

## Cloudera Flow Management Release Notes

Date published: 2019-06-26

Date modified: 2024-10-02



# Legal Notice

© Cloudera Inc. 2024. All rights reserved.

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

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

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

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

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

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

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

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

# Contents

<b>What's new in Cloudera Flow Management.....</b>	<b>4</b>
<b>Support matrix.....</b>	<b>6</b>
Component versions.....	6
System requirements.....	7
Supported operating systems.....	8
Supported NiFi Registry databases.....	9
Supported NiFi processors.....	9
Supported NiFi controller services.....	13
Supported NiFi reporting tasks.....	15
Supported NiFi parameter providers.....	15
Components supported by partners.....	16
<b>Download locations.....</b>	<b>16</b>
<b>Unsupported features.....</b>	<b>19</b>
<b>Technical preview features.....</b>	<b>19</b>
<b>Behavioral changes.....</b>	<b>20</b>
<b>Known issues in Cloudera Flow Management.....</b>	<b>21</b>
Known issues in Cloudera Flow Management 2.1.7.1000.....	21
Known issues in Cloudera Flow Management 2.1.7.....	28
<b>Fixed issues in Cloudera Flow Management.....</b>	<b>35</b>
<b>Fixed Common Vulnerabilities and Exposures in Cloudera Flow Management.....</b>	<b>36</b>

# What's new in Cloudera Flow Management

Learn about the new features and improvements in Cloudera Flow Management and see how they can benefit your workflows.

## Cloudera Flow Management 2.1.7.1000

On October 2, 2024, Cloudera Flow Management 2.1.7.1000 (Service Pack 1) was released, introducing the following new features and key updates.

The new NiFi pre-upgrade check functionality helps you to proactively assess your NiFi environment before upgrading to NiFi 2, which will be included in an upcoming Cloudera Flow Management 4.x release. The pre-upgrade check compiles a detailed list of potential breaking changes that may occur during an upgrade, allowing you to address these issues proactively. While the script identifies potential problems, it is your responsibility to take necessary actions to mitigate these issues as a preparation for a future upgrade. For more information, see [Pre-upgrade check for NiFi](#).

For detailed information about the issues resolved in Cloudera Flow Management 2.1.7.1000, see [Fixed issues](#) and [Fixed CVEs](#).

Cloudera Flow Management 2.1.7.1000 is based on Apache NiFi 1.26.0 and it also incorporates a lot of Cloudera exclusive features and improvements. If you are currently using Cloudera Flow Management 2.1.7, upgrade to Cloudera Flow Management 2.1.7.1000 to take advantage of the latest version. The new download links are available in [Download locations](#).



**Important:** You will need to be at least on Cloudera Flow Management 2.1.7 to upgrade to any future release that includes NiFi 2.0.

## Cloudera Flow Management 2.1.7

Discover the new functionalities and improvements in Cloudera Flow Management 2.1.7, and learn how these new features can benefit you.

The Cloudera Flow Management 2.1.7 release is based on Apache NiFi 1.26.0 and it also incorporates a lot of Cloudera exclusive features and improvements.

Here is an overview of what is new in this release:

### New processors

**CalculateParquetOffsets**

**CalculateParquetRowGroupOffsets**

These processors can be used in combination with ConvertRecord and Parquet Reader to significantly reduce the time required to convert very large Parquet files into another format.

**CaptureChangeDebeziumDB2**

**CaptureChangeDebeziumMySQL**

**CaptureChangeDebeziumOracle**

**CaptureChangeDebeziumPostgreSQL**

**CaptureChangeDebeziumSQLServer**

Currently in Technical Preview, these processors leverage the Debezium project to ingest Change Data Capture (CDC) events from external databases.

**DecryptContentAge**

**EncryptContentAge**

These new-generation processors are designed for data encryption and decryption. For more information, see [Modernizing Streaming Encryption with age in Apache NiFi](#).

**ListenOTLP**

This processor enables NiFi to act as a destination for OpenTelemetry Protocol (OTLP) agents to receive OpenTelemetry data from external applications. For more information about this new processor, see [Building OpenTelemetry Collection in Apache NiFi with Netty](#).

**PutClouderaHiveQL****PutClouderaHiveStreaming****PutClouderaORC****SelectClouderaHiveQL****UpdateClouderaHiveTable****TriggerClouderaHiveMetaStoreEvent**

These Cloudera exclusive components are designed to interact with Hive-based components in the Cloudera Data Platform. Since Hive components will no longer be part of Apache NiFi starting with NiFi 2.0, it is highly recommended to switch to these components as soon as possible to make the upgrade/migration to NiFi 2.0 easier.

**PutJiraIssue**

This processor allows you to create new issues in Jira using the Jira REST API.

**PutZendeskTicket**

This processor allows you to create Zendesk tickets using the Zendesk API.

**ConsumeElasticsearch**

This processor repeatedly runs a paginated query against a field using a Range query to consume new documents from an Elasticsearch index/query.

**FilterAttribute**

This processor filters the attributes of a FlowFile by retaining specified attributes and removing the rest or by removing specified attributes and retaining the rest.

**PackageFlowFile**

This processor packages FlowFile attributes and content into an output FlowFile that can be exported from NiFi and imported back into NiFi, preserving the original attributes and content.

**PublishSlack**

This processor allows you to post a message to the specified Slack channel.

**New controller services****ActiveMQJMSConnectionFactoryProvider**

This controller service allows you to interact with ActiveMQ without the need to deploy the JMS client on all of the NiFi nodes.

**ClouderaHiveConnectionPool**

This controller service allows you to interact with Hive without the need to deploy the required dependencies on all of the NiFi nodes.

**DatabaseTableSchemaRegistry**

This controller service enables you to retrieve the schema associated with a table from an external database. This allows you to validate the data going through NiFi against that schema before pushing the data into this table.

**ImpalaConnectionPool**

This controller service allows you to interact with Impala without the need to deploy the required dependencies on all NiFi nodes.

**RabbitMQJMSConnectionFactoryProvider**

This controller service allows you to interact with RabbitMQ without the need to deploy the JMS client on all NiFi nodes.

**ProtobufReader**

This record reader allows you to read Protobuf data with the record based components.

**YamlTreeReader**

This record reader allows you to read YAML data with the record based components.

**Related Information**

[Using Parameter Providers](#)

## Support matrix

Review the support matrix before you start installing Cloudera Flow Management.

### Component versions

Review the official Cloudera Flow Management component versions for compatibility with other applications.



**Note:** NiFi is compatible with the version of NiFi Registry included with your Cloudera Flow Management release or any later version.

**Cloudera Flow Management 2.1.7.1000 (SP1)**

- Apache NiFi 1.26.0.2.1.7.1000
- Apache NiFi Registry 1.26.0.2.1.7.1000

**Cloudera Flow Management 2.1.7**

- Apache NiFi 1.26.0.2.1.7.0
- Apache NiFi Registry 1.26.0.2.1.7.0

**Cloudera Flow Management 2.1.6.1000 (SP1)**

- Apache NiFi 1.23.1.2.1.6.1000
- Apache NiFi Registry 1.23.1.2.1.6.1000

**Cloudera Flow Management 2.1.6**

- Apache NiFi 1.23.1.2.1.6.0
- Apache NiFi Registry 1.23.1.2.1.6.0

**Cloudera Flow Management 2.1.5.1000 (SP1)**

- Apache NiFi 1.18.0.2.1.5.1000
- Apache NiFi Registry 1.18.0.2.1.5.1000

**Cloudera Flow Management 2.1.5**

- Apache NiFi 1.18.0.2.1.5.0
- Apache NiFi Registry 1.18.0.2.1.5.0

**Cloudera Flow Management 2.1.4.1000 (SP1)**

- Apache NiFi 1.16.0.2.1.4.1000
- Apache NiFi Registry 1.16.0.2.1.4.1000

**Cloudera Flow Management 2.1.4**

- Apache NiFi 1.16.0.2.1.4.0
- Apache NiFi Registry 1.16.0.2.1.4.0

**Cloudera Flow Management 2.1.3**

- Apache NiFi 1.15.2.2.1.3.0
- Apache NiFi Registry 1.15.2.2.1.3.0



**Note:** Apache NiFi and Apache NiFi Registry versions are unified in the 1.15.x release.

**Cloudera Flow Management 2.1.2**

- Apache NiFi 1.13.2.2.1.2.0
- Apache NiFi Registry 0.8.0.2.1.2.0

**Cloudera Flow Management 2.1.1**

- Apache NiFi 1.13.2.2.1.1.0
- Apache NiFi Registry 0.8.0.2.1.1.0

**Cloudera Flow Management 2.0.4**

- Apache NiFi 1.11.4
- Apache NiFi Registry 0.6.0

**Cloudera Flow Management 2.0.1**

- Apache NiFi 1.11.4
- Apache NiFi Registry 0.6.0

## System requirements

Review the system requirements before getting started with installing Cloudera Flow Management.

**Supported versions of CDP**

Cloudera Flow Management 2.1.7.1000 supports the following versions of CDP Private Cloud Base:

- CDP 7.1.9 and all Service Packs
- CDP 7.1.7 and all Service Packs



**Note:** Cloudera Flow Management provides a set of monitoring features when managed by Cloudera Manager. For these features to be available and working, you need to be using Cloudera Manager 7.6.1 or above.

**Supported JAVA Development Kits (JDK)**

Supported JDKs:



**Note:** If using Java 8, use only Update 252 (JDK 8u252) and later.

- Oracle Java™ SE Development Kit 8, Update 252 (JDK 8u252) and later
- OpenJDK 1.8, Update 252 (JDK 8u252) and later
- OpenJDK 11
- Azul Zulu JDK 1.8, Update 252 (JDK 8u252) and later
- Azul Zulu JDK 11
- JDK 17



**Note:** Cloudera Flow Management 2.1.7.1000 (Service Pack 1) supports JDK 17, but with specific configuration requirements. For more information, see [Limitation in JDK 17 support](#).

## Other system requirements

### ZooKeeper

You must install the ZooKeeper service available with your CDP Private Cloud Base cluster.

### Python

When deploying CFM on RHEL 8 and using Cloudera Manager with Python 2, you need to specify a symbolic link to python2.

```
ln -s /usr/bin/python2 /usr/bin/python
```

### Number of cores

Four cores per NiFi node is the minimum number of cores required by Cloudera to be supported. Cloudera recommends eight cores per NiFi node as it usually provides the best starting point for the most common use cases.

## Supported operating systems

Review the list of operating systems supported by Cloudera Flow Management.

Operating system	Versions
CentOS	<ul style="list-style-type: none"> <li>• 7.6</li> <li>• 7.7</li> <li>• 7.8</li> <li>• 7.9</li> <li>• 8.2</li> <li>• 8.4</li> </ul>
RHEL	<ul style="list-style-type: none"> <li>• 7.6</li> <li>• 7.7</li> <li>• 7.8</li> <li>• 7.9</li> <li>• 8.2</li> <li>• 8.4</li> <li>• 8.6</li> <li>• 8.7</li> <li>• 8.8</li> <li>• 8.9</li> <li>• 9.1</li> <li>• 9.2</li> </ul>
Oracle	<ul style="list-style-type: none"> <li>• 8.8</li> </ul>



Operating system	Versions
SLES	<ul style="list-style-type: none"> <li>12 SP5</li> <li>15 SP4</li> </ul>
Ubuntu	<ul style="list-style-type: none"> <li>18.04</li> <li>20.04</li> <li>22.04</li> </ul>
Windows	<ul style="list-style-type: none"> <li>10</li> <li>Server 2016</li> <li>Server 2019</li> </ul>

**Note:**

NiFi on Windows is only supported in standalone mode, not managed by Cloudera Manager or as part of a CDP cluster, and as a single instance installation. Clustering NiFi on Windows is not supported.

NiFi Registry is not supported on Windows.

## Supported NiFi Registry databases

Review the list of databases supported by NiFi Registry.

- H2
- PostgreSQL 10.x
- PostgreSQL 11.x
- PostgreSQL 12.x
- PostgreSQL 13.x
- PostgreSQL 14.x
- MySQL 8.x

### Related Information

[Supported NiFi processors](#)

[Supported NiFi controller services](#)

[Supported NiFi reporting tasks](#)

[Components supported by partners](#)

## Supported NiFi processors

Cloudera Flow Management is shipped with Apache NiFi and includes a set of processors, most of which are supported by Cloudera. You should be familiar with the available supported processors, and avoid using any unsupported processors in production environments.

Additional processors are developed and tested by the Cloudera community but are not officially supported by Cloudera. Processors 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.

