnappy()
```

```
Failed to write to HDFS due to java.lang.RuntimeException: nativ
e snappy library not available: this version of libhadoop was bu
ilt without snappy support.
```

Download and deploy missing libraries.

Important: Ensure that you complete steps 1-11 on all Kafka Connect hosts. Additionally, ensure that the advanced configuration snippet in step 12 is configured for all Kafka Connect role instances.

1. Create the /opt/nativelibs directory.

   ```
   mkdir /opt/nativelibs
   ```

2. Change the owner to kafka.

   ```
   chown kafka:kafka /opt/nativelibs
   ```

3. Locate the directory containing the Hadoop native libraries and copy its contents to the directory you created.

   ```
   cp /opt/cloudera/parcels/CDH/lib/hadoop/lib/native/* /opt/na
   tivelibs
   ```

4. Verify that libsnappy.so was copied to the directory you created.
5. Remove the following from /opt/nativelibs.

   ```
   libhadoop.a
                    libhadoop.so
                    libhadoop.so.1.0.0
   ```

6. Run the following command.

   ```
   hadoop version
   ```

   The command returns the Hadoop version running in the cluster. Note down the first three digits in the version.
7. Go to https://archive.apache.org/dist/hadoop/common/ and download the Hadoop version that matches the first three digits of the version running in the cluster.

   For example, if your Hadoop version is 3.1.1.7.1.9.0-296, then you need to download Hadoop 3.1.1.
8. Extract the downloaded archive.
9. Copy the following libraries from the downloaded archive to /opt/nativelibs on the cluster host.

   ```
   libhadoop.a
                    libhadoop.so.1.0.0
   ```

   The libraries are located in hadoop-*[***VERSION***]*/lib/native.
10. Create a symlink named libhadoop.so and point it to /opt/nativelibs/libhadoop.so.1.0.0.

    ```
    ln -s /opt/nativelibs/libhadoop.so.1.0.0 /opt/nativelibs/lib
    hadoop.so
    ```

11. Change the owner of every entry within /opt/nativelibs to kafka.

```
chown -h kafka:kafka /opt/nativelibs/*
```

12. In Cloudera Manager, go to Kafka service Configuration .
13. Add the following key-value pair to Kafka Connect Environment Advanced Configuration Snippet (Safety Valve).
    - Key: LD_LIBRARY_PATH
    - Value: /opt/nativelibs
14. Click Save Changes.
15. Restart the Kafka service.

Limitations

Collection of partition level metrics may cause Cloudera Manager performance to degrade

If the Kafka service operates with a large number of partitions, collection of partition level metrics may cause Cloudera Manager performance to degrade.

If you are observing performance degradation and your cluster is operating with a high number of partitions, you can choose to disable the collection of partition level metrics.

Important: If you are using Streams Messaging Manager to monitor Kafka or Cruise Control for rebalancing Kafka partitions, be aware that both Streams Messaging Manager and Cruise Control rely on partition level metrics. If partition level metric collection is disabled, Streams Messaging Manager will not be able to display information about partitions. In addition, Cruise Control will not operate properly.

Complete the following steps to turn off the collection of partition level metrics:

1. Obtain the Kafka service name.
    a. In Cloudera Manager, Select the Kafka service.
    b. Select any available chart, and select Open in Chart Builder from the configuration icon drop-down.
    c. Find $SERVICENAME= near the top of the display.

    The Kafka service name is the value of $SERVICENAME.
2. Turn off the collection of partition level metrics.
    a. Go to Hosts Hosts Configuration .
    b. Find and configure the Cloudera Manager Agent Monitoring Advanced Configuration Snippet (Safety Valve) configuration property.

    Enter the following to turn off the collection of partition level metrics:

    ```
    [KAFKA_SERVICE_NAME]_feature_send_broker_topic_partition_ent
    ity_update_enabled=false
    ```

    Replace [KAFKA_SERVICE_NAME] with the service name of Kafka obtained in step 1. The service name should always be in lower case.
    c. Click Save Changes.

Known Issues in Schema Registry

Known issues and technical limitations for Schema Registry are addressed in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

Known issues identified in 7.3.2

There are no new known issues identified in this release.

Known issues identified before 7.3.2

CDPD-40380: Authorization checking issue when Kerberos is disabled

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

Due to an issue in Ranger, when Kerberos is disabled then it is not possible to check authorization.

1. Open Schema Registry configuration in Cloudera Manager.
2. Find the ranger.plugin.schema-registry.service.name field.
3. Replace GENERATED_RANGER_SERVICE_NAME with the actual name of the service.
4. Restart the Schema Registry service.

CDPD-49304: AvroConverter does not support composite default values

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

AvroConverter cannot handle schemas containing a STRUCT type default value.

None.

OPSAPS-70971: Schema Registry does not have permissions to use Atlas after an upgrade

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

Following an upgrade, Schema Registry might not have the required permissions in Ranger to access Atlas. As a result, Schema Registry's integration with Atlas might not function in secure clusters where Ranger authorization is enabled.

