

Cloudera Flow Management 2.1.4

Cloudera Flow Management Release Notes

Date published: 2019-06-26

Date modified: 2022-05-12

CLOUDERA

<https://docs.cloudera.com/>

Legal Notice

© Cloudera Inc. 2025. 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

What's new in this release?	4
CFM component versions	5
Support matrix	6
System requirements	6
Supported operating systems	7
Supported NiFi Registry databases	7
Supported NiFi processors	8
Supported NiFi controller services	11
Supported NiFi reporting tasks	12
Components supported by partners	13
Download locations	13
Unsupported features	15
Technical preview features	15
Unsupported customizations	16
Behavioral changes	16
Known issues	17
Fixed issues	19
Fixed Common Vulnerabilities and Exposures	27

What's new in this release?

Learn about the new functionalities and improvements in Cloudera Flow Management (CFM) and how these new features benefit you.

Service Pack for CFM 2.1.4

On June 28, 2022, Cloudera released CFM 2.1.4 SP1 to fix some issues. If you previously upgraded to CFM 2.1.4.0, upgrade to CFM 2.1.4.1000-5 to access the latest version of the software. You can find the new download links in [Download locations](#). For more information about the issues fixed in CFM 2.1.4 SP1, see [Fixed issues](#) and [Fixed CVEs](#).

CFM 2.1.4

CFM 2.1.4 is based on Apache NiFi 1.16.0 and includes significant improvements and fixes. The most important new features and improvements of the release are:

The flow.xml.gz file replaced with flow.json.gz

CFM 2.1.4 NiFi started using a flow.json.gz instead of the flow.xml.gz file used in all previous versions. On startup, NiFi converts the existing flow.xml.gz to a flow.json.gz file. If a flow.json.gz already exists, NiFi ignores the flow.xml.gz on startup. Changes made on the canvas will currently be written to both the new flow.json.gz version as well as the flow.xml.gz file; however, only the flow.json.gz is loaded on NiFi startup.

Supported NiFi Registry databases changed

Following some security related changes and dependency upgrades, the list of supported databases for the NiFi Registry component has been updated. Make sure you are using a supported version before upgrading to CFM 2.1.4. For more information, see *Supported NiFi Registry databases*.

Kafka 0.x processors removed

The Kafka processors previously distributed for interacting with Kafka 0.x clusters have been removed. It means that these processors are no longer provided and supported by Cloudera. It is recommended to switch to the processors making use of Kafka 2.x versions.

- ConsumeKafka
- ConsumeKafka_0_10
- ConsumeKafka_0_11
- ConsumeKafkaRecord_0_10
- ConsumeKafkaRecord_0_11
- GetKafka
- PublishKafka
- PublishKafka_0_10
- PublishKafka_0_11
- PublishKafkaRecord_0_10
- PublishKafkaRecord_0_11
- PutKafka

For more information, see *Supported NiFi processors*.

Some Elasticsearch processors removed

Elasticsearch processors that were leveraging the Elasticsearch 2.0 library and the deprecated Transport Client have been removed. It means that the below processors are no longer provided and supported by Cloudera. It is recommended to switch to the processors interacting with Elasticsearch over HTTP.

- FetchElasticsearch

- PutElasticsearch

For more information, see *Supported NiFi processors*.

Retry at framework level

This feature provides the ability to define a retry strategy for the flowfiles going into a given relationship for each processor.

Clustering model and flow definition reconciliation

A disconnected node in a NiFi cluster will no longer cause the UI to be read-only for flow changes and a reconciliation will be executed on the flow definition when the node rejoins the cluster.

Snowflake Connection Pool controller service

This controller service can be used in combination with the JDBC processors to easily interact with Snowflake for both pulling and pushing data.

Query Salesforce Object processor

A new processor is available that can be used to ingest data from Salesforce with incremental changes.

OAuth 2 integration with InvokeHTTP processor

It is now possible to configure an OAuth2 token provider with InvokeHTTP to easily interact with services requiring OAuth2 authentication.

Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi processors](#)

CFM component versions

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



Note: NiFi works with the version of NiFi Registry shipped with your version of CFM or later.

CFM 2.1.4 SP1

- Apache NiFi 1.16.0.2.1.4.1000
- Apache NiFi Registry 1.16.0.2.1.4.1000

CFM 2.1.4

- Apache NiFi 1.16.0.2.1.4.0
- Apache NiFi Registry 1.16.0.2.1.4.0

CFM 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 version are unified in the 1.15.x release.

CFM 2.1.2

- Apache NiFi 1.13.2.2.1.2.0

- Apache NiFi Registry 0.8.0.2.1.2.0

CFM 2.1.1

- Apache NiFi 1.13.2.2.1.1.0
- Apache NiFi Registry 0.8.0.2.1.1.0

CFM 2.0.4

- Apache NiFi 1.11.4
- Apache NiFi Registry 0.6.0

CFM 2.0.1

- Apache NiFi 1.11.4
- Apache NiFi Registry 0.6.0

Support matrix

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

System requirements

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

Supported versions of CDP

CFM 2.1.4 supports the following versions of CDP Private Cloud Base:

- CDP 7.1.7
- CDP 7.1.7 SP1
- CDP 7.1.7 SP2
- CDP 7.1.8 with Cloudera Manager 7.7.1



Note: If you need Python 3 support, upgrade to CFM 2.1.5 or higher on CDP Private Cloud Base 7.1.8 or higher.



Note: CFM provides a set of monitoring features when managed by Cloudera Manager. These features are available and working, when you use CFM with Cloudera Manager 7.6.1 or higher.

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

Other system requirements

- ZooKeeper – You must install the ZooKeeper service available with your CDP Private Cloud Base cluster.
- 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 (CFM).

Operating system	Versions
CentOS	• 7.6 • 7.7 • 7.8 • 7.9 • 8.2 • 8.4
RHEL	• 7.6 • 7.7 • 7.8 • 7.9 • 8.2 • 8.4 • 8.6
SLES	• 12 SP5
Ubuntu	• 18.04 • 20.04
Windows	• 10 • Server 2016 • Server 2019



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 (CFM) 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.

CFM 2.1.4 supports the following NiFi processors:

AttributesToCSV	GetElasticsearch	PutDynamoDB1
AttributesToJSON	GetFile	PutDynamoDBRecord
Base64EncodeContent	GetFTP	PutElasticsearchHttp1
CalculateRecordStats	GetHBase	PutElasticsearchHttpRecord
CaptureChangeMySQL	GetHDFS	PutElasticsearchJson
CompressContent1,2	GetHDFSFileInfo	PutElasticsearchRecord
ConnectWebSocket	GetHDFSSequenceFile	PutEmail1
ConsumeAMQP	GetHTMLElement	PutFile
ConsumeAzureEventHub	GetHTTP	PutFTP
ConsumeEWS	GetIgniteCache	PutGCSObject
ConsumeGCPubSub	GetJMSQueue	PutGridFS
ConsumeGCPubSubLite	GetJMSTopic	PutHBaseCell1
ConsumeJMS	GetMongoRecord	PutHBaseJSON
ConsumeKafka_1_0	GetSFTP	PutHBaseRecord
ConsumeKafka_2_0	GetSNMP	PutHDFS
ConsumeKafka_2_6	GetSolr	PutHive3QL
ConsumeKafka2CDP	GetSplunk	PutHive3Streaming
ConsumeKafka2RecordCDP	GetSQS	PutHiveQL
ConsumeKafkaRecord_1_0	GetTCP	PutHiveStreaming
ConsumeKafkaRecord_2_0	GetTwitter	PutHTMLElement
ConsumeKafkaRecord_2_6	HandleHttpRequest	PutInfluxDB
ConsumeKinesisStream	HandleHttpResponse	PutJMS
ConsumeMQTT1	HashAttribute	PutKinesisFirehose
ConsumeWindowsEventLog	HashContent	PutKinesisStream
ControlRate	IdentifyMimeType	PutKudu
ConvertAvroSchema	InvokeAWSGatewayApi	PutLambda
ConvertAvroToJSON	InvokeGRPC	PutMongoRecord
ConvertAvroToORC	InvokeHTTP	PutORC
ConvertAvroToParquet	InvokeScriptedProcessor	PutParquet