AttributesToCSV	GetElasticsearch	PutDropbox
AttributesToJSON	GetFile	PutDynamoDB
Base64EncodeContent	GetFTP	PutDynamoDBRecord
CalculateParquetOffsets	GetGcpVisionAnnotateFilesOperationStatus	PutElasticsearchHttp1
CalculateParquetRowGroupOffsets	GetGcpVisionAnnotateImagesOperationStatus	PutElasticsearchHttpRecord1
CalculateRecordStats	GetHBase	PutElasticsearchJson

CaptureChangeDebeziumDB2 [Technical Preview]	GetHDFS	PutElasticsearchRecord1
CaptureChangeDebeziumMySQL [Technical Preview]	GetHDFSFileInfo	PutEmail
CaptureChangeDebeziumOracle [Technical Preview]	GetHDFSSequenceFile	PutFile
CaptureChangeDebeziumPostgreSQL [Technical Preview]	GetHTMLElement	PutFTP1
CaptureChangeDebeziumSQLServer [Technical Preview]	GetHTTP	PutGCSObject
CaptureChangeMySQL	GetHubSpot	PutGoogleDrive
CompressContent1, 2	GetIgniteCache	PutGridFS
ConnectWebSocket	GetJiraIssue	PutHBaseCell
ConsumeAMQP	GetJMSQueue	PutHBaseJSON
ConsumeAzureEventHub	GetJMSTopic	PutHBaseRecord1
ConsumeElasticsearch	GetMongoRecord	PutHDFS
ConsumeEWS	GetSFTP	PutHive3QL
ConsumeGCPubSub	GetShopify	PutHive3Streaming
ConsumeGCPubSubLite	GetSNMP	PutHiveQL
ConsumeJMS	GetSnowflakeIngestStatus	PutHiveStreaming
ConsumeKafka_1_0	GetSolr	PutHTMLElement
ConsumeKafka_2_0	GetSplunk	PutIceberg [Technical Preview]
ConsumeKafka_2_6	GetSQS	PutIcebergCDC
ConsumeKafka2CDP	GetTCP	PutInfluxDB
ConsumeKafka2RecordCDP	GetTwitter	PutJiraIssue
ConsumeKafkaRecord_1_0	GetWorkdayReport	PutJMS1
ConsumeKafkaRecord_2_0	GetZendesk	PutKinesisFirehose
ConsumeKafkaRecord_2_6	HandleHttpRequest	PutKinesisStream
ConsumeKinesisStream	HandleHttpResponse	PutKudu
ConsumeMQTT1	HashAttribute	PutLambda
ConsumeTwitter	HashContent	PutMongoRecord
ConsumeWindowsEventLog	IdentifyMimeType	PutORC1
ControlRate	InvokeAWSGatewayApi	PutParquet
ConvertAvroSchema	InvokeGRPC	PutRecord
ConvertAvroToJSON	InvokeGRPC	PutRedisHashRecord [Technical Preview]
ConvertAvroToORC	InvokeHTTP	PutRiemann
ConvertAvroToParquet	InvokeScriptedProcessor	PutS3Object
ConvertCharacterSet	JoinEnrichment	PutSalesforceObject
ConvertCSVToAvro	JoltTransformJSON	PutSFTP
ConvertJSONToAvro	JoltTransformRecord	PutSmbFile
ConvertJSONToSQL	JSLTTransformJSON	PutSnowflakeInternalStage
ConvertProtobuf	JsonQueryElasticsearch	PutSNS

ConvertRecord	ListAzureBlobStorage	PutSolrContentStream
CreateHadoopSequenceFile	ListAzureBlobStorage_v12	PutSolrRecord
CryptographicHashAttribute	ListAzureDataLakeStorage	PutSplunk
CryptographicHashContent	ListBoxFile	PutSplunkHTTP
DecryptContent	ListCDPObjectStore	PutSQL
DecryptContentAge	ListDatabaseTables	PutSQS1
DecryptContentCompatibility	ListDropbox	PutSyslog
DecryptContentPGP	ListenBeats	PutTCP
DeduplicateRecord	ListenFTP	PutUDP
DeleteAzureBlobStorage	ListenGRPC*	PutWebSocket
DeleteAzureBlobStorage_v12	ListenGRPC*	PutZendeskTicket
DeleteAzureDataLakeStorage	ListenHTTP	QueryAirtableTable
DeleteByQueryElasticsearch	ListenNetFlow	QueryCassandra
DeleteCDPObjectStore	ListenOTLP	QueryDatabaseTable1
DeleteDynamoDB	ListenRELP	QueryDatabaseTableRecord
DeleteGCSObject	ListenSyslog	QueryElasticsearchHttp
DeleteGridFS	ListenTCP	QueryRecord
DeleteHBaseCells	ListenTCPRecord	QuerySalesforceObject
DeleteHBaseRow	ListenTrapSNMP	QuerySolr
DeleteHDFS	ListenUDP	QuerySplunkIndexingStatus
DeleteS3Object	ListenUDPRecord	QueryWhois
DeleteSQS	ListenWebSocket	RemoveRecordField
DetectDuplicate	ListFile	ReplaceText
DistributeLoad	ListFTP	ReplaceTextWithMapping
DuplicateFlowFile	ListGCSBucket	ResizeImage1
EncodeContent	ListGoogleDrive	RetryFlowFile
EncryptContent2	ListHDFS	RouteHL7
EncryptContentAge	ListS3	RouteOnAttribute
EncryptContentPGP	ListSFTP	RouteOnContent
EnforceOrder	ListSmb	RouteText
EvaluateJsonPath	LogAttribute	SampleRecord
EvaluateXPath	LogMessage	ScanAccumulo
EvaluateXQuery	LookupAttribute	ScanAttribute1
ExecuteGroovyScript	LookupRecord	ScanContent
ExecuteInfluxDBQuery	MergeContent	ScanHBase
ExecuteProcess	MergeRecord1	ScriptedFilterRecord
ExecuteScript	ModifyCompression	ScriptedPartitionRecord
ExecuteSQL	ModifyHTMLElement	ScriptedTransformRecord
ExecuteSQLRecord	MonitorActivity	ScriptedValidateRecord
ExecuteStateless1,2	MoveAzureDataLakeStorage	ScrollElasticsearchHttp

ExecuteStreamCommand	MoveHDFS	SearchElasticsearch
ExtractAvroMetadata	Notify	SegmentContent
ExtractGrok	PackageFlowFile	SelectClouderaHiveQL
ExtractHL7Attributes	PaginatedJsonQueryElasticsearch	SelectHive3QL1
ExtractImageMetadata	ParseCEF1	SelectHiveQL
ExtractRecordSchema	ParseEvtx	SendTrapSNMP
ExtractText	ParseSyslog	SetSNMP
FetchAzureBlobStorage	PartitionRecord	SignContentPGP
FetchAzureBlobStorage_v12	PostHTTP	SplitAvro
FetchAzureDataLakeStorage	PublishAMQP	SplitContent
FetchBoxFile	PublishGCPubSub1	SplitJson1
FetchCDPObjectStore	PublishGCPubSubLite1	SplitRecord1
FetchDistributedMapCache	PublishJMS1	SplitText1
FetchDropbox	PublishKafka_1_0	SplitXml
FetchElasticsearchHttp	PublishKafka_2_0	StartAwsPollyJob
FetchFile	PublishKafka_2_6	StartAwsTextractJob
FetchFTP	PublishKafka2CDP	StartAwsTranscribeJob
FetchGCSObject	PublishKafka2RecordCDP	StartAwsTranslateJob
FetchGoogleDrive	PublishKafkaRecord_1_0	StartGcpVisionAnnotateFilesOperation
FetchGridFS	PublishKafkaRecord_2_0	StartGcpVisionAnnotateImagesOperation
FetchHBaseRow	PublishKafkaRecord_2_6	StartSnowflakeIngest
FetchHDFS	PublishMQTT	TagS3Object
FetchParquet	PublishSlack	TailFile
FetchS3Object	PutAccumuloRecord1	TransformXml
FetchSFTP	PutAzureBlobStorage	TriggerClouderaHiveMetaStoreEvent
FetchSmb	PutAzureBlobStorage_v12	TriggerHiveMetaStoreEvent
FilterAttribute	PutAzureCosmosDBRecord	UnpackContent
FlattenJson	PutAzureDataLakeStorage1	UpdateAttribute
ForkEnrichment	PutAzureEventHub	UpdateByQueryElasticsearch
ForkRecord	PutAzureQueueStorage1	UpdateClouderaHiveTable
GenerateFlowFile	PutAzureQueueStorage_v12	UpdateCounter
GenerateRecord	PutBigQuery	UpdateDatabaseTable
GenerateTableFetch	PutBigQueryBatch	UpdateDeltaLakeTable [Technical Preview]
GeoEnrichIP	PutBigQueryStreaming	UpdateHive3Table
GeoEnrichIPRecord	PutBoxFile	UpdateHiveTable
GeohashRecord	PutCassandraQL1	UpdateRecord
GetAsanaObject	PutCassandraRecord1	ValidateCsv
GetAwsPollyJobStatus	PutCDPObjectStore	ValidateJson
GetAwsTextractJobStatus	PutClouderaHiveQL	ValidateRecord
GetAwsTranscribeJobStatus	PutClouderaHiveStreaming	ValidateXml

GetAwsTranslateJobStatus	PutClouderaORC	VerifyContentMAC
GetAzureEventHub	PutCloudWatchMetric	VerifyContentPGP
GetAzureQueueStorage	PutCouchbaseKey	Wait
GetAzureQueueStorage_v12	PutDatabaseRecord1	YandexTranslate
GetCouchbaseKey1	PutDistributedMapCache	

#### Footnotes

- 1 – indicates a memory intensive processor
- 2 – indicates a CPU intensive processor
- \* – there are two ListenGRPC processors available, one is provided by Apache and the other is provided by Cloudera

#### Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi controller services](#)

[Supported NiFi reporting tasks](#)

[Components supported by partners](#)

## Supported NiFi controller services

Cloudera Flow Management is shipped with Apache NiFi and includes a set of controller services, most of which are supported by Cloudera. You should be familiar with the available supported controller services, and avoid using any unsupported controller services in production environments.

Additional controller services are developed and tested by the Cloudera community but are not officially supported by Cloudera. Controller services 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.

AccumuloService	ElasticSearchClientServiceImpl	PrometheusRecordSink
ActionHandlerLookup	ElasticSearchLookupService	ProtobufReader
ActiveMQJMSConnectionFactoryProvider	ElasticSearchStringLookupService	RabbitMQJMSConnectionFactoryProvider
ADLSCredentialsControllerService	EmailRecordSink	ReaderLookup
ADLSCredentialsControllerServiceLookup	EmbeddedHazelcastCacheManager	RecordSetWriterLookup
ADLSIDBrokerCloudCredentialsProviderControllerService	ExcelReader	RecordSinkHandler
AlertHandler	ExpressionHandler	RecordSinkServiceLookup
AmazonGlueSchemaRegistry	ExternalHazelcastCacheManager	RedisConnectionPoolService
AvroReader	FreeFormTextRecordSetWriter	RedisDistributedMapCacheClientService
AvroRecordSetWriter	GCPCredentialsControllerService	RedshiftConnectionPool
AvroSchemaRegistry	GrokReader	RestLookupService
AWSCredentialsProviderControllerService	HadoopCatalogService	ScriptedActionHandler
AWSIDBrokerCloudCredentialsProviderControllerService	HBase_1_1_2_BCPConnectionPool	ScriptedLookupService
AzureBlobIDBrokerCloudCredentialsProviderControllerService	HBase_1_1_2_ClientMapCacheClient	ScriptedReader
AzureCosmosDBClientService	HBase_1_1_2_ClientMapCacheService	ScriptedRecordSetWriter
AzureEventHubRecordSink	HBase_1_1_2_ClientService	ScriptedRecordSink
AzureServiceBusJMSConnectionFactoryProvider	HBase_1_1_2_ListLookupService	ScriptedRulesEngine
AzureStorageCredentialsControllerService	HBase_1_1_2_RecordLookupService	SimpleDatabaseLookupService

AzureStorageCredentialsControllerService_v12	HBase_2_ClientMapCacheService	SimpleKeyValueLookupService
AzureStorageCredentialsControllerServiceLookup	HBase_2_ClientService	SimpleRedisDistributedMapCacheClientService
AzureStorageCredentialsControllerServiceLookup	HBase_2_RecordLookupService	SimpleScriptedLookupService
CassandraDistributedMapCache	Hive3ConnectionPool	SiteToSiteReportingRecordSink
CassandraSessionProvider	HiveCatalogService	SmbjClientProviderService
CdpCredentialsProviderControllerService	HiveConnectionPool	SnowflakeComputingConnectionPool
CdpOauth2AccessTokenProviderControllerService	HortonworksSchemaRegistry	StandardAsanaClientProviderService
CEFReader	ImpalaConnectionPool	StandardAzureCredentialsControllerService
CiscoEmblemSyslogMessageReader	IPFIXReader	StandardDropboxCredentialService
ClouderaHiveConnectionPool	IPLookupService	StandardFileResourceService
ClouderaSchemaRegistry	JASN1Reader	StandardHashiCorpVaultClientService
CMLLookupService	JiraRecordSink	StandardHttpContextMap
ConfluentSchemaRegistry	JMSConnectionFactoryProvider	StandardJsonSchemaRegistry [Technical Preview]
CouchbaseClusterService	JndiJmsConnectionFactoryProvider	StandardOauth2AccessTokenProvider
CouchbaseKeyValueLookupService	JsonConfigBasedBoxClientService	StandardPGPPrivateKeyService
CouchbaseMapCacheClient	JsonPathReader	StandardPGPPublicKeyService
CouchbaseRecordLookupService	JsonRecordSetWriter	StandardPrivateKeyService
CSVReader	JsonTreeReader	StandardProxyConfigurationService
CSVRecordLookupService	KafkaRecordSink_1_0	StandardRestrictedSSLContextService
CSVRecordSetWriter	KafkaRecordSink_2_0	StandardS3EncryptionService
DatabaseRecordLookupService	KafkaRecordSink_2_6	StandardSnowflakeIngestManagerProviderService
DatabaseRecordSink	KerberosKeytabUserService	StandardSSLContextService
DatabaseTableSchemaRegistry	KerberosPasswordUserService	StandardWebClientServiceProvider
DBCPCConnectionPool	KerberosTicketCacheUserService	Syslog5424Reader
DBCPCConnectionPoolLookup	KeytabCredentialsService	SyslogReader
DistributedMapCacheClientService	KuduLookupService	UDPEventRecordSink
DistributedMapCacheLookupService	LoggingRecordSink	VolatileSchemaCache
DistributedMapCacheServer	LogHandler	WindowsEventLogReader
DistributedSetCacheClientService	MongoDBControllerService	XMLReader
DistributedSetCacheServer	MongoDBLookupService	XMLRecordSetWriter
EasyRulesEngineProvider	ParquetReader	YamlTreeReader
EasyRulesEngineService	ParquetRecordSetWriter	ZendeskRecordSink
EBCDICRecordReader [Technical Preview]	PostgreSQLConnectionPool	

### Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi processors](#)

[Supported NiFi reporting tasks](#)

[Components supported by partners](#)

## Supported NiFi reporting tasks

Cloudera Flow Management is shipped with Apache NiFi and includes a set of reporting tasks, most of which are supported by Cloudera. You should be familiar with the available supported reporting tasks, and avoid using any unsupported reporting tasks in production environments.