1. Access the Ranger Console (Ranger Admin web UI).
2. Click the cm_atlas resource-based service.
3. Add the schemaregistry user to the all - * policies.
4. Click Manage Service Edit Service .
5. Add the schemaregistry user to the default.policy.users property.

OPSAPS-69317: Kafka Connect Rolling Restart Check fails if SSL Client authentication is required

7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

The rolling restart action does not work in Kafka Connect when the ssl.client.auth option is set to required. The health check fails with a timeout which blocks restarting the subsequent Kafka Connect instances.

You can set ssl.client.auth to requested instead of required and initiate a rolling restart again. Alternatively, you can perform the rolling restart manually by restarting the Kafka Connect instances one-by-one and checking periodically whether the service endpoint is available before starting the next one.

Known Issues in Streams Messaging Manager

Known issues and technical limitations for Streams Messaging Manager are addressed in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

Known issues identified in 7.3.2

Alert policies based on the LEADER ELECTION PER SEC attribute do not trigger

7.3.2

If the monitored Kafka cluster is running in KRaft mode, alert policies based on the **LEADER ELECTION PER SEC** alert policy attribute will not trigger because the underlying metrics are not

available from Kafka. The attribute remains visible and selectable in Streams Messaging Manager even if the monitored cluster is running in KRaft mode.

None.

CDPD-99371: Prometheus metrics do not function properly with KRaft mode

7.3.2

If the Kafka cluster monitored by Streams Messaging Manager is running in KRaft mode and Streams Messaging Manager is configured to use Prometheus as its metrics store, alerts related to the active controller do not work as expected. In KRaft mode, the broker_activecontrollerc ount metric is reported by KRaft controllers rather than brokers. This change in metric reporting can cause aggregation issues and lead to alerts triggering indefinitely. Additionally, adding the /api/prometheus-metrics endpoint from KRaft hosts to the Prometheus configuration can cause anomalies in other alerts.

Use Cloudera Manager as the metrics store for KRaft-based Kafka clusters. Alternatively, if using Prometheus, disable all alerts based on the broker_activecontrollercount metric.

Known issues identified before 7.3.2

OPSAPS-59597: Streams Messaging Manager UI logs are not supported by Cloudera Manager

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

Cloudera Manager does not display a Log Files menu for Streams Messaging Manager UI role (and Streams Messaging Manager UI logs cannot be displayed in the Cloudera Manager UI) because the logging type used by Streams Messaging Manager UI is not supported by Cloudera Manager.

View the Streams Messaging Manager UI logs on the host.

CDPD-39313: Some numbers are not rendered properly in Streams Messaging Manager UI

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

Very large numbers can be imprecisely represented on the UI. For example, bytes larger than 8 petabytes would lose precision.

None.

OPSAPS-59553: Streams Messaging Manager bootstrap server config should be updated based on Kafka's listeners

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

Streams Messaging Manager does not show any metrics for Kafka or Kafka Connect when multiple listeners are set in Kafka.

Streams Messaging Manager cannot identify multiple listeners and still points to bootstrap server using the default broker port (9093 for SASL_SSL). You need to override bootstrap server URL (hostname:port as set in the listeners for broker). Add the bootstrap server details in Streams Messaging Manager safety valve in the following path:

1. In Cloudera Manager, go to SMMConfigurationStreams Messaging Manager Rest Admin Server Advanced Configuration Snippet (Safety Valve) for streams-messaging-manager.yaml.
2. Add the following value for bootstrap servers.

```
streams.messaging.manager.kafka.bootstrap.servers=<comma-sep
arated list of brokers>
```

3. Save your changes.
4. Restart Streams Messaging Manager.

CDPD-82560: Streams Messaging Manager known unprotected endpoint

7.2.17 and its SP and CHF releases, 7.2.18 and its SP and CHF releases, 7.3.1 and its SP and CHF releases, 7.3.2

The Streams Messaging Manager Frontend exposes the /cm-configs and /configs public endpoints without requiring authentication. These endpoints do not share any sensitive information.

None.

Known Issues in Streams Replication Manager

Known issues and technical limitations for Streams Replication Manager are addressed in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

Known issues identified in 7.3.2

There are no new known issues identified in this release.

Known issues identified before 7.3.2

CDPD-22089: Streams Replication Manager does not sync re-created source topics until the offsets have caught up with target topic

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

Messages written to topics that were deleted and re-created are not replicated until the source topic reaches the same offset as the target topic. For example, if at the time of deletion and re-creation there are a 100 messages on the source and target clusters, new messages will only get replicated once the re-created source topic has 100 messages. This leads to messages being lost.

None

CDPD-11079: Blacklisted topics appear in the list of replicated topics

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

If a topic was originally replicated but was later disallowed (blacklisted), it will still appear as a replicated topic under the /remote-topics REST API endpoint. As a result, if a call is made to this endpoint, the disallowed topic will be included in the response. Additionally, the disallowed topic will also be visible in the Streams Messaging Manager UI. However, it's Partitions and Consumer Groups will be 0, its Throughput, Replication Latency and Checkpoint Latency will show N/A.

None

CDPD-30275: Streams Replication Manager may automatically re-create deleted topics on target clusters

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

If auto.create.topics.enable is enabled, deleted topics might get automatically re-created on target clusters. This is a timing issue. It only occurs if remote topics are deleted while the replication of the topic is still ongoing.