ConvertCharacterSet	JoinEnrichment	PutRecord
ConvertCSVToAvro	JoltTransformJSON	PutRiemann
ConvertJSONToAvro	JoltTransformRecord	PutS3Object
ConvertJSONToSQL	JsonQueryElasticsearch	PutSFTP
ConvertRecord	ListAzureBlobStorage	PutSNS
CreateHadoopSequenceFile	ListAzureBlobStorage_v12	PutSolrContentStream
CryptographicHashAttribute	ListAzureDataLakeStorage	PutSolrRecord
CryptographicHashContent	ListCDPObjectStore	PutSplunk
DecryptContentPGP	ListDatabaseTables	PutSplunkHTTP1
DeduplicateRecord	ListenFTP	PutSQL
DeleteAzureBlobStorage	ListenGRPC	PutSQS
DeleteAzureBlobStorage_v12	ListenHTTP	PutSyslog
DeleteAzureDataLakeStorage	ListenRELP	PutTCP
DeleteByQueryElasticsearch	ListenSyslog	PutUDP
DeleteCDPObjectStore	ListenTCP	PutWebSocket1
DeleteDynamoDB	ListenTCPRecord	QueryCassandra
DeleteGCXObject	ListenTrapSNMP	QueryDatabaseTable
DeleteGridFS	ListenUDP	QueryDatabaseTableRecord
DeleteHBaseCells	ListenUDPRecord	QueryElasticsearchHttp
DeleteHBaseRow	ListenWebSocket	QueryRecord
DeleteHDFS	ListFile	QuerySalesforceObject
DeleteS3Object	ListFTP	QuerySolr
DeleteSQS	ListGCSBucket	QuerySplunkIndexingStatus
DetectDuplicate	ListHDFS	QueryWhois
DistributeLoad	ListS3	ReplaceText1
DuplicateFlowFile	ListSFTP	ReplaceTextWithMapping
EncryptContent2	LogAttribute	ResizeImage
EncryptContentPGP	LogMessage	RetryFlowFile
EnforceOrder	LookupAttribute	RouteHL7
EvaluateJsonPath	LookupRecord	RouteOnAttribute
EvaluateXPath	MergeContent1	RouteOnContent
EvaluateXQuery	MergeRecord	RouteText
ExecuteGroovyScript	ModifyHTMLElement	SampleRecord1
ExecuteInfluxDBQuery	MonitorActivity	ScanAccumulo
ExecuteProcess	MoveAzureDataLakeStorage	ScanAttribute
ExecuteScript	Notify	ScanContent
ExecuteSQL	PaginatedJsonQueryElasticsearch	ScanHBase
ExecuteSQLRecord	ParseCEF	ScriptedFilterRecord
ExecuteStateless	ParseEvtx	ScriptedPartitionRecord
ExecuteStreamCommand	ParseSyslog	ScriptedTransformRecord

ExtractAvroMetadata	PartitionRecord	ScriptedValidateRecord
ExtractGrok	PostHTTP	ScrollElasticsearchHttp
ExtractHL7Attributes	PublishAMQP1	SearchElasticsearch
ExtractImageMetadata	PublishGCPubSub1	SegmentContent
ExtractText	PublishGCPubSubLite	SelectHive3QL
FetchAzureBlobStorage	PublishJMS1	SelectHiveQL
FetchAzureBlobStorage_v12	PublishKafka_1_0	SendTrapSNMP
FetchAzureDataLakeStorage	PublishKafka_2_0	SetSNMP
FetchCDPObjectStore	PublishKafka_2_6	SignContentPGP
FetchDistributedMapCache	PublishKafka2CDP	SplitAvro1
FetchElasticsearchHttp	PublishKafka2RecordCDP	SplitContent1
FetchFile	PublishKafkaRecord_1_0	SplitJson1
FetchFTP	PublishKafkaRecord_2_0	SplitRecord
FetchGCSObject	PublishKafkaRecord_2_6	SplitText1
FetchGridFS	PublishMQTT1	SplitXml1
FetchHBaseRow	PutAccumuloRecord	TagS3Object
FetchHDFS	PutAzureBlobStorage	TailFile
FetchParquet	PutAzureBlobStorage_v12	TransformXml
FetchS3Object	PutAzureCosmosDBRecord1	UnpackContent
FetchSFTP	PutAzureDataLakeStorage	UpdateAttribute
FlattenJson	PutAzureEventHub1	UpdateByQueryElasticsearch
ForkEnrichment	PutAzureQueueStorage	UpdateCounter
ForkRecord	PutBigQueryBatch	UpdateHive3Table
GenerateFlowFile	PutBigQueryStreaming1	UpdateHiveTable
GenerateTableFetch	PutCassandraQL1	UpdateRecord
GeoEnrichIP	PutCassandraRecord	ValidateCsv
GeoEnrichIPRecord	PutCDPObjectStore	ValidateRecord
GeohashRecord	PutCloudWatchMetric	ValidateXml
GetAzureEventHub	PutCouchbaseKey1	VerifyContentPGP
GetAzureQueueStorage	PutDatabaseRecord	Wait
GetCouchbaseKey1	PutDistributedMapCache	YandexTranslate

Footnotes

- 1 – indicates a memory intensive processor
- 2 – indicates a CPU intensive processor

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 (CFM) 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	HiveConnectionPool
ActionHandlerLookup	HortonworksSchemaRegistry
ADLSCredentialsControllerService	IPFIXReader
ADLSIDBrokerCloudCredentialsProviderControllerService	IPLookupService
AlertHandler	JASN1Reader
AvroReader	JMSConnectionFactoryProvider
AvroRecordSetWriter	JndiJmsConnectionFactoryProvider
AvroSchemaRegistry	JsonPathReader
AWSCredentialsProviderControllerService	JsonRecordSetWriter
AWSIDBrokerCloudCredentialsProviderControllerService	JsonTreeReader
AzureBlobIDBrokerCloudCredentialsProviderControllerService	KafkaRecordSink_1_0
AzureCosmosDBClientService	KafkaRecordSink_2_0
AzureStorageCredentialsControllerService	KafkaRecordSink_2_6
AzureStorageCredentialsControllerService_v12	KerberosKeytabUserService
AzureStorageCredentialsControllerServiceLookup	KerberosPasswordUserService
CassandraDistributedMapCache	KerberosTicketCacheUserService
CassandraSessionProvider	KeytabCredentialsService
CEFReader	KuduLookupService
CouchbaseClusterService	LoggingRecordSink
CouchbaseKeyValueLookupService	LogHandler
CouchbaseMapCacheClient	MongoDBControllerService
CouchbaseRecordLookupService	MongoDBLookupService
CSVReader	ParquetReader
CSVRecordLookupService	ParquetRecordSetWriter
CSVRecordSetWriter	PrometheusRecordSink
DatabaseRecordLookupService	ReaderLookup
DatabaseRecordSink	RecordSetWriterLookup
DBCPCConnectionPool	RecordSinkHandler
DBCPCConnectionPoolLookup	RecordSinkServiceLookup
DistributedMapCacheClientService	RedisConnectionPoolService
DistributedMapCacheLookupService	RedisDistributedMapCacheClientService
DistributedMapCacheServer	RestLookupService
DistributedSetCacheClientService	ScriptedActionHandler