- `AmbariReportingTask`
- `ControllerStatusReportingTask`
- `MetricsEventReportingTask`
- `MonitorDiskUsage`
- `MonitorMemory`
- `PrometheusReportingTask`
- `QueryNiFiReportingTask`
- `ReportLineageToAtlas`
- `ScriptedReportingTask`
- `SiteToSiteBulletinReportingTask`
- `SiteToSiteMetricsReportingTask`
- `SiteToSiteProvenanceReportingTask`
- `SiteToSiteStatusReportingTask`

Additional reporting tasks are developed and tested by the Cloudera community but are not officially supported by Cloudera. Reporting tasks 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.

### Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi processors](#)

[Supported NiFi controller services](#)

[Components supported by partners](#)

## Supported NiFi parameter providers

Cloudera Flow Management is shipped with Apache NiFi and includes a set of parameter providers, most of which are supported by Cloudera. You should be familiar with the available supported parameter providers, and avoid using any unsupported parameter providers in production environments.

- `AwsSecretsManagerParameterProvider`
- `AzureKeyVaultSecretsParameterProvider`
- `CyberArkConjurParameterProvider`
- `DatabaseParameterProvider`
- `EnvironmentVariableParameterProvider`
- `FileParameterProvider`
- `GcpSecretManagerParameterProvider`
- `HashiCorpVaultParameterProvider`

Additional parameter providers are developed and tested by the Cloudera community but are not officially supported by Cloudera. Parameter providers 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.

## Components supported by partners

Learn about the processors and controller services built and supported by Cloudera partners.

These components are not officially supported by Cloudera even though Cloudera Quality Engineering teams added test coverage for them.

### Processors supported by partners

- ConsumePulsar (v1.18.0)
- ConsumePulsarRecord (v1.18.0)
- PublishPulsar (v1.18.0)
- PublishPulsarRecord (v1.18.0)

### Controller services supported by partners

- PulsarClientAthenzAuthenticationService (v1.18.0)
- PulsarClientJwtAuthenticationService (v1.18.0)
- PulsarClientOAuthAuthenticationService (v1.18.0)
- PulsarClientTlsAuthenticationService (v1.18.0)
- StandardPulsarClientService (v1.18.0)

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.

### Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi processors](#)

[Supported NiFi controller services](#)

[Supported NiFi reporting tasks](#)

## Download locations

You can download the Cloudera Flow Management software artifacts from the Cloudera Archive. There are different artifacts for different operating systems, standalone components, and Windows files.

Use the following tables to identify the Cloudera Flow Management repository location for your operating system and operational objectives.



### Note:

You must have credentials to download Cloudera Flow Management files. Your download credential is not the same as the credential you use to access the Cloudera Support Portal.

You can get download credentials in the following ways:

- Contact your Cloudera sales representative.
- Check the Welcome email you have received for your Cloudera Flow Management account.
- File a non-technical case on the [Cloudera Support Portal](#) for the Cloudera Support team to assist you.

**Table 1: RHEL/CentOS 7**

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/manifest.json</a>
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/CFM-2.1.7.1000-46-el7.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/CFM-2.1.7.1000-46-el7.parcel</a>
Parcel sha file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/CFM-2.1.7.1000-46-el7.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/CFM-2.1.7.1000-46-el7.parcel.sha</a>



File	Location
CSD	<b>NiFi</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar</a>

**Table 2: RHEL/CentOS 8**

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/manifest.json</a>
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/CFM-2.1.7.1000-46-el8.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/CFM-2.1.7.1000-46-el8.parcel</a>
Parcel sha file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/CFM-2.1.7.1000-46-el8.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/CFM-2.1.7.1000-46-el8.parcel.sha</a>
CSD	<b>NiFi:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat8/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar</a>

**Table 3: RHEL 9**

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/manifest.json</a>
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/CFM-2.1.7.1000-46-el9.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/CFM-2.1.7.1000-46-el9.parcel</a>
Parcel sha file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/CFM-2.1.7.1000-46-el9.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/CFM-2.1.7.1000-46-el9.parcel.sha</a>
CSD	<b>NiFi:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat9/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar</a>

**Table 4: SLES 12**

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/manifest.json</a>
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/CFM-2.1.7.1000-46-sles12.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/CFM-2.1.7.1000-46-sles12.parcel</a>
Parcel sha file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/CFM-2.1.7.1000-46-sles12.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/CFM-2.1.7.1000-46-sles12.parcel.sha</a>
CSD	<b>NiFi:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles12/yum/tars/parcel/NIFIRegistry-1.26.0.2.1.7.1000-46.jar</a>

**Table 5: SLES 15**

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/manifest.json</a>

File	Location
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/CFM-2.1.7.1000-46-sles15.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/CFM-2.1.7.1000-46-sles15.parcel</a>
Parcel sha file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/CFM-2.1.7.1000-46-sles15.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/CFM-2.1.7.1000-46-sles15.parcel.sha</a>
CSD	<b>NiFi:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/NIFIREGISTRY-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/sles15/yum/tars/parcel/NIFIREGISTRY-1.26.0.2.1.7.1000-46.jar</a>

Table 6: Ubuntu 20

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/manifest.json</a>
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/CFM-2.1.7.1000-46-focal.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/CFM-2.1.7.1000-46-focal.parcel</a>
Parcel SHA file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/CFM-2.1.7.1000-46-focal.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/CFM-2.1.7.1000-46-focal.parcel.sha</a>
CSD	<b>NiFi:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/NIFIREGISTRY-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu20/apt/tars/parcel/NIFIREGISTRY-1.26.0.2.1.7.1000-46.jar</a>

Table 7: Ubuntu 22

File	Location
Manifest	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/manifest.json">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/manifest.json</a>
Parcel	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/CFM-2.1.7.1000-46-jammy.parcel">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/CFM-2.1.7.1000-46-jammy.parcel</a>
Parcel SHA file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/CFM-2.1.7.1000-46-jammy.parcel.sha">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/CFM-2.1.7.1000-46-jammy.parcel.sha</a>
CSD	<b>NiFi:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/NIFI-1.26.0.2.1.7.1000-46.jar</a> <b>NiFi Registry:</b> <a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/NIFIREGISTRY-1.26.0.2.1.7.1000-46.jar">https://archive.cloudera.com/p/cfm2/2.1.7.1000/ubuntu22/apt/tars/parcel/NIFIREGISTRY-1.26.0.2.1.7.1000-46.jar</a>

Table 8: Standalone components (OS agnostic)

File	Location
NiFi (.tar.gz)	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/cdf_extensions/nifi-1.26.0.2.1.7.1000-46-bin.tar.gz">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/cdf_extensions/nifi-1.26.0.2.1.7.1000-46-bin.tar.gz</a>
NiFi (.tar.gz.sha256)	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/cdf_extensions/nifi-1.26.0.2.1.7.1000-46-bin.tar.gz.sha256">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/cdf_extensions/nifi-1.26.0.2.1.7.1000-46-bin.tar.gz.sha256</a>
NiFi Registry (.tar.gz)	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-registry-1.26.0.2.1.7.1000-46-bin.tar.gz">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-registry-1.26.0.2.1.7.1000-46-bin.tar.gz</a>
NiFi Registry (.tar.gz.sha256)	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-registry-1.26.0.2.1.7.1000-46-bin.tar.gz.sha256">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-registry-1.26.0.2.1.7.1000-46-bin.tar.gz.sha256</a>
NiFi Toolkit (.tar.gz)	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.26.0.2.1.7.1000-46-bin.tar.gz">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.26.0.2.1.7.1000-46-bin.tar.gz</a>

File	Location
NiFi Toolkit (.tar.gz.sha256)	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.26.0.2.1.7.1000-46-bin.tar.gz.sha256">https://archive.cloudera.com/p/cfm2/2.1.7.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.26.0.2.1.7.1000-46-bin.tar.gz.sha256</a>

**Table 9: Windows files**

File	Location
NiFi MSI	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/windows/nifi-2.1.7.1000-46.msi">https://archive.cloudera.com/p/cfm2/2.1.7.1000/windows/nifi-2.1.7.1000-46.msi</a>
NiFi MSI SHA file	<a href="https://archive.cloudera.com/p/cfm2/2.1.7.1000/windows/nifi-2.1.7.1000-46.msi.sha256">https://archive.cloudera.com/p/cfm2/2.1.7.1000/windows/nifi-2.1.7.1000-46.msi.sha256</a>

## Unsupported features

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 features

Rules engine components and handlers are removed in NiFi 2 to be replaced with a new rules engine. The below components are not supported and should not be used anymore.

- ActionHandlerLookup
- AlertHandler
- EasyRulesEngineProvider
- EasyRulesEngineService
- ExpressionHandler
- LogHandler
- RecordSinkHandler
- ScriptedActionHandler
- ScriptedRulesEngine

### Unsupported customizations

Cloudera cannot guarantee that default NiFi processors are compatible with proprietary protocol implementations or proprietary interface extensions. For example, Cloudera supports interfaces like JMS and JDBC that are built around standards, specifications, or open protocols, but does not support customizations of those interfaces, or proprietary extensions built on top of those interfaces.

## Technical preview features

The following features are available in Cloudera Flow Management 2.1.7.1000 but are not ready for production deployment. Cloudera encourages you to explore these technical preview features in non-production environments and provide feedback on your experiences through the [Cloudera Community Forums](#).

### Processors in technical preview

- CaptureChangeDebeziumDB2
- CaptureChangeDebeziumMySQL
- CaptureChangeDebeziumOracle
- CaptureChangeDebeziumPostgreSQL
- CaptureChangeDebeziumSQLServer

- PutIcebergCDC



**Note:** The processor supports equality deletes which may not be supported yet by other compute engines on CDP. It means that in case of delete operations, the files created by the processor may not be readable by engines like Hive, Spark, and so on. You should check the documentation of the compute engine you'd like to use to confirm if equality delete files are supported or not.

- PutRedisHashRecord
- UpdateDeltaLakeTable

#### Controller services in technical preview

- EBCDICRecordReader
- StandardJsonSchemaRegistry

## Behavioral changes

Learn about behavioral changes in Cloudera Flow Management 2.1.7.1000.

#### Secure communication between NiFi and ZooKeeper configured by default

If both ZooKeeper and NiFi services are secured, NiFi communication with ZooKeeper will be automatically configured as secured (TLS) using a new port, 2182. If you enforce TCP communication through a firewall and explicitly allow certain ports, you need to open them for port 2182.

If you do not want to use secure communication between ZooKeeper and NiFi, follow these steps to configure unsecured communication on port 2181:

##### 1. Update the ZooKeeper connection string:

- In Cloudera Manager, navigate to NiFi Configuration .
- Set `nifi.zookeeper.connect.string` by replacing `${ZK_QUORUM}` with the unsecure ZK QUORUM string, which has port 2181.

To find your ZooKeeper quorum string from a NiFi node, run the following command as root:

```
NIFI_PROC=$(ls -td /var/run/cloudera-scm-agent/process/NIFI/ | head -1);
grep "Connect String" $NIFI_PROC/state-management.xml | cut -d\> -f2 |
cut -d\< -f1; unset NIFI_PROC
```

This command will provide your connect string. For example:

```
host1:2181,host2:2181,host3:2181
```

##### 2. Add a safety valve for staging/state-management.xml in Cloudera Manager with the following property:

- Name: `xml.state-management.cluster-provider.zk-provider.property.Connect String`
- Value: `<YOUR ZOOKEEPER CONNECT STRING>`

##### 3. After upgrading to version 2.1.7, uncheck the `nifi.zookeeper.client.secure` option in Cloudera Manager.

#### ScriptedTransformRecord processor requires proper schema name attribute for record writer

NIFI-11523 introduced a fix that ensures the ScriptedTransformRecord processor uses the correct schema defined for the record writer. Previously, if the schema name attribute was set in the writer but not in the flow, it was ignored, defaulting to the reader schema. This behavior has been corrected, which may cause the processor to fail after upgrading if the schema name attribute is not set in the flow.