1. Remove the topic from the topic allowlist with srm-control. For example:

```
srm-control topics --source [SOURCE_CLUSTER] --target [TARGE
T_CLUSTER] --remove [TOPIC1]
```

2. Wait until Streams Replication Manager is no longer replicating the topic.
3. Delete the remote topic in the target cluster.

CDPD-80872: Streams Replication Manager replication-records-lag is incorrect on empty partitions with non-zero end offset

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

When a replicated partition is empty and its end offset is non-zero, the replication-records-lag metric is incorrectly reported as the end offset (instead of 0 or NaN).

Wait for a new message to be written into the partition. When it gets replicated, the metric is reported correctly.

Limitations

Streams Replication Manager cannot replicate Ranger authorization policies to or from Kafka clusters

Due to a limitation in the Kafka-Ranger plugin, Streams Replication Manager cannot replicate Ranger policies to or from clusters that are configured to use Ranger for authorization. If you are using Streams Replication Manager to replicate data to or from a cluster that uses Ranger, disable authorization policy synchronization in Streams Replication Manager. This can be achieved by clearing the Sync Topic Acls Enabled (sync.topic.acls.enabled) checkbox.

Known Issues in Cruise Control

Known issues and technical limitations for Cruise Control are addressed in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

Known issues identified in 7.3.2

There are no new known issues identified in this release.

Known issues identified before 7.3.2

CDPD-44676: Rebalancing with Cruise Control does not work if the metric reporter fails to report the CPU usage metric

7.2.17 and all its SP and CHF releases, 7.2.18 and all its SP and CHF releases, 7.3.1 and all its SP and CHF releases, 7.3.2

If the CPU usage metric is not reported, the numValidWindows in Cruise Control will be 0 and proposal generation as well as partition rebalancing will not work. If this issue is present, the following message will be included in the Kafka logs:

```
WARN com.linkedin.kafka.cruisecontrol.metricsreporter.CruiseCont
rolMetricsReporter:
       [CruiseControlMetricsReporterRunner]: Failed reporting CPU
 util.
```

```
java.io.IOException: Java Virtual Machine recent CPU usage is not
 available.
```

This issue is only known to affect Kafka broker hosts that have the following specifications:

- CPU: Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20GHz
- OS: Linux 4.18.5-1.el7.elrepo.x86_64 #1 SMP Fri Aug 24 11:35:05 EDT 2018 x86_64
- Java version: 8-18

Move the broker to a different machine where the CPU is different. This can be done by moving the host to a different cluster. For more information, see Moving a Host Between Clusters

Note: Cluster nodes affected by this issue are not displayed as unhealthy.

Known Issues in Cloudera Streaming Analytics

Learn about the known issues in Cloudera Streaming Analytics clusters, the impact or changes to the functionality, and the workaround.

Cloudera SQL Stream Builder

CSA-4858 - Kerberos encryption type detection does not always work correctly for Cloudera SQL Stream Builder

Cloudera SQL Stream Builder detects no supported encryption types even though there is a list of allowed encryption types in the krb5.conf file. This causes an error when generating keytabs from the principal and password pair.

1. Run ktutil on your cluster.
2. Change the configuration with the following commands:

```
addent -password -p <username> -k 1 -e aes256-cts
wkt /tmp/new_keytab.keytab
```

3. Upload the new keytab on Streaming SQL Console.

Flink

In Cloudera Streaming Analytics, the following SQL API features are in preview:

- Match recognize
- Top-N
- Stream-Table join (without rowtime input)

Third-party dependencies upgraded in Cloudera on cloud might cause Flink jobs to fail

After upgrading Cloudera on cloud, Flink jobs might fail due to upgraded third-party dependencies. For example, this could happen with awssdk when its version changes in a newer Cloudera on cloud release such as 7.3.2.

Verify your application's dependency versions against the Cloudera-supported versions before upgrading to a newer version of Cloudera on cloud. For more information see Updating Flink job dependencies.

DataStream conversion limitations

- Converting between Tables and POJO DataStreams is currently not supported in Cloudera Streaming Analytics.
- Object arrays are not supported for Tuple conversion.
- The java.time class conversions for Tuple DataStreams are only supported by using explicit TypeInformation: LegacyInstantTypeInfo, LocalTimeTypeInfo.getInfoFor(LocalDate/LocalDateTime/LocalTime.class).
- Only java.sql.Timestamp is supported for rowtime conversion, java.time.LocalDateTime is not supported.

Kudu catalog limitations

- CREATE TABLE
  - Primary keys can only be set by the kudu.primary-key-columns property. Using the PRIMARY KEY constraint is not yet possible.
  - Range partitioning is not supported.
- When getting a table through the catalog, NOT NULL and PRIMARY KEY constraints are ignored. All columns are described as being nullable, and not being primary keys.
- Kudu tables cannot be altered through the catalog other than simply renaming them.

Schema Registry catalog limitations

- Currently, the Schema Registry catalog / format only supports reading messages with the latest enabled schema for any given Kafka topic at the time when the SQL query was compiled.
- No time-column and watermark support for Registry tables.

- No CREATE TABLE support. Schemas have to be registered directly in the SchemaRegistry to be accessible through the catalog.
- The catalog is read-only. It does not support table deletions or modifications.
- By default, it is assumed that Kafka message values contain the schema id as a prefix, because this is the default behaviour for the SchemaRegistry Kafka producer format. To consume messages with schema written in the header, the following property must be set for the Registry client: store.schema.version.id.in.header: true.