DistributedSetCacheServer	ScriptedLookupService
EasyRulesEngineProvider	ScriptedReader
EasyRulesEngineService	ScriptedRecordSetWriter
ElasticSearchClientServiceImpl	ScriptedRecordSink
ElasticSearchLookupService	ScriptedRulesEngine
ElasticSearchStringLookupService	SimpleDatabaseLookupService
EmailRecordSink	SimpleKeyValueLookupService
EmbeddedHazelcastCacheManager	SimpleScriptedLookupService
ExpressionHandler	SiteToSiteReportingRecordSink
ExternalHazelcastCacheManager	SnowflakeComputingConnectionPool
FreeFormTextRecordSetWriter	StandardHttpContextMap
GCPCredentialsControllerService	StandardOAuth2AccessTokenProvider
GrokReader	StandardPGPPrivateKeyService
HadoopDBCPCConnectionPool	StandardPGPPublicKeyService
HazelcastMapCacheClient	StandardProxyConfigurationService
HBase_1_1_2_ClientMapCacheService	StandardRestrictedSSLContextService
HBase_1_1_2_ClientService	StandardS3EncryptionService
HBase_1_1_2_ListLookupService	StandardSSLContextService
HBase_1_1_2_RecordLookupService	Syslog5424Reader
HBase_2_ClientMapCacheService	SyslogReader
HBase_2_ClientService	VolatileSchemaCache
HBase_2_RecordLookupService	WindowsEventLogReader
Hive3ConnectionPool	XMLReader
	XMLRecordSetWriter

Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi processors](#)

[Supported NiFi reporting tasks](#)

[Components supported by partners](#)

Supported NiFi reporting tasks

Cloudera Flow Management (CFM) 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.

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.

- AmbariReportingTask
- ControllerStatusReportingTask
- MetricsEventReportingTask
- MonitorDiskUsage

- MonitorMemory
- PrometheusReportingTask
- QueryNiFiReportingTask
- ReportLineageToAtlas
- ScriptedReportingTask
- SiteToSiteBulletinReportingTask
- SiteToSiteMetricsReportingTask
- SiteToSiteProvenanceReportingTask
- SiteToSiteStatusReportingTask

Related Information

[Supported NiFi Registry databases](#)

[Supported NiFi processors](#)

[Supported NiFi controller services](#)

[Components supported by partners](#)

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 (v1.15.2)
- ConsumePulsarRecord (v1.15.2)
- PublishPulsar (v1.15.2)
- PublishPulsarRecord (v1.15.2)

Controller services supported by partners

- PulsarClientAthenzAuthenticationService (v1.15.2)
- PulsarClientJwtAuthenticationService (v1.15.2)
- PulsarClientOAuthAuthenticationService (v1.15.2)
- PulsarClientTlsAuthenticationService (v1.15.2)
- StandardPulsarClientService (v1.15.2)

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 (CFM) software artifacts from the Cloudera Archive. There are different CFM artifacts for different operating systems, standalone components, and Windows files.

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

**Note:**

You must have credentials to download CFM 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 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	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/parcel/manifest.json
Parcel	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/parcel/CFM-2.1.4.1000-5-el7.parcel
Parcel sha file	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/parcel/CFM-2.1.4.1000-5-el7.parcel.sha

Table 2: RHEL/CentOS 8

File	Location
Manifest	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat8/yum/tars/parcel/manifest.json
Parcel	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat8/yum/tars/parcel/CFM-2.1.4.1000-5-el8.parcel
Parcel sha file	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat8/yum/tars/parcel/CFM-2.1.4.1000-5-el8.parcel.sha

Table 3: SLES 12

File	Location
Manifest	https://archive.cloudera.com/p/cfm2/2.1.4.1000/sles12/yum/tars/parcel/manifest.json
Parcel	https://archive.cloudera.com/p/cfm2/2.1.4.1000/sles12/yum/tars/parcel/CFM-2.1.4.1000-5-sles12.parcel
Parcel sha file	https://archive.cloudera.com/p/cfm2/2.1.4.1000/sles12/yum/tars/parcel/CFM-2.1.4.1000-5-sles12.parcel.sha

Table 4: Ubuntu 18

File	Location
Manifest	https://archive.cloudera.com/p/cfm2/2.1.4.1000/ubuntu18/apt/tars/parcel/manifest.json
Parcel	https://archive.cloudera.com/p/cfm2/2.1.4.1000/ubuntu18/apt/tars/parcel/CFM-2.1.4.1000-5-bionic.parcel
Parcel sha file	https://archive.cloudera.com/p/cfm2/2.1.4.1000/ubuntu18/apt/tars/parcel/CFM-2.1.4.1000-5-bionic.parcel.sha

Table 5: Ubuntu 20

File	Location
Manifest	https://archive.cloudera.com/p/cfm2/2.1.4.1000/ubuntu20/apt/tars/parcel/manifest.json
Parcel	https://archive.cloudera.com/p/cfm2/2.1.4.1000/ubuntu20/apt/tars/parcel/CFM-2.1.4.1000-5-focal.parcel
Parcel SHA file	https://archive.cloudera.com/p/cfm2/2.1.4.1000/ubuntu20/apt/tars/parcel/CFM-2.1.4.1000-5-focal.parcel.sha

Table 6: CSD files (OS agnostic)

File	Location
NiFi	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/parcel/NIFI-1.16.0.2.1.4.1000-5.jar

File	Location
NiFi Registry	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/parcel/NIFIREGISTRY-1.16.0.2.1.4.1000-5.jar

Table 7: Standalone components (OS agnostic)

File	Location
NiFi (.tar.gz)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-1.16.0.2.1.4.1000-5-bin.tar.gz
NiFi (.tar.gz.sha256)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-1.16.0.2.1.4.1000-5-bin.tar.gz.sha256
NiFi (.zip)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-1.16.0.2.1.4.1000-5-bin.zip
NiFi (.zip.sha256)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-1.16.0.2.1.4.1000-5-bin.zip.sha256
NiFi Registry (.tar.gz)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-registry-1.16.0.2.1.4.1000-5-bin.tar.gz
NiFi Registry (.tar.gz.sha256)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-registry-1.16.0.2.1.4.1000-5-bin.tar.gz.sha256
NiFi Registry (.zip)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-registry-1.16.0.2.1.4.1000-5-bin.zip
NiFi Registry (.zip.sha256)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-registry-1.16.0.2.1.4.1000-5-bin.zip.sha256
NiFi Toolkit (.tar.gz)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.16.0.2.1.4.1000-5-bin.tar.gz
NiFi Toolkit (.tar.gz.sha256)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.16.0.2.1.4.1000-5-bin.tar.gz.sha256
NiFi Toolkit (.zip)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.16.0.2.1.4.1000-5-bin.zip
NiFi Toolkit (.zip.sha256)	https://archive.cloudera.com/p/cfm2/2.1.4.1000/redhat7/yum/tars/nifi/nifi-toolkit-1.16.0.2.1.4.1000-5-bin.zip.sha256

Table 8: Windows files

File	Location
NiFi MSI	https://archive.cloudera.com/p/cfm2/2.1.4.1000/windows/nifi-2.1.4.1000-5.msi
NiFi MSI SHA file	https://archive.cloudera.com/p/cfm2/2.1.4.1000/windows/nifi-2.1.4.1000-5.msi.sha256

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.