The failure is typically logged as:

```
org.apache.nifi.schema.access.SchemaNotFoundException: ${schema.name} did not provide appropriate Schema Name
```

To prevent failures, ensure that the schema name attribute is properly configured in the flow or match it to the schema defined for the record reader for identical behavior.

## Known issues in Cloudera Flow Management

Review the list of known issues in Cloudera Flow Management.

### Known issues in Cloudera Flow Management 2.1.7.1000

Review the list of known issues in Cloudera Flow Management 2.1.7.1000.

#### Known issues

##### **CFM-3870: QueryAirtableTable processor is no longer working**

The use of API keys for authentication in Airtable has been deprecated. As a result, the QueryAirtableTable processor no longer functions with API keys.

To ensure continued functionality, generate a Personal Access Token (PAT) in Airtable and replace the API key in the "API Key" property of the QueryAirtableTable processor for authentication.


##### **CFM-4200: Hive3QL slowness**

You can experience a performance issue due to slower record processing on EC2 clusters when using the Hive3QL processor in Cloudera Private Cloud Base 7.1.9 Service Pack 1.

##### **Limitation in JDK 17 support**

Cloudera Flow Management 2.1.7.1000 (Service Pack 1) supports JDK 17, but with specific configuration requirements. To ensure proper functionality and avoid any potential issues with JDK17 compatibility, add the following lines to the bootstrap.conf file (located in the NiFi Node Advanced Configuration Snippet for staging/bootstrap.conf.xml in Cloudera Manager):

```
java.arg.add-opens.java.lang=--add-opens=java.base/java.lang=ALL-UNNAMED
java.arg.add-opens.java.nio=--add-opens=java.base/java.nio=ALL-UNNAMED
java.arg.add-opens.java.net=--add-opens=java.base/java.net=ALL-UNNAMED
```



The screenshot shows a list of configuration items in the Cloudera Flow Management UI. There are three items visible, each with a Name, Value, Description, and a 'Final' checkbox.

Name	Value	Description	Final
java.arg.add-opens.java.lang	--add-opens=java.base/java.lang=ALL-UNNAMED		<input type="checkbox"/>
java.arg.add-opens.java.nio	--add-opens=java.base/java.nio=ALL-UNNAMED		<input type="checkbox"/>
java.arg.add-opens.java.net	--add-opens=java.base/java.net=ALL-UNNAMED		<input type="checkbox"/>

### Unused NiFi configuration values

The following NiFi configuration values are no longer in use. They are still visible in the UI, but they are obsolete and have no effect on functionality.

- nifi.nar.hotfix.provider.file.list.identifier
- nifi.nar.hotfix.provider.location.identifier
- nifi.nar.hotfix.provider.last.modification.identifier
- nifi.nar.hotfix.provider.directory.identifier
- nifi.nar.hotfix.provider.date.time.format
- nifi.nar.hotfix.provider.proxy.user
- nifi.nar.hotfix.provider.proxy.password
- nifi.nar.hotfix.provider.proxy.server
- nifi.nar.hotfix.provider.proxy.server.port
- nifi.nar.hotfix.provider.connect.timeout
- nifi.nar.hotfix.provider.read.timeout
- nifi.nar.hotfix.provider.nar.location
- nifi.nar.hotfix.provider.poll.interval
- nifi.nar.hotfix.provider.implementation
- nifi.nar.hotfix.provider.user.name
- nifi.nar.hotfix.provider.password
- nifi.nar.hotfix.provider.base.url
- nifi.nar.hotfix.provider.required.version
- nifi.nar.hotfix.provider.enabled

### CVEs not fixed

The following Common Vulnerabilities and Exposures (CVE) remain unresolved in Cloudera Flow Management 2.1.7.1000.

**CVE-2022-31159: Partial Path Traversal in com.amazonaws:aws-java-sdk-s3**

The AWS SDK for Java enables Java developers to work with Amazon Web Services. A partial-path traversal issue exists within the `downloadDirectory` method in the AWS S3 TransferManager component of the AWS SDK for Java v1 prior to version 1.12.261. Applications using the SDK control the `destinationDirectory` argument, but S3 object keys are determined by the application that uploaded the objects. The `downloadDirectory` method allows the caller to pass a filesystem object in the object key but contained an issue in the validation logic for the key name. A knowledgeable actor could bypass the validation logic by including a UNIX double-dot in the bucket key. Under certain conditions, this could permit them to retrieve a directory from their S3 bucket that is one level up in the filesystem from their working directory. This issue's scope is limited to directories whose name prefix matches the `destinationDirectory`. E.g. for `destinationDirectory`/tmp/foo``, the actor can cause a download to ``/tmp/foo-bar``, but not ``/tmp/bar``. If `com.amazonaws.services.s3.transfer.TransferManager::downloadDirectory`` is used to download an untrusted buckets contents, the contents of that bucket can be written outside of the intended destination directory. Version 1.12.261 contains a patch for this issue. As a workaround, when calling `com.amazonaws.services.s3.transfer.TransferManager::downloadDirectory``, pass a `KeyFilter`` that forbids `S3ObjectSummary`` objects that `getKey`` method return a string containing the substring ``.``.

#### **CVE-2019-11358**

jQuery before 3.4.0, as used in Drupal, Backdrop CMS, and other products, mishandles `jQuery.extend(true, {}, ...)` because of Object.prototype pollution. If an unsanitized source object contained an enumerable `__proto__` property, it could extend the native Object.prototype.

#### **CVE-2020-11022: Potential XSS vulnerability in jQuery**

In jQuery versions greater than or equal to 1.2 and before 3.5.0, passing HTML from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. `.html()`, `.append()`, and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.

#### **CVE-2020-11023: Potential XSS vulnerability in jQuery**

In jQuery versions greater than or equal to 1.0.3 and before 3.5.0, passing HTML containing `<option>` elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. `.html()`, `.append()`, and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.

#### **CVE-2024-21634: Ion Java StackOverflow vulnerability**

Amazon Ion is a Java implementation of the Ion data notation. Prior to version 1.10.5, a potential denial-of-service issue exists in `ion-java`` for applications that use `ion-java`` to deserialize Ion text encoded data, or deserialize Ion text or binary encoded data into the `IonValue`` model and then invoke certain `IonValue`` methods on that in-memory representation. An actor could craft Ion data that, when loaded by the affected application and/or processed using the `IonValue`` model, results in a `StackOverflowError`` originating from the `ion-java`` library. The patch is included in `ion-java`` 1.10.5. As a workaround, do not load data which originated from an untrusted source or that could have been tampered with.

#### **CVEs excluded based on the NiFi exclusion list**

#### **CVE-2024-35255: Azure Identity Libraries And Microsoft Authentication Library Elevation Of Privilege Vulnerability**

Azure Identity Libraries and Microsoft Authentication Library Elevation of Privilege Vulnerability

**Reason:** Transitive dep for msal4j 1.16.0, no newer version available.

#### **CVE-2018-14335**

An issue was discovered in H2 1.4.197. Insecure handling of permissions in the backup function allows attackers to read sensitive files (outside of their permissions) via a symlink to a fake database file.

**Reason:** No suggested resolution available yet.

**CVE-2022-3171: Memory handling vulnerability in ProtocolBuffers Java core and lite**

A parsing issue with binary data in protobuf-java core and lite versions prior to 3.21.7, 3.20.3, 3.19.6 and 3.16.3 can lead to a denial of service attack. Inputs containing multiple instances of non-repeated embedded messages with repeated or unknown fields causes objects to be converted back-n-forth between mutable and immutable forms, resulting in potentially long garbage collection pauses. We recommend updating to the versions mentioned above.

**Reason:** This is protobuf shaded by Hadoop / dependency comes from hive-exec 3.1.3000.7.1.9.1000-103.

**CVE-2022-3509: Parsing issue in protobuf textformat**

A parsing issue similar to CVE-2022-3171, but with textformat in protobuf-java core and lite versions prior to 3.21.7, 3.20.3, 3.19.6 and 3.16.3 can lead to a denial of service attack. Inputs containing multiple instances of non-repeated embedded messages with repeated or unknown fields causes objects to be converted back-n-forth between mutable and immutable forms, resulting in potentially long garbage collection pauses. We recommend updating to the versions mentioned above.

**Reason:** This is protobuf shaded by Hadoop / dependency comes from hive-exec 3.1.3000.7.1.9.1000-103.

**CVE-2021-22569: Denial of Service of protobuf-java parsing procedure**

An issue in protobuf-java allowed the interleaving of com.google.protobuf.UnknownFieldSet fields in such a way that would be processed out of order. A small malicious payload can occupy the parser for several minutes by creating large numbers of short-lived objects that cause frequent, repeated pauses. We recommend upgrading libraries beyond the vulnerable versions.

**Reason:** This is protobuf shaded by Hadoop / dependency comes from hive-exec 3.1.3000.7.1.9.1000-103.

**CVE-2024-36114: Decompressors Can Crash The JVM And Leak Memory Content In Aircompressor**

Aircompressor is a library with ports of the Snappy, LZ4, LZ0, and Zstandard compression algorithms to Java. All decompressor implementations of Aircompressor (LZ4, LZ0, Snappy, Zstandard) can crash the JVM for certain input, and in some cases also leak the content of other memory of the Java process (which could contain sensitive information). When decompressing certain data, the decompressors try to access memory outside the bounds of the given byte arrays or byte buffers. Because Aircompressor uses the JDK class `sun.misc.Unsafe` to speed up memory access, no additional bounds checks are performed and this has similar security consequences as out-of-bounds access in C or C++, namely it can lead to non-deterministic behavior or crash the JVM. Users should update to Aircompressor 0.27 or newer where these issues have been fixed. When decompressing data from untrusted users, this can be exploited for a denial-of-service attack by crashing the JVM, or to leak other sensitive information from the Java process. There are no known workarounds for this issue.

**Reason:** Dependency comes from hive-exec 3.1.3000.7.1.9.1000-103.

**CVE-2023-36479: Jetty vulnerable to errant command quoting in CGI Servlet**

Eclipse Jetty Canonical Repository is the canonical repository for the Jetty project. Users of the CgiServlet with a very specific command structure may have the wrong command executed. If a user sends a request to a org.eclipse.jetty.servlets.CGI Servlet for a binary with a space in its name, the servlet will escape the command by wrapping it in quotation marks. This wrapped command, plus an optional command prefix, will then be executed through a call to Runtime.exec. If the original binary name provided by the user contains a quotation mark followed by a space, the resulting command line will contain multiple tokens instead of one. This issue was patched in version 9.4.52, 10.0.16, 11.0.16 and 12.0.0-beta2.

**Reason:** Jetty 10 requires Java 11



**CVE-2021-42392**

The org.h2.util.JdbcUtils.getConnection method of the H2 database takes as parameters the class name of the driver and URL of the database. An attacker may pass a JNDI driver name and a URL leading to a LDAP or RMI servers, causing remote code execution. This can be exploited through various attack vectors, most notably through the H2 Console which leads to unauthenticated remote code execution.

**Reason:** h2-database-v14 using this version - We recommend to remove the NAR when not needed.

**CVE-2022-23221**

H2 Console before 2.1.210 allows remote attackers to execute arbitrary code via a jdbc:h2:mem JDBC URL containing the IGNORE\_UNKNOWN\_SETTINGS=TRUE;FORBID\_CREATION=FALSE;INIT=RUNSCRIPT substring, a different vulnerability than CVE-2021-42392.

**Reason:** h2-database-v14 using this version - We recommend to remove the NAR when not needed.

**CVE-2021-23463: XML External Entity (XXE) Injection**

The package com.h2database:h2 from 1.4.198 and before 2.0.202 are vulnerable to XML External Entity (XXE) Injection via the org.h2.jdbc.JdbcSQLXML class object, when it receives parsed string data from org.h2.jdbc.JdbcResultSet.getSQLXML() method. If it executes the getSource() method when the parameter is DOMSource.class it will trigger the vulnerability.

**Reason:** h2-database-v14 using this version - We recommend to remove the NAR when not needed.

**CVE-2022-45868**

The web-based admin console in H2 Database Engine before 2.2.220 can be started via the CLI with the argument -webAdminPassword, which allows the user to specify the password in cleartext for the web admin console. Consequently, a local user (or an attacker that has obtained local access through some means) would be able to discover the password by listing processes and their arguments. NOTE: the vendor states "This is not a vulnerability of H2 Console ... Passwords should never be passed on the command line and every qualified DBA or system administrator is expected to know that." Nonetheless, the issue was fixed in 2.2.220.