Deprecation notices in Cloudera Data Flow for Data Hub 7.3.2

Certain features and functionalities have been removed or deprecated in Cloudera Data Flow for Data Hub 7.3.2. You must review these items to understand whether you must modify your existing configuration. You can also learn about the features that will be removed or deprecated in the future release to plan for the required changes.

Terminology

Items in this section are designated as follows:

Deprecated

Technology that Cloudera is removing in a future release. Marking an item as deprecated gives you time to plan for removal in a future release.

Moving

Technology that Cloudera is moving from a future release and is making available through an alternative Cloudera offering or subscription. Marking an item as moving gives you time to plan for removal in a future release and plan for the alternative Cloudera offering or subscription for the technology.

Removed

Technology that Cloudera has removed from the product and is no longer available or supported as of this release. Take note of technology marked as removed since it can potentially affect your upgrade plans.

Deprecation notices for Streams Messaging

Certain features and functionality in Streams Messaging components are deprecated or removed in Cloudera DataFlow for Data Hub 7.3.2. You must review these changes along with the information about features that will be removed or deprecated in a future release.

Deprecation Notices for Apache Kafka

Certain features and functionality in Apache Kafka are deprecated or removed in Cloudera DataFlow for Data Hub 7.3.2. You must review these changes along with the information about features that will be removed or deprecated in a future release.

Important: The following list of deprecated and removed items is not exhaustive and only contains items that have a direct and immediate effect on Kafka in Cloudera. For a full list of deprecation and/or removals in the version Apache Kafka shipped with Cloudera Runtime and Cloudera DataFlow for Data Hub, review the *Notable Changes* as well as the *Release Notes* on https://kafka.apache.org/.

Deprecated

ZooKeeper

Deploying new or using existing Kafka clusters running in ZooKeeper mode is deprecated. Support for ZooKeeper-based Kafka clusters will be removed in a future release. Cloudera recommends the following:

- Deploy all new Kafka clusters in KRaft mode.
- Migrate existing ZooKeeper-based clusters to KRaft following an upgrade to Cloudera Runtime 7.3.2.

  This is the only version where migration is possible. Neither previous or future major, minor, and maintenance versions support migration.

  Important: Migrating Kafka from ZooKeeper to KRaft is in Technical Preview for Cloudera Runtime 7.3.2.0 in Cloudera on cloud The migration creates a three node setup for KRaft, which cannot be scaled.

  For additional information, see the following resources:

  - Setting up your Streams Messaging cluster
  - Kafka KRaft overview
  - Migrating Kafka from ZooKeeper to KRaft overview

MirrorMaker (MM1)

MirrorMaker is deprecated. Cloudera recommends that you use Streams Replication Manager instead.

--zookeeper

The --zookeeper option is only supported for the kafka-configs tool and should be only used when updating SCRAM Credential configurations. The --zookeeper option is either deprecated in or removed from other Kafka command line tools. Cloudera recommends that you use the --bootstrap-server option instead.

Deprecation Notices for Streams Replication Manager

Certain features and functionality in Streams Replication Manager are deprecated or removed in Cloudera DataFlow for Data Hub 7.3.2. You must review these changes along with the information about features that will be removed or deprecated in a future release.

Deprecated

Legacy per-flow REST server implementation

The legacy per-flow REST server implementation of Streams Replication Manager is deprecated in Cloudera Runtime 7.3.2 and will be removed in a future release. In the previous implementation, the Streams Replication Manager Driver role started a dedicated REST server for each replication flow. The new implementation uses a single REST server for all flows. Cloudera recommends that you migrate your Streams Replication Manager clusters to the new implementation as part of upgrading to 7.3.2. For migration steps, see Configuring the SRM REST server for replication.

Deprecation Notices for Cruise Control

Certain features and functionality in Cruise Control are deprecated or removed in Cloudera DataFlow for Data Hub 7.3.2. You must review these changes along with the information about features that will be removed or deprecated in a future release.

Removed

Cloudera Manager metrics reporter

Cloudera Manager metrics reporter is removed from Cloudera Runtime 7.3.2. Migrate to Cruise Control metrics reporter.

Deprecation Notices for Cloudera Streaming Analytics

Certain features and functionality are removed in Cloudera Streaming Analytics. You must review these changes before upgrading.

Removed

Removed support for JavaScript UDFs

JavaScript UDF support has been removed from Cloudera Streaming Analytics. You must migrate any legacy JavaScript UDFs to Python UDFs, which are enabled by default.

For more information, see Adding Python UDFs.

Removed Data Transformation for Kafka tables

Data Transformation for Kafka tables (also referred to as input transforms) used JavaScript and was supported only on Java Development Kit (JDK) 11 and lower versions. It has been removed from Cloudera Streaming Analytics 1.16.0 because Cloudera Streaming Analytics runs on Java 17. You must migrate any equivalent logic to Flink SQL or Python UDFs.

For more information, see Adding Python UDFs.

Cloudera SQL Stream Builder v1 REST API

The v1 REST API for Cloudera SQL Stream Builder is removed in Cloudera Streaming Analytics 1.16.0 and later releases. Migrate scripts and integrations to the v2 API.