Technical preview features

The following features are available in Cloudera Flow Management (CFM) 2.1.4 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](#).

- The following rules engine and handlers controller services:
 - EasyRulesEngineService
 - EasyRulesEngineProvider
 - ScriptedRulesEngine
 - ActionHandlerLookup
 - AlertHandler
 - ExpressionHandler
 - LogHandler
 - RecordSinkHandler
 - ScriptedActionHandler

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.

Behavioral changes

Learn about the change in behavior in Cloudera Flow Management (CFM) 2.1.4.

- CFM 2.1.4 NiFi started using a flow.json.gz instead of the flow.xml.gz file used in all previous versions. On startup, NiFi converts the existing flow.xml.gz to a flow.json.gz file. If a flow.json.gz already exists, NiFi ignores the flow.xml.gz on startup. Changes made on the canvas will currently be written to both the new flow.json.gz version as well as the flow.xml.gz file; however, only the flow.json.gz is loaded on NiFi startup.
- Following some security related changes and dependency upgrades, the list of supported databases for the NiFi Registry component has changed. Make sure you are using a supported version before upgrading to CFM 2.1.4. For more information, see the *Support matrix*.
- The Kafka processors previously distributed for interacting with Kafka 0.x clusters have been removed. It means that these processors are no longer provided nor supported by Cloudera and it is recommended to switch to the processors making use of Kafka 2.x versions.
 - ConsumeKafka
 - ConsumeKafka_0_10
 - ConsumeKafka_0_11
 - ConsumeKafkaRecord_0_10
 - ConsumeKafkaRecord_0_11
 - GetKafka
 - PublishKafka
 - PublishKafka_0_10
 - PublishKafka_0_11
 - PublishKafkaRecord_0_10
 - PublishKafkaRecord_0_11
 - PutKafka
- Elasticsearch processors that were leveraging the Elasticsearch 2.0 library and the deprecated Transport Client have been removed. It means that the below processors are no longer provided and supported by Cloudera. It is recommended to switch to the processors interacting with Elasticsearch over HTTP.
 - FetchElasticsearch
 - PutElasticsearch

Related Information

[Support matrix](#)

Known issues

Review the list of known issues in Cloudera Flow Management (CFM) 2.1.4.

Special characters in Keystore/Truststore passwords

If there are special characters in the passwords of the truststores/keystores, the normal operation of NiFi and its integration with Cloudera Manager (command and control, monitoring, etc) is affected.

Update the passwords using only [A-Z a-z 0-9] characters or upgrade to CFM 2.1.5. You can also file a support case to get a hotfix from Cloudera Support.

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.

KafkaRecordSink puts multiple records in one message

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

For more information, see [NIFI-8326](#).

There is no workaround for this issue.

Kudu Client

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

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`.

Parameter Context inheritance may be lost during NiFi restart

When restarting NiFi, the inheritance between parameter contexts may be lost under specific conditions.

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

For more information, see [NIFI-10096](#).

Cloudera recommends to upgrade to the latest version or to request a HOTFIX through the Support Portal.

KeyStoreException: placeholder not found

After an upgrade, NiFi may fail to start, displaying 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.

Technical Service Bulletins

TSB 2022-580: NiFi Processors cannot write to content repository

If the content repository disk is filled more than 50% (or any other value that is set in `nifi.properties` for `nifi.content.repository.archive.max.usage.percentage`), and if there is no data in the content

repository archive, the following warning message can be found in the logs: "Unable to write flowfile content to content repository container default due to archive file size constraints; waiting for archive cleanup". This would block the processors and no more data is processed.

This appears to only happen if there is already data in the content repository on startup that needs to be archived, or if the following message is logged: "Found unknown file XYZ in the File System Repository; archiving file".

Upstream JIRA

- [NIFI-10023](#)
- [NIFI-9993](#)

Knowledge article

For the latest update on this issue see the corresponding Knowledge article: [TSB 2022-580: NiFi Processors cannot write to content repository](#)

TSB 2022-589: CVE-2022-33140 Apache NiFi ShellUserGroupProvider Vulnerability

The optional ShellUserGroupProvider in Apache NiFi 1.10.0 to 1.16.2 and Apache NiFi Registry 0.6.0 to 1.16.2 does not neutralize arguments for group resolution commands, allowing injection of operating system commands on Linux and macOS platforms. The ShellUserGroupProvider is not included in the default configuration. Command injection requires ShellUserGroupProvider to be one of the enabled User Group Providers (UGP) in the Authorizers configuration. Command injection also requires an authenticated user with elevated privileges. Apache NiFi requires an authenticated user with authorization to modify access policies in order to execute the command. Apache NiFi Registry requires an authenticated user with authorization to read user groups in order to execute the command. The resolution removes command formatting based on user-provided arguments.

Knowledge article

For the latest update on this issue see the corresponding Knowledge article: [TSB 2022-589: CVE-2022-33140 Apache NiFi ShellUserGroupProvider Vulnerability](#)

Fixed issues

Review the list of resolved issues.



Note: On June 28, 2022, Cloudera released a Service Pack for CFM 2.1.4 to fix some issues. If you previously upgraded to CFM 2.1.4.0, upgrade to CFM 2.1.4.1000-5 to access the latest version of the software. You can find the new download links in [Download locations](#).

Issues fixed in CFM 2.1.4.1000-5

- NIFI-10036: Corrected Elasticsearch Client Service to prefix paths for all requests
- NIFI-9944: Fixed issue with using modules in InvokeScriptedProcessor
- NIFI-10023: Ensure that any files that are archived upon startup properly increment the archival count in the content repo
- NIFI-10020: Corrected ClassNotFoundException in EmailNotificationService
- NIFI-10026: Added jakarta.activation runtime dependency to bootstrap
- NIFI-10031: Downgraded Jakarta Mail from 2.0.1 to 1.6.7
- NIFI-10001: Enabling collection of Controller Services logic
- NIFI-9991: The asn1 StandardRecordModelIteratorProvider's iterator returns only the first object read from the input stream
- NIFI-9470: Allow creation of Parameter Context without any Inherited Parameter Contexts
- NIFI-9990: Improved FTP 550 file unavailable handling
- NIFI-10074: Fix potential UnsupportedOperationException in PutSmbFile.customValidate

- NIFI-10088: Set SSH Transport Timeout using SFTP Data Timeout property
- NIFI-10096: Corrected nested inherited Parameter Context loading
- NIFI-10114: Removed Single Refresh from ShellUserGroupProvider
- [TSB 2022-580](#): NiFi Processors cannot write to content repository
- [TSB 2022-589](#): CVE-2022-33140 Apache NiFi ShellUserGroupProvider Vulnerability