**Reason:** h2-database-v14 using this version - We recommend to remove the NAR when not needed.

**CVE-2018-1000840**

Processing Foundation Processing version 3.4 and earlier contains a XML External Entity (XXE) vulnerability in loadXML() function that can result in An attacker can read arbitrary files and exfiltrate their contents via HTTP requests. This attack appear to be exploitable via The victim must use Processing to parse a crafted XML document.

**Reason:** nifi-xml-processing is marked as vulnerability, no clear solution, might be false positive.

**CVE-2023-48795**

The SSH transport protocol with certain OpenSSH extensions, found in OpenSSH before 9.6 and other products, allows remote attackers to bypass integrity checks such that some packets are omitted (from the extension negotiation message), and a client and server may consequently end up with a connection for which some security features have been downgraded or disabled, aka a Terrapin attack. This occurs because the SSH Binary Packet Protocol (BPP), implemented by these extensions, mishandles the handshake phase and mishandles use of sequence numbers. For example, there is an effective attack against SSH's use of ChaCha20-Poly1305 (and CBC with Encrypt-then-MAC). The bypass occurs in chacha20-poly1305@openssh.com and (if CBC is used) the -etm@openssh.com MAC algorithms. This also affects Maverick Synergy Java SSH API before 3.1.0-SNAPSHOT, Dropbear through 2022.83, Ssh before 5.1.1 in Erlang/OTP, PuTTY before 0.80, AsyncSSH before 2.14.2, golang.org/x/crypto before 0.17.0, libssh before 0.10.6, libssh2 through 1.11.0, Thorn Tech SFTP Gateway before 3.4.6, Tera Term before 5.1, Paramiko before 3.4.0, jsch before 0.2.15, SFTPGO before 2.5.6, Netgate pfSense Plus through 23.09.1, Netgate

pfSense CE through 2.7.2, HPN-SSH through 18.2.0, ProFTPD before 1.3.8b (and before 1.3.9rc2), ORYX CycloneSSH before 2.3.4, NetSarang XShell 7 before Build 0144, CrushFTP before 10.6.0, ConnectBot SSH library before 2.2.22, Apache MINA sshd through 2.11.0, sshj through 0.37.0, TinySSH through 20230101, trilead-ssh2 6401, LANCOM LCOS and LANconfig, FileZilla before 3.66.4, Nova before 11.8, PKIX-SSH before 14.4, SecureCRT before 9.4.3, Transmit5 before 5.10.4, Win32-OpenSSH before 9.5.0.0p1-Beta, WinSCP before 6.2.2, Bitvise SSH Server before 9.32, Bitvise SSH Client before 9.33, KiTTY through 0.76.1.13, the net-ssh gem 7.2.0 for Ruby, the mscdex ssh2 module before 1.15.0 for Node.js, the thrussh library before 0.35.1 for Rust, and the Russh crate before 0.40.2 for Rust.

**Reason:** This is in NiFi Registry Web API

#### **CVE-2018-17196**

In Apache Kafka versions between 0.11.0.0 and 2.1.0, it is possible to manually craft a Produce request which bypasses transaction/idempotent ACL validation. Only authenticated clients with Write permission on the respective topics are able to exploit this vulnerability. Users should upgrade to 2.1.1 or later where this vulnerability has been fixed.

**Reason:** Not fixed, we recommend customers to use nifi-kafka-2-6-nar\*

#### **CVE-2024-39901: OpenSearch Observability Does Not Properly Restrict Access To Private Tenant Resources**

OpenSearch Observability is collection of plugins and applications that visualize data-driven events. An issue in the OpenSearch observability plugins allows unintended access to private tenant resources like notebooks. The system did not properly check if the user was the resource author when accessing resources in a private tenant, leading to potential data being revealed. The patches are included in OpenSearch 2.14.

**Reason:** CDP 7.1.9 SP1 related dependency, can't be fixed in NiFi

#### **CVE-2023-39017**

quartz-jobs 2.3.2 and below was discovered to contain a code injection vulnerability in the component org.quartz.jobs.ee.jms.SendQueueMessageJob.execute. This vulnerability is exploited via passing an unchecked argument. NOTE: this is disputed by multiple parties because it is not plausible that untrusted user input would reach the code location where injection must occur.

**Reason:** Still RC1 is the latest available version.

#### **CVE-2024-29025: Netty HttpPostRequestDecoder Can OOM**

Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers & clients. The `HttpPostRequestDecoder` can be tricked to accumulate data. While the decoder can store items on the disk if configured so, there are no limits to the number of fields the form can have, an attacker can send a chunked post consisting of many small fields that will be accumulated in the `bodyListHttpData` list. The decoder cumulates bytes in the `undecodedChunk` buffer until it can decode a field, this field can cumulate data without limits. This vulnerability is fixed in 4.1.108.Final.

**Reason:** CDP issue. Can't fix it in NiFi.

#### **CVE-2023-51437: Apache Pulsar: Timing Attack In SASL Token Signature Verification**

Observable timing discrepancy vulnerability in Apache Pulsar SASL Authentication Provider can allow an attacker to forge a SASL Role Token that will pass signature verification. Users are recommended to upgrade to version 2.11.3, 3.0.2, or 3.1.1 which fixes the issue. Users should also consider updating the configured secret in the `sasIJaasServerRoleTokenSignerSecretPath` file. Any component matching an above version running the SASL Authentication Provider is affected. That includes the Pulsar Broker, Proxy, Websocket Proxy, or Function Worker. 2.11 Pulsar users should upgrade to at least 2.11.3. 3.0 Pulsar users should upgrade to at least 3.0.2. 3.1 Pulsar users should upgrade to at least 3.1.1. Any users running Pulsar 2.8, 2.9, 2.10, and earlier should upgrade to one

of the above patched versions. For additional details on this attack vector, please refer to <https://codahale.com/a-lesson-in-timing-attacks/>.

**Reason:** Upgrade to higher nifi-pulsar-nar version is not possible, 1.20.0 requires Java17, 2.0 requires Java21

#### **CVE-2024-21490**

This affects versions of the package angular from 1.3.0. A regular expression used to split the value of the ng-srcset directive is vulnerable to super-linear runtime due to backtracking. With large carefully-crafted input, this can result in catastrophic backtracking and cause a denial of service.

**Note:** This package is EOL and will not receive any updates to address this issue. Users should migrate to [angular/core](https://www.npmjs.com/package/@angular/core).

**Reason:** Angularjs update is not possible, this fw is not supported, will be replaced by Angular 2+ in NiFi 2.x

#### **CVE-2023-50386: Apache Solr: Backup/Restore APIs allow for deployment of executables in malicious ConfigSets**

Improper Control of Dynamically-Managed Code Resources, Unrestricted Upload of File with Dangerous Type, Inclusion of Functionality from Untrusted Control Sphere vulnerability in Apache Solr. This issue affects Apache Solr: from 6.0.0 through 8.11.2, from 9.0.0 before 9.4.1. In the affected versions, Solr ConfigSets accepted Java jar and class files to be uploaded through the ConfigSets API. When backing up Solr Collections, these configSet files would be saved to disk when using the LocalFileSystemRepository (the default for backups). If the backup was saved to a directory that Solr uses in its ClassPath/ClassLoaders, then the jar and class files would be available to use with any ConfigSet, trusted or untrusted. When Solr is run in a secure way (Authorization enabled), as is strongly suggested, this vulnerability is limited to extending the Backup permissions with the ability to add libraries. Users are recommended to upgrade to version 8.11.3 or 9.4.1, which fix the issue. In these versions, the following protections have been added: \* Users are no longer able to upload files to a configSet that could be executed via a Java ClassLoader. \* The Backup API restricts saving backups to directories that are used in the ClassLoader.

**Reason:** Dependency coming from CDP

#### **CVE-2023-50291: Apache Solr: System Property redaction logic inconsistency can lead to leaked passwords**

Insufficiently Protected Credentials vulnerability in Apache Solr. This issue affects Apache Solr: from 6.0.0 through 8.11.2, from 9.0.0 before 9.3.0. One of the two endpoints that publishes the Solr process' Java system properties, /admin/info/properties, was only setup to hide system properties that had "password" contained in the name. There are a number of sensitive system properties, such as "basicauth" and "aws.secretKey" do not contain "password", thus their values were published via the "/admin/info/properties" endpoint. This endpoint populates the list of System Properties on the home screen of the Solr Admin page, making the exposed credentials visible in the UI. This /admin/info/properties endpoint is protected under the "config-read" permission. Therefore, Solr Clouds with Authorization enabled will only be vulnerable through logged-in users that have the "config-read" permission. Users are recommended to upgrade to version 9.3.0 or 8.11.3, which fixes the issue. A single option now controls hiding Java system property for all endpoints, "-Dsolr.hiddenSysProps". By default all known sensitive properties are hidden (including "-Dbasicauth"), as well as any property with a name containing "secret" or "password". Users who cannot upgrade can also use the following Java system property to fix the issue: '-Dsolr.redaction.system.pattern=.\*(password|secret|basicauth).\*'

**Reason:** Dependency coming from CDP

#### **CVE-2023-50292: Apache Solr: Solr Schema Designer blindly "trusts" all configsets, possibly leading to RCE by unauthenticated users**

Incorrect Permission Assignment for Critical Resource, Improper Control of Dynamically-Managed Code Resources vulnerability in Apache Solr. This issue affects Apache Solr: from 8.10.0 through

8.11.2, from 9.0.0 before 9.3.0. The Schema Designer was introduced to allow users to more easily configure and test new Schemas and configSets. However, when the feature was created, the "trust" (authentication) of these configSets was not considered. External library loading is only available to configSets that are "trusted" (created by authenticated users), thus non-authenticated users are unable to perform Remote Code Execution. Since the Schema Designer loaded configSets without taking their "trust" into account, configSets that were created by unauthenticated users were allowed to load external libraries when used in the Schema Designer. Users are recommended to upgrade to version 9.3.0, which fixes the issue.

**Reason:** Dependency coming from CDP

#### **CVE-2023-50298: Apache Solr: Solr can expose ZooKeeper credentials via Streaming Expressions**

Exposure of Sensitive Information to an Unauthorized Actor vulnerability in Apache Solr. This issue affects Apache Solr: from 6.0.0 through 8.11.2, from 9.0.0 before 9.4.1. Solr Streaming Expressions allows users to extract data from other Solr Clouds, using a "zkHost" parameter. When original SolrCloud is setup to use ZooKeeper credentials and ACLs, they will be sent to whatever "zkHost" the user provides. An attacker could setup a server to mock ZooKeeper, that accepts ZooKeeper requests with credentials and ACLs and extracts the sensitive information, then send a streaming expression using the mock server's address in "zkHost". Streaming Expressions are exposed via the "/streaming" handler, with "read" permissions. Users are recommended to upgrade to version 8.11.3 or 9.4.1, which fix the issue. From these versions on, only zkHost values that have the same server address (regardless of chroot), will use the given ZooKeeper credentials and ACLs when connecting.

**Reason:** Dependency coming from CDP

## Known issues in Cloudera Flow Management 2.1.7

Review the list of known issues in Cloudera Flow Management 2.1.7.

### Known issues

#### **Truststore changes with Ranger Plugin causing TLS handshake errors**

When using the Ranger plugin, the default truststore is changed from cacerts to AutoTLS truststore (cm-auto-global\_truststore.jks). This can lead to unintended issues such as TLS handshake errors with common CAs. Connections with common CAs may fail, causing service outages because the AutoTLS truststore contains only internal CA certificates and not the public root certificates.

Add the required certificates manually to the Cloudera Manager truststore.

1. Open Cloudera Manager and navigate to Administration Security Update Auto-TLS Truststore .
2. Import the certificates in PEM format.

#### **Configuration of java.arg.7**

A property has been added for defining java.arg.7 to provide the ability to override the default location of the temporary directory used by JDK. By default this value is empty in Cloudera Manager. If you use this argument for another purpose, change it to a different, unused argument number (or use letters instead: java.arg.mycustomargument). Not changing the argument can impact functionalities after upgrades/migrations.

#### **JDK error**

JDK 8 version u252 is supported. Any lower version may result in this error when NiFi starts:

```
SHA512withRSAandMGF1 Signature not available
```

When using Java 8, only version u252, and above are supported.

#### **JDK limitation**

JDK 8u271, JDK 8u281, and JDK 8u291 may cause socket leak issues in NiFi due to JDK-8245417 and JDK-8256818. Verify the build version of your JDK. Later builds are fixed as described in [JDK-8256818](#).

When using Java 8, only version u252, and above are supported.

### Kudu Client

All the records are sent as a single Kafka message containing an array of records.

There is an issue in the Kudu client preventing the creation of a new tables using the NiFi processors. The table needs to exist before NiFi tries to push data into it. You may see this error when this issue arises:

```
Caused by: org.apache.kudu.client.NonRecoverableException: failed
to wait for Hive Metastore notification log listener to catch
up: failed to retrieve notification log events: failed to open
Hive Metastore connection: SASL(-15): mechanism too weak for
this user
```

Verify the necessary table exists in Kudu.

### NiFi Node Connection test failures

In Cloudera Flow Management 2.1.3, Cloudera Manager includes a new health check feature. The health check alerts users if a NiFi instance is running but disconnected from the NiFi cluster. For this health check to be successful, you must update a Ranger policy. There is a known issue when the NiFi service is running but the NiFi Node(s) report Bad Health due to the NiFi Node Connection test.