See the Cloudera SQL Stream Builder REST API reference for v2 API details.

Fixed Issues in Cloudera DataFlow for Data Hub 7.3.2

Fixed issues represent selected issues that were previously logged through Cloudera Support, but are addressed in the current release. These issues may have been reported in previous versions within the Known Issues section; meaning they were reported by customers or identified by Cloudera Quality Engineering team.

Review the list of issues that are resolved in Cloudera DataFlow for Data Hub 7.3.2.

Fixed issues in Cloudera Flow Management

Review the list of issues resolved in Cloudera Flow Management 4.12.0.

CFM-6474: Improved schema conversion in EBCDICRecordReader for integral data types

Previously, all integral fields were converted to LONG, regardless of their size. The conversion now uses the PIC digit count to determine the appropriate type:

- PIC S9(1)–S9(9) # INTEGER
- PIC S9(10)–S9(18) # LONG

Fixed issues in Edge Management [Technical Preview]

Learn about the fixed issues in Edge Management clusters, the impact or changes to the functionality, and any available workaround.

For Edge Management fixed issues, see the Cloudera Edge Management documentation.

Fixed Issues in Streams Messaging

Review the list of Streams Messaging issues that are resolved in Cloudera DataFlow for Data Hub 7.3.2.

Cloudera DataFlow for Data Hub 7.3.2 resolves Streams Messaging issues and incorporates fixes from Cloudera Runtime 7.3.1.100 through 7.3.1.706. For a comprehensive record of all Streams Messaging fixes in Cloudera Runtime 7.3.1.x, see Fixed Issues.

Fixed Issues in Kafka

Fixed issues and resolved maintenance items for Kafka are addressed in Cloudera DataFlow for Data Hub 7.3.2 and its associated service packs.

7.3.2

OPSAPS-71679: Ranger resource-based policy creation attempted on each service restart

7.3.2.0

On each service restart, Kafka, Kafka Connect, and Schema Registry attempted to create Ranger resource-based policies even when the Ranger policy cache was already present. Resource-based policy creation is now skipped when the policy cache file exists, reducing unnecessary Ranger operations on restart.

OPSAPS-75094: Broker rolling restart checks fail on FIPS-enabled clusters with JDK 11 or higher

7.3.2.0

On FIPS-enabled clusters with JDK 11 or higher, CSD_JAVA_OPTS was not propagated to the Kafka broker rolling restart check scripts, which caused the checks to fail. The rolling restart check scripts now receive the CSD_JAVA_OPTS value on FIPS-enabled clusters.

DBZ-4990: The Debezium Db2 Source connector does not support schema evolution

7.3.2.0

This issue is fixed. For more information, see DBZ-4990.

None.

OPSAPS-69317: Kafka Connect Rolling Restart Check fails if SSL Client authentication is required

7.3.2.0

Rolling restart checks for Kafka Connect failed when mutual TLS was required (ssl.client.auth set to required), even though the service was healthy. This issue is fixed.

Fixed Issues in Schema Registry

Fixed issues and resolved maintenance items for Schema Registry are addressed in Cloudera DataFlow for Data Hub 7.3.2 and its associated service packs.

7.3.2

OPSAPS-71866: Schema Registry fails to start when schema.registry.oauth.clock.skew is set in Cloudera Manager

7.3.2.0

Schema Registry failed to start when the schema.registry.oauth.clock.skew configuration property was set in Cloudera Manager. This issue is now fixed.

OPSAPS-71679: Ranger resource-based policy creation attempted on each service restart

7.3.2.0

On each service restart, Kafka, Kafka Connect, and Schema Registry attempted to create Ranger resource-based policies even when the Ranger policy cache was already present. Resource-based

policy creation is now skipped when the policy cache file exists, reducing unnecessary Ranger operations on restart.

CDPD-73277: Schema Registry configuration with Oracle TLS fails

7.3.2.0

The TLS-related Oracle database configuration parameters no longer cause issues during the Schema Registry startup if they are set.

CDPD-94371: Schema Registry does not start with MySQL 8.4 on new installations

7.3.2.0

Fixed an issue that caused startup failure on new Schema Registry installations when using MySQL 8.4.

Fixed Issues in Streams Messaging Manager

Fixed issues and resolved maintenance items for Streams Messaging Manager are addressed in Cloudera DataFlow for Data Hub 7.3.2 and its associated service packs.

7.3.2

OPSAPS-75519: Streams Messaging Manager Rest Admin Server fails to start when the keystore or truststore password contains special characters

7.3.2

The Streams Messaging Manager Rest Admin Server did not handle special characters in keystore or truststore passwords correctly in the generated configuration, which prevented the role from starting. This issue is fixed.

OPSAPS-72814: Streams Messaging Manager used incorrect Ranger policy cache directory

7.3.2

Streams Messaging Manager used an incorrect Ranger policy cache directory setting, which led to permission errors when the Kafka Ranger plugin wrote its policy cache. This issue is fixed.

CDPD-70906: NullPointerException no longer displayed when user is not authorized to access Cruise Control

7.3.2

A message with the text NullPointerException is no longer displayed if the logged-in user is not authorized to access Cruise Control.