Issues fixed in CFM 2.1.4.0

- NIFI-9993 Fixed a bug in initialization when the Content Repo did not properly increment the counter for how many files exist in the archive directories.
- NIFI-9988 Corrected Property Decryption for Authorizers and Providers
- NIFI-9980 Corrected conflicting GCP dependencies
- NIFI-9978 Updated the condition under which the Download Flow menu items are shown
- NIFI-9956 Fixed QuerySalesforceObject schema parsing with nested field strategy
- NIFI-9955 Upgraded ZooKeeper 3.4.14 to 3.5.9
- NIFI-9954 Upgraded Spring Framework from 5.3.18 to 5.3.19
- NIFI-9952 Upgraded Jackson BOM to 2.13.2.20220328
- NIFI-9950 Identify MariaDB database type and use MySQL migration scripts
- NIFI-9949 Corrected intermittent failures in TestPutSFTP
- NIFI-9943 Added Transform Provider to nifi-xml-processing
- NIFI-9942 Removed load test from TestPutUDP
- NIFI-9941 Fixed handling of ReportingTask stop request
- NIFI-9939 Upgraded SSHJ from 0.32.0 to 0.33.0
- NIFI-9935 Upgraded Zip4j from 2.9.1 to 2.10.0
- NIFI-9934 Removed unused Groovy dependency management from nifi-graph-test-clients
- NIFI-9933 Upgraded Apache Ant to 1.10.12
- NIFI-9928 Removed nifi-security-utils from nifi-prometheus-reporting-task
- NIFI-9925 Adjusted PutUDP and SNMP tests to run without preferring IPv4
- NIFI-9924 Corrected text encoding in PutEmail filenames and TestFTP
- NIFI-9923 Corrected SimpleProcessLogger handling of Throwable causes
- NIFI-9917 Improved the tooltip of "delivery guarantee/best effort"
- NIFI-9911 Fixed checking HDFS URI in ListHDFS' Directory property
- NIFI-9910 Call toString() on the provided key in CSVRecordLookupService instead of attempting to cast to a String
- NIFI-9903 When using the 'success' relationship only for LookupRecord, lookup all records until a match is found, to determine the resultant schema. Refactored code to eliminate AbstractRouteRecord and to use an inner interface to clean up code.
- NIFI-9902 Improved handling of charsets when performing string/byte conversions
- NIFI-9901 Removed provided scope from nifi-xml-processing registry-ranger - Added nifi-xml-processing to nifi-commons
- NIFI-9899 Corrected MongoDBLookupService attribute handling
- NIFI-9898 Used CloseableHttpClient and replaced deprecated class/method usages in ProxyAwareTransportFactory in GCP module
- NIFI-9897 Refactored GRPC SSL Configuration
- NIFI-9896 Fixed bug with overwriting Script Body value in InvokeScriptedProcessor
- NIFI-9893 Ensured orderly cluster node removal on node delete via UI
- NIFI-9888 Fixed Azure header datetime format
- NIFI-9887 Set minimum Java build version to 1.8.0-251
- NIFI-9886 Upgraded MongoDB driver to version 4.5.0
- NIFI-9885 When creating ContainerState, ensured that we pass the correct value for the archiveEnabled flag
- NIFI-9884 JacksonCSVRecordReader ignores specified encoding

- NIFI-9883 Updated Ranger assembly exclude property protection modules
- NIFI-9883 Refactored Property Protection using Isolated ClassLoader
- NIFI-9882 Updated HtmlDocumentationWriter to avoid writing null characters
- NIFI-9881 Refactored the JSON services to use Jackson 2
- NIFI-9875 In StandardProcessGroupSynchronizer.updateConnectionDestinations handled special case when output port is moved to a child process group while kept connected.
- NIFI-9874 Consider Flow Difference Type of PARAMETER_CONTEXT_CHANGED an environmental change
- NIFI-9873 Upgraded Kotlin to 1.6.20
- NIFI-9872 Upgraded azure-sdk-bom to 1.2.0 for Sensitive Properties
- NIFI-9871 Corrected Component Stack Trace Logging
- NIFI-9870 Upgraded Jetty from 9.4.45 to 9.4.46
- NIFI-9869 Upgraded Apache POI from 5.0.0 to 5.2.2
- NIFI-9862 Updated JsonTreeReader to read Records from Nested Field
- NIFI-9861 Removed BlockListClassLoader in favor of AllowListClassLoader
- NIFI-9860 Upgraded JRuby from 9.2.6.0 to 9.3.4.0
- NIFI-9859 Upgraded Apache Curator from 5.2.0 to 5.2.1
- NIFI-9858 Refactored nifi-system-test-suite using generated certificates
- NIFI-9854 Included slf4j-api as runtime scope for nifi-stateless-system-test-suite. When constructing the classes to block for the stateless ClassLoader, allow through any classes that come from Java Home's lib/ directory.
- NIFI-9853 Refactored StandardProcessGroupSynchronizer
- NIFI-9852 Upgraded Spring Framework from 5.3.16 to 5.3.18
- NIFI-9850 Added support for multiple expressions to GrokReader
- NIFI-9847: Switched LifecycleState to use a WeakHashMap
- NIFI-9846 Implemented pagination listing for Azure List processors
- NIFI-9845 Fixed NiFi Registry database ITs
- NIFI-9844 Refactored Encryptor tests using JUnit 5
- NIFI-9842 Refactored nifi-commons using JUnit 5
- NIFI-9841 Improved Encryptor test reliability
- NIFI-9839 Fixed a bug in which Process Group would return true for isEmpty even if it has a Controller Service
- NIFI-9838 Added Client Certificate attributes to ListenTCPRecord FlowFiles
- NIFI-9836 Fixed database selection during nifi registry startup
- NIFI-9835 Fixed threading bug in which NioAsyncLoadBalanceClient calls LoadBalanceSession.isComplete() followed by LoadBalanceSession.isCanceled() but it is possible for the complete flag to change before the canceled flag (they are not updated atomically). - Fixed a typo in docs for MergeRecord.
- NIFI-9834 When calling ByteArrayContentRepository.read() on a null Content Claim, return an empty ByteArrayInputStream instead of throwing NullPointerException.
- NIFI-9832 Fixed disappearing XML element content when the element has attribute
- NIFI-9831 Fixed XML schema inference for XML elements with attributes
- NIFI-9827 Upgraded AWS Java SDK to 1.12.182 to pick up new AWS Regions
- NIFI-9826 Moved build locale settings from Maven to GitHub Workflow
- NIFI-9824 Addressed WARN log message in ProcessorAuditor on action RunOnce
- NIFI-9820 Reduced Kudu Client Worker Count default setting
- NIFI-9818 Fixed flaky tests
- NIFI-9816 Added lang attribute to html tag for WCAG 2.1 Accessibility
- NIFI-9815 Corrected log message formatting in multiple classes
- NIFI-9807 Added Refresh Window Property to OAuth2 Token Provider
- NIFI-9806 Introduced ConfigurableExtensionDefinition and VersionedConfigurableExtension
- NIFI-9801 Fixed error in previous correction of AccessToken.isExpired() margin calculation and stabilized shaky AccessTokenTest
- NIFI-9800 Unwrap SQLException in PutDatabaseRecord when table does not exist
- NIFI-9799 Enabled style checking for nifi-system-tests in ci-workflow