Update the policy:

1. From the Ranger UI, access the Controller policy for the NiFi service.
2. Verify the `nifi` group is set in the policy.
3. Add the `nifi` user, to the policy, with READ permissions.

### NiFi UI Performance considerations

A known issue in Chrome 92.x causes significant slowness in the NiFi UI and may lead to high CPU consumption.

For more information, see the *Chrome Known Issues documentation* at [1235045](#).

Use another version of Chrome or a different browser.

### SSHJ version change and key negotiation issue with old SSH servers

ListSFTP and PutSFTP processors fail when using the legacy `ssh-rsa` algorithm for authentication with the following error:

```
UserAuthException: Exhausted available authentication methods
```

Set Key Algorithms Allowed property in PutSFTP to `ssh-rsa`.

### KeyStoreException: placeholder not found

After an upgrade, NiFi may fail to start with the following error:

```
WARN org.apache.nifi.web.server.JettyServer: Failed to start web
server... shutting down.
java.security.KeyStoreException: placeholder not found
```

The error is caused by missing configuration for the type of the keystore and truststore files.

1. Go to Cloudera Manager -> NiFi service -> Configuration.

2. Add the below properties for NiFi Node Advanced Configuration Snippet (Safety Valve) for staging/nifi.properties.xml.

```
nifi.security.keystoreType=**[value]**  
nifi.security.truststoreType=**[value]**
```

Where value must be PKCS12, JKS, or BCFKS. JKS is the preferred type, BCFKS and PKCS12 files are loaded with BouncyCastle provider.

3. Restart NiFi.

### InferAvroSchema may fail when inferring schema for JSON data

In Apache NiFi 1.17, the dependency on Apache Avro has been upgraded to 1.11.0. However, the InferAvroSchema processor depends on the hadoop-libraries NAR from which the Avro version comes from, causing a NoSuchMethodError exception.



**Important:** This processor is not supported by Cloudera and its use is highly discouraged as inferring a schema from the data is not recommended in production data flows.

Having well defined schemas ensures consistent behavior, allows for proper schema versioning and prevents downstream systems to generate errors because of unexpected schema changes. Besides, schema inference may not always be 100% accurate and can be an expensive operation in terms of performances.

Use the ExtractRecordSchema processor to infer the schema of your data with an appropriate reader and add the schema as a FlowFile attribute.

### CVEs not fixed

The following Common Vulnerabilities and Exposures (CVE) remain unresolved in Cloudera Flow Management 2.1.7.

#### CVE-2020-36518

jackson-databind before 2.13.0 allows a Java StackOverflow exception and denial of service via a large depth of nested objects.

**Reason:** com.cloudera:jwtprovider-knox:jar:shaded contains jackson-databind:2.10.5.1, and the dependency cannot be excluded upstream because it uses a downstream-specific package ('com.cloudera').

#### CVE-2021-46877

jackson-databind 2.10.x through 2.12.x before 2.12.6 and 2.13.x before 2.13.1 allows attackers to cause a denial of service (2 GB transient heap usage per read) in uncommon situations involving JsonNode JDK serialization.

**Reason:** com.cloudera:jwtprovider-knox:jar:shaded contains jackson-databind:2.10.5.1, and the dependency cannot be excluded upstream because it uses a downstream-specific package ('com.cloudera').

#### CVE-2022-42003

In FasterXML jackson-databind before versions 2.13.4.1 and 2.12.17.1, resource exhaustion can occur because of a lack of a check in primitive value deserializers to avoid deep wrapper array nesting, when the UNWRAP\_SINGLE\_VALUE\_ARRAYS feature is enabled.

**Reason:** com.cloudera:jwtprovider-knox:jar:shaded contains jackson-databind:2.10.5.1, and the dependency cannot be excluded upstream because it uses a downstream-specific package ('com.cloudera').

#### CVE-2022-42004



In FasterXML jackson-databind before 2.13.4, resource exhaustion can occur because of a lack of a check in BeanDeserializer.\_deserializeFromArray to prevent use of deeply nested arrays. An application is vulnerable only with certain customized choices for deserialization.

**Reason:** com.cloudera:jwtprovider-knox:jar:shaded contains jackson-databind:2.10.5.1, and the dependency cannot be excluded upstream because it uses a downstream-specific package ('com.cloudera').

#### **CVE-2021-23463: XML External Entity (XXE) Injection**

The package com.h2database:h2 from 1.4.198 and before 2.0.202 are vulnerable to XML External Entity (XXE) Injection via the org.h2.jdbc.JdbcSQLXML class object, when it receives parsed string data from org.h2.jdbc.JdbcResultSet.getSQLXML() method. If it executes the getSource() method when the parameter is DOMSource.class it will trigger the vulnerability.

**Reason:** The h2-database-v14 package uses this vulnerable version. Cloudera recommends removing the NAR when not needed.

#### **CVE-2021-42392**

The org.h2.util.JdbcUtils.getConnection method of the H2 database takes as parameters the class name of the driver and URL of the database. An attacker may pass a JNDI driver name and a URL leading to LDAP or RMI servers, causing remote code execution. This can be exploited through various attack vectors, most notably through the H2 Console which leads to unauthenticated remote code execution.

**Reason:** The h2-database-v14 package uses this vulnerable version. Cloudera recommends removing the NAR when not needed.

#### **CVE-2022-23221**

H2 Console before 2.1.210 allows remote attackers to execute arbitrary code via a jdbc:h2:mem JDBC URL containing the IGNORE\_UNKNOWN\_SETTINGS=TRUE;FORBID\_CREATION=FALSE;INIT=RUNSCRIPT substring, a different vulnerability than CVE-2021-42392.

**Reason:** The h2-database-v14 package uses this vulnerable version. Cloudera recommends removing the NAR when not needed.

#### **CVE-2022-45868**

The web-based admin console in H2 Database Engine before 2.2.220 can be started via the CLI with the argument -webAdminPassword, which allows the user to specify the password in cleartext for the web admin console. Consequently, a local user (or an attacker that has obtained local access through some means) would be able to discover the password by listing processes and their arguments. NOTE: the vendor states "This is not a vulnerability of H2 Console ... Passwords should never be passed on the command line and every qualified DBA or system administrator is expected to know that." Nonetheless, the issue was fixed in 2.2.220.

**Reason:** The h2-database-v14 package uses this vulnerable version. Cloudera recommends removing the NAR when not needed.

#### **CVE-2018-14335**

An issue was discovered in H2 1.4.197. Insecure handling of permissions in the backup function allows attackers to read sensitive files (outside of their permissions) via a symlink to a fake database file.

**Reason:** No suggested resolution is available yet.

#### **CVE-2023-36415: Azure Identity SDK Remote Code Execution Vulnerability**

Azure Identity SDK Remote Code Execution Vulnerability

**Reason:** No suggested resolution is available yet.

**CVE-2020-8908: Temp directory permission issue in Guava**

A temp directory creation vulnerability exists in all versions of Guava, allowing an attacker with access to the machine to potentially access data in a temporary directory created by the Guava API `com.google.common.io.Files.createTempDir()`. By default, on unix-like systems, the created directory is world-readable (readable by an attacker with access to the system). The method in question has been marked `@Deprecated` in versions 30.0 and later and should not be used. For Android developers, we recommend choosing a temporary directory API provided by Android, such as `context.getCacheDir()`. For other Java developers, we recommend migrating to the Java 7 API `java.nio.file.Files.createTempDirectory()` which explicitly configures permissions of 700, or configuring the Java runtime's `java.io.tmpdir` system property to point to a location whose permissions are appropriately configured.

**Reason:** The `gcs-connector` found in the POM file is shaded by Ranger.

**CVE-2023-2976: Use of temporary directory for file creation in `FileBackedOutputStream` in Guava**

Use of Java's default temporary directory for file creation in `FileBackedOutputStream` in Google Guava versions 1.0 to 31.1 on Unix systems and Android Ice Cream Sandwich allows other users and apps on the machine with access to the default Java temporary directory to be able to access the files created by the class. Even though the security vulnerability is fixed in version 32.0.0, we recommend using version 32.0.1 as version 32.0.0 breaks some functionality under Windows.

**Reason:** The `gcs-connector` found in the POM file is shaded by Ranger.

**CVE-2021-22569: Denial of Service of protobuf-java parsing procedure**

An issue in `protobuf-java` allowed the interleaving of `com.google.protobuf.UnknownFieldSet` fields in such a way that would be processed out of order. A small malicious payload can occupy the parser for several minutes by creating large numbers of short-lived objects that cause frequent, repeated pauses. We recommend upgrading libraries beyond the vulnerable versions.

**Reason:** This is `protobuf` shaded by Hadoop.

**CVE-2022-3171: Memory handling vulnerability in ProtocolBuffers Java core and lite**

A parsing issue with binary data in `protobuf-java` core and lite versions prior to 3.21.7, 3.20.3, 3.19.6 and 3.16.3 can lead to a denial of service attack. Inputs containing multiple instances of non-repeated embedded messages with repeated or unknown fields cause objects to be converted back-n-forth between mutable and immutable forms, resulting in potentially long garbage collection pauses. We recommend updating to the versions mentioned above.

**Reason:** This is `protobuf` shaded by Hadoop.

**CVE-2022-3509: Parsing issue in protobuf textformat**

A parsing issue similar to CVE-2022-3171, but with `textformat` in `protobuf-java` core and lite versions prior to 3.21.7, 3.20.3, 3.19.6 and 3.16.3 can lead to a denial of service attack. Inputs containing multiple instances of non-repeated embedded messages with repeated or unknown fields cause objects to be converted back-n-forth between mutable and immutable forms, resulting in potentially long garbage collection pauses. We recommend updating to the versions mentioned above.

**Reason:** This is `protobuf` shaded by Hadoop.

**CVE-2019-11358**

jQuery before 3.4.0, as used in Drupal, Backdrop CMS, and other products, mishandles `jQuery.extend(true, { }, ...)` because of `Object.prototype` pollution. If an unsanitized source object contained an enumerable `__proto__` property, it could extend the native `Object.prototype`.

**Reason:** JQuery version upgrade is needed in the UI.

**CVE-2020-11022: Potential XSS vulnerability in jQuery**



In jQuery versions greater than or equal to 1.2 and before 3.5.0, passing HTML from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. `.html()`, `.append()`, and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.

**Reason:** JQuery version upgrade is needed in the UI.

#### **CVE-2020-11023: Potential XSS vulnerability in jQuery**

In jQuery versions greater than or equal to 1.0.3 and before 3.5.0, passing HTML containing `<option>` elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. `.html()`, `.append()`, and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.

**Reason:** JQuery version upgrade is needed in the UI.

#### **CVE-2021-29425: Possible limited path traversal vulnerability in Apache Commons IO**

In Apache Commons IO before 2.7, When invoking the method `FileNameUtils.normalize` with an improper input string, like `"../foo"`, or `"\\..\\foo"`, the result would be the same value, thus possibly providing access to files in the parent directory, but not further above (thus "limited" path traversal), if the calling code would use the result to construct a path value.

**Reason:** jwtprovider-knox found in the POM file is shaded by Ranger.

#### **CVE-2023-1370: Stack exhaustion in json-smart leads to denial of service when parsing malformed JSON**

[Json-smart](<https://netplex.github.io/json-smart/>) is a performance focused, JSON processor lib. When reaching a '[' or '{' character in the JSON input, the code parses an array or an object respectively. It was discovered that the code does not have any limit to the nesting of such arrays or objects. Since the parsing of nested arrays and objects is done recursively, nesting too many of them can cause a stack exhaustion (stack overflow) and crash the software.

**Reason:** jwtprovider-knox found in the POM file is shaded by Ranger.

#### **CVE-2018-17196**

In Apache Kafka versions between 0.11.0.0 and 2.1.0, it is possible to manually craft a Produce request which bypasses transaction/idempotent ACL validation. Only authenticated clients with Write permission on the respective topics are able to exploit this vulnerability. Users should upgrade to 2.1.1 or later where this vulnerability has been fixed.