CDPD-74620: Data Explorer returns 500 Server Error when switching partitions

7.3.2

In the Streams Messaging Manager Data Explorer, switching between partitions now waits for the new partition's offset information before fetching data. This eliminates the 500 Server Error that occurred when the previous partition's offset was incorrectly used.

CDPD-76375: Topic column name is incorrect in the replications table

7.3.2

Previously, on the **Cluster Replications** page, in the replications table, the topic column name was **Source Topic Name** instead of Target Topic Name.

This issue is now fixed and the table name is the correct **Target Topic Name**.

CDPD-91339: Connector tasks that belong to the same worker cannot be selected individually

7.3.2

Fixed an issue where checking a checkbox in the **Tasks** section on a **Connector Profile** page sometimes selected multiple tasks. Now selecting a single task always selects a single task.

CDPD-62315: Add error boundary to catch runtime errors

7.3.2

An error page fallback is added for unexpected runtime UI errors.

CDPD-45542: Excessive decimal places displayed in metrics

7.3.2

Fixed an issue where task metric percentages in the **Tasks** section of a **Connector Profile** page displayed excessive decimal places, causing UI overflow. Metrics now display with one decimal place.

CDPD-88488: The Streams Messaging Manager server fails to start if its keystore password starts with a special character or includes quotation marks

7.3.2

Fixed an issue where using special characters in keystore passwords made the Streams Messaging Manager fail to start.

Fixed Issues in Streams Replication Manager

Fixed issues and resolved maintenance items for Streams Replication Manager are addressed in Cloudera DataFlow for Data Hub 7.3.2 and its associated service packs.

7.3.2

There are no fixed issues in this release.

Fixed Issues in Cruise Control

Fixed issues and resolved maintenance items for Cruise Control are addressed in Cloudera DataFlow for Data Hub 7.3.2 and its associated service packs.

7.3.2

There are no fixed issues in this release.

Fixed Issues in Cloudera Streaming Analytics

Review the list of Flink and Cloudera SQL Stream Builder issues that are resolved in Cloudera Data Flow for Cloudera Data Hub 7.3.2.

CSA-6208 - Truststore configuration displays incorrectly for Kafka properties

Previously, the truststore configuration for the Kafka source displayed incorrectly as a custom property in the data source user interface. This issue is now fixed.

CSA-6111 - Table cleaner drops newly created tables prematurely

Previously, the materialized view table cleaner process potentially dropped newly created tables before the associated job could start. This issue is now fixed.

CSA-5969 - Password placeholders do not support special characters

Previously, special characters within password placeholders caused system crashes during resolution. This issue is now fixed. The system now escapes these characters to prevent crashes.

CSA-5902 - Improvements to project authorization and data security

Previously, certain project permission configurations did not sufficiently restrict data access for standard users. This issue is now resolved.

CSA-5873 - Environment variables apply during data source validation

Data sources referencing Cloudera SQL Stream Builder environment variables are now handled in the validation process in the v2 API.

CSA-5743 - Query analysis handles Python user-defined functions

Query analysis now correctly instantiates Python UDFs. Queries containing Python UDFs are properly analyzed and validated, even when the UDFs are not pre-loaded outside of a dedicated user session.

CSA-5484 - Dashboard link directs to stale instance

The Apache Flink dashboard link now correctly points to the active job rather than a stale instance.

CSA-4602 - Changing existing MV filter type can't be saved

CSA-5038 - Widget is empty when added to the dashboard before initialization completes

CSA-5140 - Fix No Rows To Show message when switching from sampler to MV in dashboard preview

CSA-5024 - Polling samples feedback is on even when polling is turned off

CSA-5025 - Cursor jumps to the end after the first keystroke when using templates

CSA-5026 - Oversize widget cannot be sized down

CSA-5294 - Add job save button to job settings component

Fixed CVEs in Cloudera DataFlow for Data Hub 7.3.2

Review the list of CVEs that are resolved in Cloudera DataFlow for Data Hub 7.3.2.

Fixed CVEs in Flow Management

Review the list of Common Vulnerabilities and Exposures (CVEs) fixed in Flow Management in Cloudera DataFlow for Data Hub.

Behavioral Changes in Cloudera DataFlow for Data Hub 7.3.2

You can review the changes in certain features or functionalities of components that have resulted in a change in behavior from the previously released version to this version of Cloudera DataFlow for Data Hub 7.3.2.

Behavioral changes in Flow Management

Review the list of Flow Management behavioral changes in Cloudera Data Flow for Data Hub.

Behavioral Changes in Streams Messaging

Review the list of Streams Messaging behavioral changes in Cloudera DataFlow for Data Hub 7.3.2.

Cloudera DataFlow for Data Hub 7.3.2 introduces Streams Messaging functional adjustments and behavioral updates, and includes all service packs and cumulative hotfixes from Cloudera Runtime 7.3.1.100 through 7.3.1.706. For a comprehensive record of all Streams Messaging functional adjustments in Cloudera Runtime 7.3.1.x, see Behavioral Changes.

Behavioral Changes in Kafka

Functional adjustments and behavioral updates for Kafka are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Component-level custom Java home configuration removed

Previous behavior:

You could configure a component-specific Java home for Kafka Connect.

New behavior:

The component-level custom Java home configuration options are removed. Kafka Connect now uses the host-level java_home configuration. If you previously set a component-specific Java home for this service, verify the host-level java_home setting after upgrading.