- NIFI-9797 Corrected AccessToken.isExpired() margin calculation
- NIFI-9796 Updated Registry Security Configuration to avoid warnings
- NIFI-9795 Checkstyle, rat issues in nifi-system-test-suite module
- NIFI-9794 If a node is OFFLOADING, do not allow connections to be deleted. Also, if we fail to notify the node that it needs to offload its data, change its state back to DISCONNECTED.
- NIFI-9793 Added wait for Controller Service run status in tests
- NIFI-9791 Use maven.build.timestamp during manifest generation instead of buildhelper timestamp
- NIFI-9790 Fixed race condition in which SwappablePriorityQueue could attempt to access the 0th element of an empty collection
- NIFI-9789 Upgraded Logback from 1.2.10 to 1.2.11
- NIFI-9788 Updated commons-codec to 1.15 across all modules
- NIFI-9787 Increased visibility of selected slickgrid row with 1px
- NIFI-9786 Added debug to KeyStoreUtils.isStoreValid
- NIFI-9785 Improved Login Credentials Writer File Handling
- NIFI-9783 When migrating FlowFiles from one ProcessSession to another, if any FlowFile had already been transferred, and the Relationship to which it was transferred was auto-terminated, fixed the wrong member variable being updated, which threw off stats for the processor.
- NIFI-9782 Excluded H2 DB from nifi-druid-bundle
- NIFI-9781 Fixed handling when selecting array element via QueryRecord
- NIFI-9778 Fixed additional details for ScriptedPartitionRecord
- NIFI-9777 Added support to remove attributes from verification requests
- NIFI-9776 Added the possibility to export flow definition with referenced services
- NIFI-9775 Created RuntimeManifestService
- NIFI-9774 Upgraded Netty from 4.1.73 to 4.1.74
- NIFI-9773 Refactored nifi-system-tests using JUnit 5
- NIFI-9771 When a Kafka record is obtained during config verification, an invalid response should be produced if the Record Reader is not able to produce any records from it.
- NIFI-9766 Avoid intermittent SearchElasticsearchTest failures in CI pipeline
- NIFI-9765 Added documentation that covers how to build a custom binary distribution
- NIFI-9764 Atlas reporting task sends 'unknown' hive_table when table is name not available
- NIFI-9763 Additional escaped VALUE column due to H2 changes
- NIFI-9763 Escaped VALUE column for Configure Details Auditing
- NIFI-9762 Added DBCPConnectionPool config verification
- NIFI-9761 Corrected PeerChannel processing for TLS 1.3
- NIFI-9759 Upgraded Spring Framework from 5.3.15 to 5.3.16
- NIFI-9757 Upgraded SLF4J from 1.7.35 to 1.7.36
- NIFI-9756 Added documentation for framework-level retry in Processors and update processor tab images
- NIFI-9754 Introduced VersionedExternalFlow - Updated stateless and StandardProcessGroup to make use of VersionedExternalFlow - Updated StatelessDataflowDefinition to use ExternalVersionedFlow instead of generic type - Updated Stateless Bootstrap to avoid loading stateless engine libs from root classpath but instead use a NarClassLoader to load the stateless nar
- NIFI-9751 Poll as needed during system-tests to ensure expected state
- NIFI-9750 Logging improvements to support LoadBalanceProtocol troubleshooting
- NIFI-9749 Captured additional logging for system-test workflow runs
- NIFI-9748 Added new property for Output Format to LogAttribute - Made the FlowFile Properties (file size, entry date, lineage start date) optional and renamed from 'Standard FlowFile Attributes' to 'FlowFile Properties'
- NIFI-9747 Track PID in nifi-bootstrap logging on shutdown
- NIFI-9745 Prevent insertion of revisions in NiFi registry when revision feature is disabled
- NIFI-9743 Upgraded Jetty from 9.4.44 to 9.4.45
- NIFI-9741 Made the close() method of WriteAvroResultWithExternalSchema idempotent
- NIFI-9738 VersionedComponent data members should derive from Object

- NIFI-9736 Improved TestRouteText to avoid intermittent failures
- NIFI-9735 Corrected Jetty Duplicate Mapping Warning
- NIFI-9734 Standardized exception cause message formatting
- NIFI-9733 Fixed StandardConnection.verifySourceStoppedOrFunnel infinite recursion
- NIFI-9732 Upgraded Zip4j from 2.8.0 to 2.9.1
- NIFI-9731 Updated to use a shorter, simpler output format for FlowFiles when creating bulletins
- NIFI-9730 Consider a change in value for retry-related fields from 'null' to the default value as an environmental change so that it is not flagged as a Local Modification, which would prevent users from updating the version of the Process Group that they are using
- NIFI-9729 When restarting components in the VersionedFlowSynchronizer, first filter out any components that are intended to be stopped.
- NIFI-9728 Added support for User Assigned Managed Identity authentication for Azure ADLS and Blob_v12 processors
- NIFI-9727 IndexOutOfBoundsException in CorrelationAttributePartitioner
- NIFI-9726 Removed duplicate nifi-utils dependencies from graph modules
- NIFI-9725 On shutdown, instead of spawning a background thread to shutdown Cluster Coordinator, do it in the calling thread. This avoids a race condition whereby the cluster coordinator cannot be determined because the other thread has shutdown the FlowController.
- NIFI-9724 Added set-sensitive-properties-algorithm command
- NIFI-9723 When controller-level Controller Services are added on restart of NiFi, ensure that all Controller Services are updated to include their property values. Also ensure that for these services and reporting tasks the property values are decrypted.
- NIFI-9722 Do not throw an Exception from verifyCanUpdateProperties when property descriptor and parameter descriptor's sensitivities do not match - instead allow the set to happen and let processor become invalid. Also, allow values such as abc#{param} for ghost processors.
- NIFI-9721 Support enum types in AvroTypeUtil.buildAvroSchema()
- NIFI-9720 Added QuerySalesforceObject Processor
- NIFI-9716 & NIFI-9577 Addressed issue in the PathFilter for GetFile/ListFile. For any file that is found in the Input Directory directly, it was previously being listed/fetched even if it did not match the PathFilter. Additionally, updated the code to create a new File Filter for every invocation of onTrigger. This was necessary for NIFI-9577 because the directory to monitor supports Expression Language and as a result may change from invocation to invocation, if using a function such as now() but the PathFilter would always relativize the path based on the value that was obtained when the processor was scheduled.
- NIFI-9715 Added option to output empty FlowFile from Elasticsearch REST API Json Query processors when there are no hits from query
- NIFI-9714 Added overloaded toMap to MapRecord that can convert sub-records into maps
- NIFI-9713 TagS3Object do not have provenance data
- NIFI-9711 Added support for flow.json.gz in SetSensitivePropertiesKey
- NIFI-9707 Resolved duplicate JLine dependency in nifi-toolkit
- NIFI-9704 Updated the ContentRepositoryScanTask to show details of how much content in the content repo is retained by each queue in the dataflow - Changed default for nifi.content.claim.max.appendable.size property from 1 MB to 50 KB - Updated documentation to reflect the new default value and explain what the property does and how it is used
- NIFI-9702 ListSFTP will set mime.type when Record Writer configured
- NIFI-9701 Corrected No Tracking strategy to create one flow file when using a Record Writer
- NIFI-9699 Updated oidcCallback method to handle error cases. Added some unit tests.
- NIFI-9698 When creating an Avro schema, ensure that any default value is converted from what is returned by RecordField.getDefaultValue() to what Avro requires.
- NIFI-9696 DeleteS3Object don't have provenance
- NIFI-9692 Upgraded Apache Commons Lang3 to 3.12.0
- NIFI-9691 Added ForkEnrichment, JoinEnrichment processors