**Reason:** Not fixed. Cloudera recommends to use nifi-kafka-2-6-nar\*

#### **CVE-2023-48795**

The SSH transport protocol with certain OpenSSH extensions, found in OpenSSH before 9.6 and other products, allows remote attackers to bypass integrity checks such that some packets are omitted (from the extension negotiation message), and a client and server may consequently end up with a connection for which some security features have been downgraded or disabled, aka a Terrapin attack. This occurs because the SSH Binary Packet Protocol (BPP), implemented by these extensions, mishandles the handshake phase and mishandles use of sequence numbers. For example, there is an effective attack against SSH's use of ChaCha20-Poly1305 (and CBC with Encrypt-then-MAC). The bypass occurs in `chacha20-poly1305@openssh.com` and (if CBC is used) the `etm@openssh.com` MAC algorithms. This also affects Maverick Synergy Java SSH API before 3.1.0-SNAPSHOT, Dropbear through 2022.83, Ssh before 5.1.1 in Erlang/OTP, PuTTY before 0.80, AsyncSSH before 2.14.2, `golang.org/x/crypto` before 0.17.0, libssh before 0.10.6, libssh2 through 1.11.0, Thorn Tech SFTP Gateway before 3.4.6, Tera Term before 5.1, Paramiko before 3.4.0, jsch before 0.2.15, SFTPGO before 2.5.6, Netgate pfSense Plus through 23.09.1, Netgate pfSense CE through 2.7.2, HPN-SSH through 18.2.0, ProFTPD before 1.3.8b (and before 1.3.9rc2), ORYX CycloneSSH before 2.3.4, NetSarang XShell 7 before Build 0144, CrushFTP before 10.6.0, ConnectBot SSH library before 2.2.22, Apache MINA sshd through 2.11.0, sshj through 0.37.0, TinySSH through 20230101, trilead-ssh2 6401, LANCOM LCOS and LANconfig, FileZilla before

3.66.4, Nova before 11.8, PKIX-SSH before 14.4, SecureCRT before 9.4.3, Transmit5 before 5.10.4, Win32-OpenSSH before 9.5.0.0p1-Beta, WinSCP before 6.2.2, Bitvise SSH Server before 9.32, Bitvise SSH Client before 9.33, KiTTY through 0.76.1.13, the net-ssh gem 7.2.0 for Ruby, the mscdex ssh2 module before 1.15.0 for Node.js, the thrussh library before 0.35.1 for Rust, and the Russh crate before 0.40.2 for Rust.

**Reason:** This is in the NiFi Registry Web API.

#### **CVE-2023-36479: Jetty vulnerable to errant command quoting in CGI Servlet**

Eclipse Jetty Canonical Repository is the canonical repository for the Jetty project. Users of the CgiServlet with a very specific command structure may have the wrong command executed. If a user sends a request to a org.eclipse.jetty.servlets.CGI Servlet for a binary with a space in its name, the servlet will escape the command by wrapping it in quotation marks. This wrapped command, plus an optional command prefix, will then be executed through a call to Runtime.exec. If the original binary name provided by the user contains a quotation mark followed by a space, the resulting command line will contain multiple tokens instead of one. This issue was patched in version 9.4.52, 10.0.16, 11.0.16 and 12.0.0-beta2.

**Reason:** Jetty 10 requires Java 11.

#### **CVE-2018-1000840**

Processing Foundation Processing version 3.4 and earlier contains a XML External Entity (XXE) vulnerability in loadXML() function that can result in An attacker can read arbitrary files and exfiltrate their contents via HTTP requests. This attack appears to be exploitable via The victim must use Processing to parse a crafted XML document.

**Reason:** nifi-xml-processing is marked as a vulnerability, with no clear solution.

#### **CVE-2020-5408: Dictionary attack with Spring Security queryable text encryptor**

Spring Security versions 5.3.x prior to 5.3.2, 5.2.x prior to 5.2.4, 5.1.x prior to 5.1.10, 5.0.x prior to 5.0.16 and 4.2.x prior to 4.2.16 use a fixed null initialization vector with CBC Mode in the implementation of the queryable text encryptor. A malicious user with access to the data that has been encrypted using such an encryptor may be able to derive the unencrypted values using a dictionary attack.

**Reason:** The solution suggests downgrading the current version of the spring-security-crypto dependency, which is currently not feasible.

#### **CVEs excluded based on the NiFi exclusion list**

##### **CVE-2023-4759: Improper handling of case insensitive filesystems in Eclipse JGit allows arbitrary file write**

Arbitrary File Overwrite in Eclipse JGit <= 6.6.0 In Eclipse JGit, all versions <= 6.6.0.202305301015-r, a symbolic link present in a specially crafted git repository can be used to write a file to locations outside the working tree when this repository is cloned with JGit to a case-insensitive filesystem, or when a checkout from a clone of such a repository is performed on a case-insensitive filesystem. This can happen on checkout (DirCacheCheckout), merge (ResolveMerger via its WorkingTreeUpdater), pull (PullCommand using merge), and when applying a patch (PatchApplier). This can be exploited for remote code execution (RCE), for instance if the file written outside the working tree is a git filter that gets executed on a subsequent git command. The issue occurs only on case-insensitive file systems, like the default file systems on Windows and macOS. The user performing the clone or checkout must have the rights to create symbolic links for the problem to occur, and symbolic links must be enabled in the git configuration. Setting the git configuration option core.symlinks = false before checking out avoids the problem. The issue was fixed in Eclipse JGit version 6.6.1.202309021850-r and 6.7.0.202309050840-r, available via Maven Central <https://repo1.maven.org/maven2/org/eclipse/jgit/> and [repo.eclipse.org https://repo.eclipse.org/content/repositories/jgit-releases/](https://repo.eclipse.org/content/repositories/jgit-releases/) . A backport is available in 5.13.3 starting from

5.13.3.202401111512-r. The JGit maintainers would like to thank RyotaK for finding and reporting this issue.

#### **CVE-2024-21634: Ion Java StackOverflow vulnerability**

Amazon Ion is a Java implementation of the Ion data notation. Prior to version 1.10.5, a potential denial-of-service issue exists in `ion-java` for applications that use `ion-java` to deserialize Ion text encoded data, or deserialize Ion text or binary encoded data into the `IonValue` model and then invoke certain `IonValue` methods on that in-memory representation. An actor could craft Ion data that, when loaded by the affected application and/or processed using the `IonValue` model, results in a `StackOverflowError` originating from the `ion-java` library. The patch is included in `ion-java` 1.10.5. As a workaround, do not load data which originated from an untrusted source or that could have been tampered with.

## Fixed issues in Cloudera Flow Management

Review the list of issues resolved in Cloudera Flow Management.

#### **Fixed issues in Cloudera Flow Management 2.1.7.1000**

- NIFI-13496 HDFS processors' classloader isolation key should include Hadoop configuration files
- NIFI-13498 Deprecate several processors and controller services
- NIFI-13542 Fix missing max string length parameter usage in multiple JSON Readers and Schema Access Strategy
- NIFI-13550 Add additional details regarding the 'Use Starting Row' Strategy to the ExcelReader's additional details
- NIFI-13553 PutAzureDataExplorer broken due to bug in msal4j v1.15.0
- NIFI-13557 Record Inference Code (org.apache.nifi.util.SchemaInferenceUtil.getDataType) Does not Allow Single Digit Months
- NIFI-13566 JettyServer remains started when a ClassNotFoundException happens
- NIFI-13573 Bump google.libraries.version from 26.37.0 to 26.40.0
- NIFI-13574 Upgrade Azure SDK BOM to 1.2.25 and msal4j to 1.16.1
- NIFI-13593 PutIceberg issue with decimal scale
- NIFI-13605 Make AbstractHadoopProcessor.KERBEROS\_USER\_SERVICE public
- NIFI-13621 Upgrade JGit to 5.13.3.202401111512
- NIFI-13623 Bump gcp.sdk.version to 26.40.0 for nifi-property-protection-gcp
- NIFI-13627 Bump azure-sdk-bom to 1.2.25, and msal4j to 1.16.1 for nifi-property-protection-azure
- NIFI-13640 Extract Ranger Solr version to property
- NIFI-13655 Upgrade 1.x Shared Dependencies including JacksonXML and others
- NIFI-13666 S3 processors fail to catch IllegalArgumentException
- NIFI-13669 Add reference to alternative processor in InvokeAWSGatewayAPI deprecation notice
- NIFI-13675 Fix tooltip for Parameter description
- NIFI-13691 Add Kerberos User Service to KuduLookupService (NiFi 1.x)
- NIFI-13692 ResizeImage fails to catch exceptions in many cases
- NIFI-13715 StandardProvenanceEventRecord.hashCode() is not consistent with equals() in handling Parent/Child FlowFiles
- NIFI-13720 Component is not reloaded when the isolation key depends on service property
- NIFI-13722 Kerberos ticket renewal issue due static thread pool in Iceberg library
- CFM-2600 Default bucket in nifi registry is missing
- CFM-3518 Pre-upgrade check script/toolkit
- CFM-3845 Diagnostic bundle should include as many logs as possible in scope of limit size
- CFM-3846 PutIcebergCDC issue: Failed to specify server's Kerberos principal name

- CFM-3921 CFM DataHub CSD should allow Persistent repository configuration
- CFM-4020 Expose nifi.working.directory property during initial NiFi service install
- CFM-4040 In CSD code nifi.jdk.home (CUSTOM\_JAVA\_HOME) is not recognized
- CFM-4148 Add Kerberos User Service to CDPObjectStore processors
- CFM-4153 Sometimes, the default bucket is not created in NiFi Registry
- CFM-4161 Fix double escaping for password properties with already escaped specific symbols in NiFiXmlTransformer.java
- CFM-4067 Bump postgresql driver version in debezium-connector-postgres lib to mitigate CVE-2024-1597

### Fixed issues in Cloudera Flow Management 2.1.7

In addition to Apache NiFi 1.26.0, the following fixes have been implemented:

- NIFI-13181: Updated msal4j to version 1.15.0
- NIFI-13151: Deprecated Couchbase Components
- NIFI-13152: Deprecated DataDogReportingTask
- NIFI-13008: Added CLI command to upgrade all instances of a versioned flow

## Fixed Common Vulnerabilities and Exposures in Cloudera Flow Management

Review the list of fixed Common Vulnerabilities and Exposures (CVE) in Cloudera Flow Management.

### Fixed CVEs in Cloudera Flow Management 2.1.7.1000

#### **CVE-2023-36415: Azure Identity SDK Remote Code Execution Vulnerability**

Azure Identity SDK Remote Code Execution Vulnerability

#### **CVE-2024-37389: Apache NiFi: Improper Neutralization Of Input In Parameter Context Description**

Apache NiFi 1.10.0 through 1.26.0 and 2.0.0-M1 through 2.0.0-M3 support a description field in the Parameter Context configuration that is vulnerable to cross-site scripting. An authenticated user, authorized to configure a Parameter Context, can enter arbitrary JavaScript code, which the client browser will execute within the session context of the authenticated user. Upgrading to Apache NiFi 1.27.0 or 2.0.0-M4 is the recommended mitigation.

#### **CVE-2024-1597: Pgjdbc SQL Injection Via Line Comment Generation**

pgjdbc, the PostgreSQL JDBC Driver, allows attacker to inject SQL if using `PreferQueryMode=SIMPLE`. Note this is not the default. In the default mode there is no vulnerability. A placeholder for a numeric value must be immediately preceded by a minus. There must be a second placeholder for a string value after the first placeholder; both must be on the same line. By constructing a matching string payload, the attacker can inject SQL to alter the query, bypassing the protections that parameterized queries bring against SQL Injection attacks. Versions before 42.7.2, 42.6.1, 42.5.5, 42.4.4, 42.3.9, and 42.2.28 are affected.

#### **CVE-2022-41946: TemporaryFolder on unix-like systems does not limit access to created files in pgjdbc**

pgjdbc is an open source postgresql JDBC Driver. In affected versions a prepared statement using either `PreparedStatement.setText(int, InputStream)` or `PreparedStatement.setBytea(int, InputStream)` will create a temporary file if the `InputStream` is larger than 2k. This will create a temporary file which is readable by other users on Unix like systems, but not MacOS. On Unix like systems, the system's temporary directory is shared between all users on that system. Because of this, when files and directories are written into this directory they are, by default, readable by other users on that same system. This vulnerability does not allow other users to overwrite the contents of these directories or files. This is purely an information disclosure vulnerability. Because certain JDK file system APIs were only added in JDK 1.7, this fix is dependent upon the version of the

JDK you are using. Java 1.7 and higher users: this vulnerability is fixed in 4.5.0. Java 1.6 and lower users: no patch is available. If you are unable to patch, or are stuck running on Java 1.6, specifying the `java.io.tmpdir` system environment variable to a directory that is exclusively owned by the executing user will mitigate this vulnerability.

#### **CVE-2023-4759**

Arbitrary File Overwrite in Eclipse JGit <= 6.6.0 In Eclipse JGit, all versions <= 6.6.0.202305301015-r, a symbolic link present in a specially crafted git repository can be used to write a file to locations outside the working tree when this repository is cloned with JGit to a case-insensitive filesystem, or when a checkout from a clone of such a repository is performed on a case-insensitive filesystem. This can happen on checkout (DirCacheCheckout), merge (ResolveMerger via its WorkingTreeUpdater), pull (PullCommand using merge), and when applying a patch (PatchApplier). This can be exploited for remote code execution (RCE), for instance if the file written outside the working tree is a git filter that gets executed on a subsequent git command. The issue occurs only on case-insensitive filesystems, like the default filesystems on Windows and macOS. The user performing the clone or checkout must have the rights to create symbolic links for the problem to occur, and symbolic links must be enabled in the git configuration. Setting git configuration option `core.symlinks = false` before checking out avoids the problem. The issue was fixed in Eclipse JGit version 6.6.1.202309021850-r and 6.7.0.202309050840-r, available via Maven Central <https://repo1.maven.org/maven2/org/eclipse/jgit/> and [repo.eclipse.org https://repo.eclipse.org/content/repositories/jgit-releases/](https://repo.eclipse.org/content/repositories/jgit-releases/). A backport is available in 5.13.3 starting from 5.13.3.202401111512-r. The JGit maintainers would like to thank RyotaK for finding and reporting this issue.