High Watermark no longer advances when ISR is below MinISR

Previous behavior:

The High Watermark (HWM) advanced regardless of whether the in-sync replica (ISR) count was below min.insync.replicas. When producers used acks=1 or acks=0, messages were written to the leader and became consumable once the HWM advanced, even if the ISR had dropped below the minimum threshold. The min.insync.replicas setting only affected acks=all produce requests, blocking writes when ISR was insufficient, but did not prevent HWM advancement or consumer reads for acks=0/acks=1 messages.

New behavior:

The HWM no longer advances when the ISR count falls below min.insync.replicas. As a result, consumers are blocked from reading new messages in this condition, even if producers with acks =1 or acks=0 are still writing to the leader. This ensures data is only consumable when it meets the cluster's minimum durability requirements. If you use min.insync.replicas=2 or higher, you may see reduced consumer throughput when the ISR count drops below the configured minimum.

For more information, see KIP-966: Eligible Leader Replicas and KAFKA-15583.

Kafka protocol version is set automatically during upgrades

Previous behavior:

The inter.broker.protocol.version property for ZooKeeper-based clusters and the metadata.version property for KRaft-based clusters were not set automatically before an upgrade. Manually configuring these properties to the current protocol and metadata version was required before an upgrade.

New behavior:

During a cluster upgrade, Cloudera Manager now automatically sets the inter.broker.protocol.version property for ZooKeeper-based clusters and the metadata.version property for KRaft-based clusters. Manual configuration is no longer required.

Note:

Manually clearing these properties after the upgrade is still necessary. However, both properties are now available for direct configuration in Cloudera Manager. Using advanced configuration snippets is no longer required.

Behavioral Changes in Schema Registry

Functional adjustments and behavioral updates for Schema Registry are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Component-level custom Java home configuration removed

Previous behavior:

You could configure a component-specific Java home for Schema Registry.

New behavior:

The component-level custom Java home configuration options are removed. Schema Registry now uses the host-level java_home configuration. If you previously set a component-specific Java home for this service, verify the host-level java_home setting after upgrading.

Schema Registry now defaults to IPv4-only communication

The default value of the schema.registry.additional.java.options configuration parameter was updated to set the IP protocol to IPv4.

If you changed the default value of this parameter before upgrading, the new default value is not applied on upgrade. You can apply it manually after the upgrade.

Behavioral Changes in Streams Messaging Manager

Functional adjustments and behavioral updates for Streams Messaging Manager are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Component-level custom Java home configuration removed

Previous behavior:

You could configure a component-specific Java home for Streams Messaging Manager.

New behavior:

The component-level custom Java home configuration options are removed. Streams Messaging Manager now uses the host-level java_home configuration. If you previously set a component-specific Java home for this service, verify the host-level java_home setting after upgrading.

Default JMX settings changed to restrict connections to localhost

Previous behavior:

The default value of the SMM_JMX_OPTS Cloudera Manager configuration option was -Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.authenticate=false -Dcom.sun.management.jmxremote.ssl=false, which allowed unrestricted, unauthenticated JMX connections.

New behavior:

The default value of SMM_JMX_OPTS has been changed to restrict JMX connections to localhost only and enable SSL. If you previously customized SMM_JMX_OPTS, your custom value is preserved on upgrade. To revert to open JMX, update SMM_JMX_OPTS in Cloudera Manager.

Streams Messaging Manager now defaults to IPv4-only communication

A new argument was added to SMM_JVM_PERF_OPTS that sets the IP protocol to IPv4 by default.

If you changed the default value of this parameter before upgrading, the new default value is not applied on upgrade. You can apply it manually after the upgrade.

Streams Messaging Manager UI Migration to Java

The Streams Messaging Manager UI service is migrated from a NodeJS runtime to a Java-based server. This change addresses security vulnerabilities associated with NodeJS dependencies and aligns Streams Messaging Manager with the centralized dependency management of the platform.

As a result of this migration, the following changes apply:

- Runtime environment

  The Streams Messaging Manager UI service now runs on the JVM. Configuration for the runtime environment is now managed via the SMM_JAVA_OPTS environment variable.
- TLS configuration

  TLS configuration moved from OpenSSL-style parameters to standard Java JSSE configuration. New Cloudera Manager parameters manage TLS protocols and cipher suites:

  - streams.messaging.manager.ui.ssl.supportedCipherSuites
  - streams.messaging.manager.ui.ssl.excludedCipherSuites
  - streams.messaging.manager.ui.ssl.supportedProtocols
  - streams.messaging.manager.ui.ssl.excludedProtocols
- Configuration migration

  During upgrade, Cloudera Manager attempts to automatically migrate existing TLS settings (including those found in the NODE_OPTIONS environment variable within safety valves) to the new Java-based configuration parameters. However, manual verification is strongly recommended.
- Safety valves

  Any properties previously set in the Streams Messaging Manager UI Server Environment Advanced configuration Snippet (Safety Valve) using NODE_OPTIONS that are not related to TLS must be manually translated to their Java equivalents (if applicable) and set using SMM_JAVA_OPTS.