- NIFI-9689 When checking FlowFile Availability, consider swap queue and trigger data to be swapped in, since calling poll() will no longer happen if no data is available.
- NIFI-9689 When all FlowFiles in a FlowFile Queue are penalized, do not schedule the destination to run. Also expose this fact through the ConnectionStatusSnapshotDTO, as this allows the front-end to render this information to the user in order to avoid confusion when it appears that the Processor has data but does nothing.
- NIFI-9688 Improved Logback shutdown handling
- NIFI-9687 Added additional documentation for nifi.cluster.node.protocol.max.threads property to Admin Guide
- NIFI-9686 Renamed SNMP integration tests correctly
- NIFI-9685 Upgraded JNA to 5.10.0
- NIFI-9684 When starting/stopping a selected process group, it sends the parent process group id to the REST interface that is responsible to enable/disable transmission for all remote process groups within a process group. Need to send the id of the select process group instead.
- NIFI-9681 Upgraded Apache Commons DBCP to 2.9.0
- NIFI-9679 Added access-environment-credentials permission
- NIFI-9678 Updated Elasticsearch REST API processor integration-tests for Elasticsearch 8.x
- NIFI-9673 Improved DBCP and HikariCP test reliability
- NIFI-9672 Fixed flaky tests caused by the use of HashMap
- NIFI-9669 Added PutDynamoDBRecord processor
- NIFI-9668 Added informative error when setting same value in 'set-param' CLI command
- NIFI-9663 Set the "csv-escape" property has no effect in SelectHive3QL
- NIFI-9662 Removed unused mail-1.4.7 dependency from nifi-framework-bundle
- NIFI-9660 Upgraded Apache Tika to 2.3.0
- NIFI-9657 Created MoveADLS processor
- NIFI-9655 Added Queue Logging to ListenUDP
- NIFI-9652 Upgraded jetty-schemas from 3.1 to 5.2
- NIFI-9650 Upgraded OkHttp from 4.9.2 to 4.9.3
- NIFI-9649 Upgraded SLF4J from 1.7.32 to 1.7.35
- NIFI-9647 Added ExtractDocumentText Processor
- NIFI-9645 Updated PutSplunk to allow idle connection timeouts
- NIFI-9644 Improved TestExecuteStateless increasing TestRunner.run() wait allowed
- NIFI-9642 Updated Admin Guide and User Guide with correct nifi.properties default values
- NIFI-9641 Adjusted the extraction of the chroot suffix for solr client connections.
- NIFI-9639: Determined how long it takes to find cluster coordinator and perform DNS lookup when sending heartbeats and include in the logs
- NIFI-9638 Refactored Google Guava usage in extensions
- NIFI-9638 Refactored Google Guava references
- NIFI-9635 Upgraded Netty from 4.1.72 to 4.1.73
- NIFI-9634 Upgraded Spring Framework to 5.3.15
- NIFI-9632 Removed nifi-lumberjack-bundle
- NIFI-9631 Enabled cli.sh to be used with a symbolic link
- NIFI-9630 Migrated Registry REST API docs to swagger-codegen
- NIFI-9629: Ensure that when we are setting default values on Avro GenericRecord objects that we convert from the schema's default value to the proper type.
- NIFI-9628 Added a uiOnly flag when requesting Controller Service
- NIFI-9626 Allowing Stateless NiFi to parse flow snapshots with unrecognized fields
- NIFI-9625 Refactored Distributed Cache Server and Client Tests - Replaced TestServerAndClient with separate classes for Set Server and Map Server - Implemented before and after annotations for starting and stopping server instances NIFI-9625 - Added check for cache directory existence before clean NIFI-9625 - Updated Map and Set Cache Server Tests to use random port
- NIFI-9624 Removed JCenter Repository
- NIFI-9621 Added Ignore Reserved Characters to FlattenJson

- NIFI-9620 Added isStateful to StatelessDataflow
- NIFI-9619 Removed GPG key from Security Mailing List reporting
- NIFI-9618 Upgraded Checkstyle to 9.2.1
- NIFI-9617 Removed unused screenshots from documentation
- NIFI-9616 Included SLF4J bridge libraries in NiFi Stateless Kafka Connect assembly
- NIFI-9611 Restored commons-io to minifi-assembly
- NIFI-9611 Removed unnecessary references to nifi-processor-utils
- NIFI-9611 Removed duplicate nifi-utils dependency
- NIFI-9610 Refactored nifi-processor-utils to separate modules
- NIFI-9609 Added nifi-snowflake-bundle with SnowflakeComputingConnectionPool
- NIFI-9608 Disabled system-tests workflow for pull requests
- NIFI-9607 Honor Update Keys when Quoting Identifiers in PutDatabaseRecord
- NIFI-9606 Removed nifi-security-utils from nifi-framework-api
- NIFI-9601 Upgraded nifi-bootstrap to Jakarta Mail 2
- NIFI-9600 Removed Elasticsearch 2 Processors
- NIFI-9599 Updated explicit plugin/build version references
- NIFI-9597 Fixed all explicit version refs to main latest
- NIFI-9597 Fixed Dockerfile URLs
- NIFI-9596 Fixed newline bug in JythonScriptRunner NIFI-9596 - Added comment to indicate why the Apache header is missing
- NIFI-9595 Removed nifi-kafka-0.x modules
- NIFI-9594 When converting Record to Avro GenericRecord, ensure that any default values that are defined in the GenericRecord's schema get applied, regardless of whether or not the field exists in the associated RecordSchema.
- NIFI-9593 Missing catch clauses in Confluent Schema Registry client
- NIFI-9591 Removed nifi-kite-bundle
- NIFI-9590 Added support for sensitive properties in Azure authorizers to encrypt-config
- NIFI-9589 Support initial loading from the current max values in QueryDatabaseTable* processors
- NIFI-9588 Updated documentation for `nifi.content.repository.archive.max.retention.period`
- NIFI-9587 Added JSON format for Prometheus Flow Metrics
- NIFI-9586 Removed Surefire ForkNodeFactory configuration
- NIFI-9586 Excluded additional assembly modules from ci-workflow
- NIFI-9585 Upgraded H2 from 1.4 to 2.1.210
- NIFI-9581 Added PutElasticsearchRecord relationship for output of successful Records sent to Elasticsearch
- NIFI-9580 UI work for framework-level retry in Processors
- NIFI-9578 NiFi user list no longer breaks if a user and group have the same id
- NIFI-9576 Introduced a BlockListClassLoader that can be used by stateless in order to isolate both the Stateless Engine and the NiFi extensions from extraneous classes that exist in the System ClassLoader
- NIFI-9576 Allowed Stateless' BlockListClassLoader to load java11 jars/classes
- NIFI-9575 Updated copyright year to 2022
- NIFI-9571 Corrected Session commit handling in PutTCP
- NIFI-9570 Separate classpath for NiFi Registry sensitive property providers
- NIFI-9569 SNMP manager UDP transportmapping changed to 0.0.0.0
- NIFI-9568 Updated nifi-jolt-transform-json-ui pom.xml to only include CSS and JS assets in WAR
- NIFI-9564 Removed unnecessary logback-classic test dependencies
- NIFI-9563 Enabled ListenTCP Pool Receive Buffers property
- NIFI-9552 Fixed NoClassDefFound error in case of nifi-registry-ranger-assembly
- NIFI-9548 When disabling RPG transmission, wait for the ports to complete in a background thread instead of blocking the web thread. - Moved the RPG initialization logic into flow controller instead of flow service and added a delay in order to reduce likelihood of ConnectException happening when pointing to nodes in the same cluster.
- NIFI-9545 Fixed in-place replacement for LookupRecord processor