#### **CVE-2024-23081**

ThreeTen Backport v1.6.8 was discovered to contain a `NullPointerException` via the component `org.threeten.bp.LocalDate::compareTo(ChronoLocalDate)`. NOTE: this is disputed by multiple third parties who believe there was not reasonable evidence to determine the existence of a vulnerability. The submission may have been based on a tool that is not sufficiently robust for vulnerability identification.

#### **CVE-2024-23082**

ThreeTen Backport v1.6.8 was discovered to contain an integer overflow via the component `org.threeten.bp.format.DateTimeFormatter::parse(CharSequence, ParsePosition)`. NOTE: this is disputed by multiple third parties who believe there was not reasonable evidence to determine the existence of a vulnerability. The submission may have been based on a tool that is not sufficiently robust for vulnerability identification.

### **Fixed CVEs in Cloudera Flow Management 2.1.7**

#### **CVE-2018-10237**

Unbounded memory allocation in Google Guava 11.0 through 24.x before 24.1.1 allows remote attackers to conduct denial of service attacks against servers that depend on this library and deserialize attacker-provided data, because the `AtomicDoubleArray` class (when serialized with Java serialization) and the `CompoundOrdering` class (when serialized with GWT serialization) perform eager allocation without appropriate checks on what a client has sent and whether the data size is reasonable.

#### **CVE-2019-10172**

A flaw was found in `org.codehaus.jackson:jackson-mapper-asl:1.9.x` libraries. XML external entity vulnerabilities similar to CVE-2016-3720 also affects `codehaus jackson-mapper-asl` libraries but in different classes.

#### **CVE-2020-25649**

A flaw was found in FasterXML Jackson Databind, where it did not have entity expansion secured properly. This flaw allows vulnerability to XML external entity (XXE) attacks. The highest threat from this vulnerability is data integrity.

#### **CVE-2021-0341**

In verifyHostName of OkHostnameVerifier.java, there is a possible way to accept a certificate for the wrong domain due to improperly used crypto. This could lead to remote information disclosure with no additional execution privileges needed. User interaction is not needed for exploitation. Product: Android Versions: Android-8.1 Android-9 Android-10 Android-11 Android ID: A-171980069

#### **CVE-2022-25647: Deserialization of Untrusted Data**

The package com.google.code.gson:gson before 2.8.9 are vulnerable to Deserialization of Untrusted Data using the writeReplace() method in internal classes, which may lead to DoS attacks.

#### **CVE-2023-0833: Red hat a-mq streams: component version with information disclosure flaw**

A flaw was found in Red Hat's AMQ-Streams, which ships a version of the OKHttp component with an information disclosure flaw via an exception triggered by a header containing an illegal value. This issue could allow an authenticated attacker to access information outside of their regular permissions.

#### **CVE-2023-34055: Spring Boot server Web Observations DoS Vulnerability**

In Spring Boot versions 2.7.0 - 2.7.17, 3.0.0-3.0.12 and 3.1.0-3.1.5, it is possible for a user to provide specially crafted HTTP requests that may cause a denial-of-service (DoS) condition. Specifically, an application is vulnerable when all of the following are true: \* the application uses Spring MVC or Spring WebFlux \* org.springframework.boot:spring-boot-actuator is on the classpath.

#### **CVE-2023-34462: netty-handler SniHandler 16MB allocation**

Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers & clients. The `SniHandler` can allocate up to 16MB of heap for each channel during the TLS handshake. When the handler or the channel does not have an idle timeout, it can be used to make a TCP server using the `SniHandler` to allocate 16MB of heap. The `SniHandler` class is a handler that waits for the TLS handshake to configure a `SslHandler` according to the indicated server name by the `ClientHello` record. For this matter it allocates a `ByteBuf` using the value defined in the `ClientHello` record. Normally the value of the packet should be smaller than the handshake packet but there are no checks done here and the way the code is written, it is possible to craft a packet that makes the `SslClientHelloHandler`. This vulnerability has been fixed in version 4.1.94.Final.

#### **CVE-2023-35116**

jackson-databind through 2.15.2 allows attackers to cause a denial of service or other unspecified impact via a crafted object that uses cyclic dependencies. NOTE: the vendor's perspective is that this is not a valid vulnerability report, because the steps of constructing a cyclic data structure and trying to serialize it cannot be achieved by an external attacker.

#### **CVE-2023-35887: Apache MINA SSHD: Information disclosure bugs with RootedFilesystem**

Exposure of Sensitive Information to an Unauthorized Actor vulnerability in Apache Software Foundation Apache MINA. In SFTP servers implemented using Apache MINA SSHD that use a RootedFilesystem, logged users may be able to discover "exists/does not exist" information about items outside the rooted tree via paths including parent navigation ("..") beyond the root, or involving symlinks. This issue affects Apache MINA: from 1.0 before 2.10. Users are recommended to upgrade to 2.10

#### **CVE-2023-36414: Azure Identity SDK Remote Code Execution Vulnerability**

Azure Identity SDK Remote Code Execution Vulnerability



**CVE-2023-39017**

quartz-jobs 2.3.2 and below was discovered to contain a code injection vulnerability in the component `org.quartz.jobs.ee.jms.SendQueueMessageJob.execute`. This vulnerability is exploited via passing an unchecked argument. NOTE: this is disputed by multiple parties because it is not plausible that untrusted user input would reach the code location where injection must occur.

**CVE-2023-39196: Apache Ozone: Missing mutual TLS authentication in one of the service internal Ozone Storage Container Manager endpoints**

Improper Authentication vulnerability in Apache Ozone. The vulnerability allows an attacker to download metadata internal to the Storage Container Manager service without proper authentication. The attacker is not allowed to do any modification within the Ozone Storage Container Manager service using this vulnerability. The accessible metadata does not contain sensitive information that can be used to exploit the system later on, and the accessible data does not make it possible to gain access to actual user data within Ozone. This issue affects Apache Ozone: 1.2.0 and subsequent releases up until 1.3.0. Users are recommended to upgrade to version 1.4.0, which fixes the issue.

**CVE-2023-39410: Apache Avro Java SDK: Memory when deserializing untrusted data in Avro Java SDK**

When deserializing untrusted or corrupted data, it is possible for a reader to consume memory beyond the allowed constraints and thus lead to out of memory on the system. This issue affects Java applications using Apache Avro Java SDK up to and including 1.11.2. Users should update to `apache-avro` version 1.11.3 which addresses this issue.

**CVE-2023-44487**

The HTTP/2 protocol allows a denial of service (server resource consumption) because request cancellation can reset many streams quickly, as exploited in the wild in August through October 2023.

**CVE-2023-49145: Apache NiFi: Improper Neutralization of Input in Advanced User Interface for Jolt**

Apache NiFi 0.7.0 through 1.23.2 include the JoltTransformJSON Processor, which provides an advanced configuration user interface that is vulnerable to DOM-based cross-site scripting. If an authenticated user, who is authorized to configure a JoltTransformJSON Processor, visits a crafted URL, then arbitrary JavaScript code can be executed within the session context of the authenticated user. Upgrading to Apache NiFi 1.24.0 or 2.0.0-M1 is the recommended mitigation.

**CVE-2023-50291: Apache Solr: System Property redaction logic inconsistency can lead to leaked passwords**

Insufficiently Protected Credentials vulnerability in Apache Solr. This issue affects Apache Solr: from 6.0.0 through 8.11.2, from 9.0.0 before 9.3.0. One of the two endpoints that publishes the Solr process' Java system properties, `/admin/info/properties`, was only setup to hide system properties that had "password" contained in the name. There are a number of sensitive system properties, such as "basicauth" and "aws.secretKey" do not contain "password", thus their values were published via the `/admin/info/properties` endpoint. This endpoint populates the list of System Properties on the home screen of the Solr Admin page, making the exposed credentials visible in the UI. This `/admin/info/properties` endpoint is protected under the "config-read" permission. Therefore, Solr Clouds with Authorization enabled will only be vulnerable through logged-in users that have the "config-read" permission. Users are recommended to upgrade to version 9.3.0 or 8.11.3, which fixes the issue. A single option now controls hiding Java system property for all endpoints, `-Dsolr.hiddenSysProps`. By default all known sensitive properties are hidden (including `-Dbasicauth`), as well as any property with a name containing "secret" or "password". Users who cannot upgrade can also use the following Java system property to fix the issue: `-Dsolr.redaction.system.pattern=.*(password|secret|basicauth).*`

**CVE-2023-50292: Apache Solr: Solr Schema Designer blindly "trusts" all configsets, possibly leading to RCE by unauthenticated users**

Incorrect Permission Assignment for Critical Resource, Improper Control of Dynamically-Managed Code Resources vulnerability in Apache Solr. This issue affects Apache Solr: from 8.10.0 through 8.11.2, from 9.0.0 before 9.3.0. The Schema Designer was introduced to allow users to more easily configure and test new Schemas and configSets. However, when the feature was created, the "trust" (authentication) of these configSets was not considered. External library loading is only available to configSets that are "trusted" (created by authenticated users), thus non-authenticated users are unable to perform Remote Code Execution. Since the Schema Designer loaded configSets without taking their "trust" into account, configSets that were created by unauthenticated users were allowed to load external libraries when used in the Schema Designer. Users are recommended to upgrade to version 9.3.0, which fixes the issue.

**CVE-2023-50298: Apache Solr: Solr can expose ZooKeeper credentials via Streaming Expressions**

Exposure of Sensitive Information to an Unauthorized Actor vulnerability in Apache Solr. This issue affects Apache Solr: from 6.0.0 through 8.11.2, from 9.0.0 before 9.4.1. Solr Streaming Expressions allows users to extract data from other Solr Clouds, using a "zkHost" parameter. When the original SolrCloud is set up to use ZooKeeper credentials and ACLs, they will be sent to whatever "zkHost" the user provides. An attacker could setup a server to mock ZooKeeper, that accepts ZooKeeper requests with credentials and ACLs and extracts the sensitive information, then sends a streaming expression using the mock server's address in "zkHost". Streaming Expressions are exposed via the "/streaming" handler, with "read" permissions. Users are recommended to upgrade to version 8.11.3 or 9.4.1, which fix the issue. From these versions on, only zkHost values that have the same server address (regardless of chroot), will use the given ZooKeeper credentials and ACLs when connecting.

**CVE-2023-50386: Apache Solr: Backup/Restore APIs allow for deployment of executables in malicious ConfigSets**

Improper Control of Dynamically-Managed Code Resources, Unrestricted Upload of File with Dangerous Type, Inclusion of Functionality from Untrusted Control Sphere vulnerability in Apache Solr. This issue affects Apache Solr: from 6.0.0 through 8.11.2, from 9.0.0 before 9.4.1. In the affected versions, Solr ConfigSets accepted Java jar and class files to be uploaded through the ConfigSets API. When backing up Solr Collections, these configSet files would be saved to disk when using the LocalFileSystemRepository (the default for backups). If the backup was saved to a directory that Solr uses in its ClassPath/ClassLoaders, then the jar and class files would be available to use with any ConfigSet, trusted or untrusted. When Solr is run in a secure way (Authorization enabled), as is strongly suggested, this vulnerability is limited to extending the Backup permissions with the ability to add libraries. Users are recommended to upgrade to version 8.11.3 or 9.4.1, which fix the issue. In these versions, the following protections have been added: \* Users are no longer able to upload files to a configSet that could be executed via a Java ClassLoader. \* The Backup API restricts saving backups to directories that are used in the ClassLoader.

**CVE-2023-50572**

An issue in the component GroovyEngine.execute of jline-groovy v3.24.1 allows attackers to cause an OOM (OutOfMemory) error.

**CVE-2023-52428**

In Connect2id Nimbus JOSE+JWT before 9.37.2, an attacker can cause a denial of service (resource consumption) via a large JWE p2c header value (aka iteration count) for the PasswordBasedDecrypter (PBKDF2) component.

**CVE-2024-22243: Spring Framework URL Parsing with Host Validation**

Applications that use UriComponentsBuilder to parse an externally provided URL (e.g. through a query parameter) AND perform validation checks on the host of the parsed URL may be vulnerable to an open redirect <https://cwe.mitre.org/data/definitions/601.html> attack or to a SSRF attack if the URL is used after passing validation checks.

**CVE-2024-25710: Apache Commons Compress: Denial of service caused by an infinite loop for a corrupted DUMP file**



Loop with Unreachable Exit Condition ('Infinite Loop') vulnerability in Apache Commons Compress. This issue affects Apache Commons Compress: from 1.3 through 1.25.0. Users are recommended to upgrade to version 1.26.0 which fixes the issue.

**CVE-2024-26308: Apache Commons Compress: OutOfMemoryError unpacking broken Pack200 file**

Allocation of Resources Without Limits or Throttling vulnerability in Apache Commons Compress. This issue affects Apache Commons Compress: from 1.21 before 1.26. Users are recommended to upgrade to version 1.26, which fixes the issue.

**CVE-2020-12668**

Jinjava before 2.5.4 allows access to arbitrary classes by calling Java methods on objects passed into a Jinjava context. This could allow for abuse of the application class loader, including Arbitrary File Disclosure.

**Related Information**

[Support matrix](#)