Behavioral Changes in Streams Replication Manager

Functional adjustments and behavioral updates for Streams Replication Manager are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Component-level custom Java home configuration removed

Previous behavior:

You could configure a component-specific Java home for Streams Replication Manager.

New behavior:

The component-level custom Java home configuration options are removed. Streams Replication Manager now uses the host-level java_home configuration. If you previously set a component-specific Java home for this service, verify the host-level java_home setting after upgrading.

Default JMX settings changed to restrict connections to localhost

Previous behavior:

The default value of the SRM_JMX_OPTS Cloudera Manager configuration option was -Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.authenticate=false -Dcom.sun.management.jmxremote.ssl=false, which allowed unrestricted, unauthenticated JMX connections.

New behavior:

The default value of SRM_JMX_OPTS has been changed to restrict JMX connections to localhost only and enable SSL. If you previously customized SRM_JMX_OPTS, your custom value is preserved on upgrade. To revert to open JMX, update SRM_JMX_OPTS in Cloudera Manager.

Public Kafka Connect endpoints removed from SRM REST server

The public Kafka Connect endpoints are removed from the SRM REST server. Previously, these endpoints allowed users to interact with the internal Kafka Connect cluster inside the SRM instance, including starting arbitrary connectors and modifying internal connectors. These endpoints were undocumented and not part of the official SRM API. They are no longer available.

Streams Replication Manager now defaults to IPv4-only communication

A new argument was added to SRM_JVM_PERF_OPTS that sets the IP protocol to IPv4 by default.

If you changed the default value of this parameter before upgrading, the new default value is not applied on upgrade. You can apply it manually after the upgrade.

Change in internal topic filtering logic

Summary:

The logic that identifies and filters internal topics in Streams Replication Manager has changed. This enables the replication of topics that appear to be internal but are not truly internal to Kafka and Streams Replication Manager, reducing the risk of unintentionally excluding user topics from replication.

Previous behavior:

Topics were filtered from replication by the `DefaultReplicationPolicy` and `IdentityReplicationPolicy` policies if the topic name ended with [***SEPARATOR***]internal. For example, .internal. In addition, the default deny list regex pattern was the following:

```
.*[\\-\\.]internal, .*\\.replica, __.*
```

New behavior:

Topics are now filtered from replication by the `DefaultReplicationPolicy` and `IdentityReplicationPolicy` policies if:

- Name starts with mm2 and ends with [***SEPARATOR***]internal
- Name ends with [***SEPARATOR***]checkpoints[***SEPARATOR***]internal

In addition, the default deny list regex pattern is now the following:

```
mm2.*\\.internal, .*\\.replica, __.*
```

Internal Kafka topics that start with a dot (.) or two underscores (__) are continued to be filtered from replication by default.

Note: The `ReplicationPolicy` interface also changed. Previously, topics were filtered from replication if their names ended with .internal or -internal. From now on, names ending with -internal are no longer filtered. In addition, topics are filtered from replication if:

- Name starts with mm2 and ends with .internal
- Name ends with .checkpoints.internal

Internal Kafka topics that start with a dot (.) or two underscores (__) are continued to be filtered from replication by default. If you have a custom policy that implements the interface, you might need to update it to correctly filter internal topics.

Behavioral Changes in Cruise Control

Functional adjustments and behavioral updates for Cruise Control are introduced in Cloudera DataFlow for Data Hub 7.3.2, its service packs, and cumulative hotfixes.

7.3.2

Summary: ZooKeeper service dependency removed from Cruise Control

Previous behavior:

Cruise Control had a ZooKeeper service dependency in Cloudera Manager.

New behavior:

The ZooKeeper service dependency is removed.

Summary: Default JMX settings changed to restrict connections to localhost

Previous behavior:

No dedicated Cloudera Manager configuration option existed to control JMX JVM flags for Cruise Control. JMX connections were unrestricted by default.

New behavior:

A new CC_JMX_OPTS Cloudera Manager configuration option is available. Its default value restricts JMX connections to localhost only and enables SSL. If you previously set CC_JMX_OPTS, your custom value is preserved on upgrade. To revert to open JMX, update CC_JMX_OPTS in Cloudera Manager.

Summary: Default Supported Goals and Anomaly Detection Goals updated in Cloudera Manager

Previous behavior:

The Cloudera Manager defaults for Supported Goals did not include BrokerSetAwareGoal. The defaults for Anomaly Detection Goals covered rack, replica, and disk capacity only, not network inbound, network outbound, or CPU capacity goals.

New behavior:

The default Supported Goals list includes BrokerSetAwareGoal. The default Anomaly Detection Goals list also includes NetworkInboundCapacityGoal, NetworkOutboundCapacityGoal, and CpuC apacityGoal. Cloudera Manager defaults for these goal lists now match the Cruise Control defaults. No action is required after upgrade.

Behavioral Changes in Cloudera Streaming Analytics

Review the list of Cloudera Streaming Analytics behavioral changes in Cloudera Data Flow for Cloudera Data Hub 7.3.2.

project-permissions REST API

The project-permissions REST API now restricts data requests to the user's own user ID or their assigned projects, and the endpoint now returns BasicUserView.

java.io.tmpdir for Cloudera SQL Stream Builder

Cloudera SQL Stream Builder processes now use the /var/tmp directory as the value for java.io.tmpdir for internal file operations.