- NIFI-9544 Fixed LookupRecord behavior when no matching value in the LRS
- NIFI-9543 Added bring-to-front functionality to labels
- NIFI-9527 Upgraded snappy-java to 1.1.8.4
- NIFI-9525 Modified lib packaging to use files from build directory
- NIFI-9501 Added REST end-point to retrieve a RuntimeManifest
- NIFI-9481 Excluded Data Transfer REST methods from DoSFilter
- NIFI-9475 Provide Framework-Level Retries for NiFi Relationships
- NIFI-9455 Added aggregated predictions to Prometheus Flow Metrics
- NIFI-9453 Refactored ListenBeats using Netty
- NIFI-9438 Refactored sensitive-property-provider to multiple modules
- NIFI-9435 Added registries and names include parameters to Flow Metrics
- NIFI-9425 Added auto-load NAR capability to MiNiFi
- NIFI-9400 Ensure that the CollectionUsage metrics in the mbeans are always used instead of the Usage metrics.
- NIFI-9390 Updated MergeContent / MergeRecord so that they play nicely within Stateless
- NIFI-9348 Added temporary suffix and fixed NIFI-7863 creation of the directories
- NIFI-9341 Corrected annotation syntax problem
- NIFI-9341 Added CEF RecordReader
- NIFI-9327 Added timewindow query to QueryNiFiReportingTask and MetricsEventReportingTask
- NIFI-9316 Registry Sort by label should be 'Last Updated (newest)' not 'Newest (update)'
- NIFI-9293 Ensure that we properly set the scheduled flag in the LifecycleState when stopping processors. - Added system test to verify that @OnScheduled, onTrigger, @OnUnscheduled, @OnStopped are all called and in the expected order.
- NIFI-9286 JOLT Expression Language fixes NIFI-6213 and adds in functionality to use expression language in class and module specification - Added JOLT unit tests - Addressed PR feedback - Fixed a problem with the scope of the EL for module directory - Alignment of JOLT processors NIFI-9286 fix checkstyle
- NIFI-9281 Enabled building on Java 17
- NIFI-9233 Improved reliability of system integration tests
- NIFI-9227 Run Once not working when scheduling strategy is CRON or Event driven
- NIFI-9166 Refactored nifi-standard-services to use JUnit 5
- NIFI-9134 Refactored nifi-metrics-reporting-bundle to use JUnit 5
- NIFI-9133 Refactored nifi-media-bundle to use JUnit 5
- NIFI-9124 Refactored nifi-jms-bundle to use JUnit 5
- NIFI-9113 Refactored nifi-grpc-bundle to use JUnit 5
- NIFI-9103 Refactored nifi-datadog-bundle to use JUnit 5
- NIFI-9085 Refactored the Elasticsearch bundle to use JUnit 5
- NIFI-9072 Improvements to ValidateXML
- NIFI-9065 Added support for OAuth2AccessTokenProvider in InvokeHTTP
- NIFI-9064 Support Oracle timestamp when `Use Avro Logical Types` is true for ExecuteSQLRecord and QueryDatabaseTableRecord
- NIFI-9058 Corrected AttributesToJSON Core Attributes filtering
- NIFI-8927 Added option to start/stop all controllers
- NIFI-8899 Added NiFi Registry version information to the Registry under an "about" button
- NIFI-8677 Added endpoint suffix for Azure EventHub Processors
- NIFI-8676 Added 'Tracking Entities' listing strategy to 'ListS3' and 'ListGCSBucket'
- NIFI-8549 Upgraded MiNiFi sensitive properties algorithm
- NIFI-8521 Removed nifi-influxdb-nar package from nifi-assembly
- NIFI-8492 Addressed issue in DatabaseReader class of IPLookupService that was attempting to set values on the JSON returned by MaxMind.
- NIFI-8209 Added Neo4J 4.X and 3.X clients by splitting the current controller service along release lines. This was necessary because Neo4J broke compatibility in their client drivers for Java between 3.X and 4.X at the Java

- API level. - Updated module name - Updated parent module - Renamed a few misnamed modules - Updated 1.15.0-SNAPSHOT references in cypher v3 package. - Updated neo4j 4.x driver
- NIFI-8124 Added Cookie Strategy property to InvokeHTTP
 - NIFI-8040 When changing version of a flow, stop processors that have a state of Starting in addition to those with a state of Running
 - NIFI-7865 amqp\$header is splitted in the wrong way for "," and "}"
 - NIFI-7840 Upgraded to Groovy 3.0.8 and Spock 2.1
 - NIFI-7835 Added authenticated SOCKS proxy support for SFTP
 - NIFI-7333 Added OIDC trust store strategy property
 - NIFI-7262 Added debug logging for the generated query in PutCassandraRecord
 - NIFI-7254 Upgraded geoip2 from 2.12.0 to 2.16.1
 - NIFI-7253 Upgraded Jackson to 2.13.2 using jackson-bom
 - NIFI-7192 Added systemd reload to nifi.sh install on systemd servers
 - NIFI-6871 Added HikariCPConnectionPool controller service
 - NIFI-6740 Added configuration options to specify NiFi/Bootstrap communication ports
 - NIFI-6699 Corrected SFTP symbolic link handling
 - NIFI-6390 & NIFI-1825 When writing to a FlowFile results in a 0-byte FlowFile, remove the content claim altogether. This is more efficient to process, but far more importantly it prevents a 0-byte FlowFile from holding content in the Content Repository. - Fixed issue in which a Provenance Event cannot be replayed if there is no ContentClaim.
 - NIFI-6266 Corrected proxy FTP connect handling
 - NIFI-6047 Cleaned up code to allow tests to run against 1.13.0-snapshot Removed DMC - Started integrating changes from NIFI-6014 - Added DMC tests - Added cache identifier recordpath test - Added additional details and removed old additional details - Removed EL from the dynamic properties - Removed a Java 11 - Renamed processor to DeduplicateRecord
 - NIFI-1468 Added tests to handle invalid cipher streams missing Salt/IV

Fixed Common Vulnerabilities and Exposures

Review the list of fixed common vulnerabilities and exposures.



Note: On June 28, 2022, Cloudera released a Service Pack for CFM 2.1.4 to fix some issues. If you previously upgraded to CFM 2.1.4.0, upgrade to CFM 2.1.4.1000-5 to access the latest version of the software. You can find the new download links in [Download locations](#).

CVE fixed in CFM 2.1.4.1000-5

CVE-2022-33140

The optional ShellUserGroupProvider in Apache NiFi 1.10.0 to 1.16.2 and Apache NiFi Registry 0.6.0 to 1.16.2 does not neutralize arguments for group resolution commands, allowing injection of operating system commands on Linux and macOS platforms. Command injection requires ShellUserGroupProvider to be one of the enabled User Group Providers in the Authorizers configuration. Command injection also requires an authenticated user with elevated privileges. Apache NiFi requires an authenticated user with authorization to modify access policies to execute the command. Apache NiFi Registry requires an authenticated user with authorization to read user groups to execute the command.

CVEs fixed in CFM 2.1.4

CVE-2021-42392

Apache NiFi uses H2 database for storing various NiFi runtime details. H2 database had a critical vulnerability similar to Log4Shell that potentially allows JNDI remote codebase loading. In NiFi, by default, console access to the database is restricted to local machine access only and remote access is disabled, which limits the severity of this vulnerability. More detailed information on the

H2 vulnerability can be found in [this blog post](#). Note that the fix for this CVE impacts the list of external databases Cloudera supports for the NiFi Registry instance. See the *Support Matrix* for more information.

CVE-2022-26850

When creating or updating credentials for single-user access, NiFi wrote a copy of the Login Identity Providers configuration to the operating system temporary directory. The Login Identity Providers configuration file contains the username and a bcrypt hash of the configured password. On most platforms, the operating system temporary directory has global read permissions. NiFi immediately moved the temporary file to the final configuration directory, which significantly limited the window of opportunity for access.

CVE-2020-36518

The vulnerable jackson-databind dependency allowed a Java stack overflow exception and denial of service through a large depth of nested objects.

CVE-2022-29265

Multiple components in Apache NiFi versions 0.0.1 to 1.16.0 do not restrict XML External Entity references in the default configuration. The Standard Content Viewer service attempts to resolve XML External Entity references when viewing formatted XML files. The following Processors attempt to resolve XML External Entity references when configured with default property values:

- EvaluateXPath
- EvaluateXQuery
- ValidateXml

Apache NiFi flow configurations that include these processors are vulnerable to malicious XML documents that contain Document Type Declarations with XML External Entity references.

Related Information

[Support matrix](#)