Informatica

Informatica® Mass Ingestion
January 2023

Connectors and Connections

Informatica Mass Ingestion Connectors and Connections
January 2023

Publication Date: 2023-01-16

Table of Contents

Preface

Use *Mass Ingestion Connectors and Connections* to determine the types of connectors that you need to download to be able to access sources and targets for each type of ingestion task. It also describes the properties that you configure when you define a connection to be used by an ingestion task. You download connectors and define connections in Administrator.

CHAPTER 1

Connectors and Connections

Connections provide access to data in cloud and on-premises applications, platforms, databases, and flat files. Before you can define a connection, ensure that the connector for the source or target type is installed in Informatica Intelligent Cloud Services.

If multiple connectors are available for a source or target type, get the one that your ingestion type supports. Some connectors are pre-installed. If you need a connector that is not pre-installed, you can download it from the **Add-On Connectors** page in Administrator.

Mass Ingestion connectors

You must have the correct connectors to create connections for the sources and targets you use in your ingestion tasks.

Before you can define connections, your organization administrator must ensure that the source and target connectors that the organization uses are installed. Also, you must enable connectors for your runtime environment.

For more information about connectors and connections, see "Licenses," "Runtime Environment," and "Connections" in the Administrator help.

Mass Ingestion Applications connectors

Before you define a connection for application ingestion tasks, verify that the connectors for your source and target types are available in Informatica Intelligent Cloud Services.

The following table lists the connectors that Mass Ingestion Applications requires to connect to a source or target:

Source or target type	Connector	Use for
Adobe Analytics	Adobe Analytics Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
Amazon Redshift	Amazon Redshift V2	Targets in initial load, incremental load, and combined initial and incremental load operations
Amazon S3	Amazon S3 V2	Targets in initial load, incremental load, and combined initial and incremental load operations
Apache Kafka	Kafka	Targets in incremental load operations

Source or target type	Connector	Use for
Databricks Delta	Databricks Delta	Targets in initial load, incremental load, and combined initial and incremental load operations
Google Analytics	Google Analytics Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
Google BigQuery	Google BigQuery V2	Targets in initial load, incremental load, and combined initial and incremental load operations
Google Cloud Storage	Google Cloud Storage V2	Targets in initial load, incremental load, and combined initial and incremental load operations
Marketo	Marketo V3	Sources in initial load, incremental load, and combined initial and incremental load operations
Microsoft Azure Data Lake Storage Gen2	Microsoft Azure Data Lake Storage Gen2	Targets in initial load, incremental load, and combined initial and incremental load operations
Microsoft Azure Synapse Analytics[1]	Microsoft Azure Synapse Analytics Database Ingestion	Targets in initial load, incremental load, and combined initial and incremental load operations
Microsoft Dynamics 365	Microsoft Dynamics 365 Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
NetSuite	NetSuite Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
Oracle Fusion Cloud Applications	Oracle Fusion Cloud Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
Salesforce	Salesforce Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
Salesforce Marketing Cloud	Salesforce Marketing Cloud	Sources in initial load operations
SAP ECC	- SAP Mass Ingestion - SAP ODP Extractor	Sources in initial load, incremental load, and combined initial and incremental load operations
SAP S/4HANA	SAP ODP Extractor	Sources in initial load, incremental load, and combined initial and incremental load operations
ServiceNow	ServiceNow Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations
Snowflake	Snowflake Data Cloud	Targets in initial load, incremental load, and combined initial and incremental load operations
Workday	Workday Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations

Source or target type	Connector	Use for
Zendesk	Zendesk Mass Ingestion	Sources in initial load, incremental load, and combined initial and incremental load operations

1. For Microsoft Azure Synapse Analytics targets, Mass Ingestion Applications uses Microsoft Azure SQL Data Lake Storage Gen2 to store staging files. Before you configure a connection for a Microsoft Azure Synapse Analytics target, ensure that you have Microsoft Azure SQL Data Lake Storage Gen2 installed.

Mass Ingestion Databases connectors

Before you begin defining connections for database ingestion tasks, verify that the connectors for your source and target types are available in Informatica Intelligent Cloud Services.

The following table lists the connectors that Mass Ingestion Databases requires to connect to a source or target that can be configured in a database ingestion task:

Source or target type	Connector	Use for
Amazon Redshift	Amazon Redshift V2	Targets in initial load, incremental load, and initial and incremental load jobs
Amazon S3	Amazon S3 V2	Targets in initial load and incremental load jobs
Databricks Delta	Databricks Delta	Targets in initial load, incremental load, and initial and incremental load jobs
Db2 for i	Db2 for i Database Ingestion	Sources in initial load, incremental load, and initial and incremental load jobs
Db2 for Linux, UNIX, and Windows	Db2 for LUW Database Ingestion	Sources in initial load jobs
Db2 for z/OS	Db2 for zOS Database Ingestion	Sources in initial load and incremental load jobs
Flat file	No connector required	Targets in initial load and incremental load jobs.
Google BigQuery	Google BigQuery V2	Targets in initial load, incremental load, and initial and incremental load jobs
Google Cloud Storage	Google Cloud Storage V2	Targets in initial load and incremental load jobs
Kafka, including Apache Kafka, Confluent Kafka, Amazon Managed Streaming for Apache Kafka, and Kafka-enabled Azure Event Hubs	Kafka	Targets in incremental load jobs
Microsoft Azure Data Lake Storage Gen2	Microsoft Azure Data Lake Storage Gen2	Targets in initial load and incremental load jobs
Microsoft Azure SQL Database	SQL Server	Sources in initial load jobs

Source or target type	Connector	Use for
Microsoft Azure Synapse Analytics[1]	Microsoft Azure Synapse Analytics Database Ingestion	Targets in initial load, incremental load, and initial and incremental load jobs
Microsoft SQL Server	SQL Server	Sources in initial load, incremental load, and initial and incremental load jobs Targets in initial load jobs
MongoDB	MongoDB Mass Ingestion	Sources in initial load and incremental load jobs
MySQL, including RDS for MySQL	MySQL	Sources in initial load and incremental load jobs. RDS for MySQL in initial load jobs only.
Netezza	Netezza	Sources in initial load jobs
Oracle, including RDS for Oracle	Oracle Database Ingestion	Sources in initial load, incremental load, and initial and incremental load jobs Targets in initial load, incremental load, and initial and incremental load jobs
PostgreSQL, including RDS for PostgreSQL and Amazon Aurora PostgreSQL	PostgreSQL	Sources in initial load and incremental load jobs
SAP HANA	SAP HANA Database Ingestion	Sources in initial load and incremental load jobs
Snowflake	Snowflake Data Cloud Connector	Targets in initial load, incremental load, and initial and incremental load jobs
Teradata Data Warehouse Appliance	Teradata	Sources in initial load jobs

1. For this target type, Mass Ingestion Databases uses Microsoft Azure SQL Data Lake Storage Gen2 to store staging files. Ensure that you have Microsoft Azure SQL Data Lake Storage Gen2 installed.

Mock connectors

Mass Ingestion Databases supports mock, or sample, connections for some of the sources and targets. Use mock connections to learn how to create initial load database ingestion tasks without creating real connections to the database.

A mock connector does not connect to a real database. Instead, a source mock connector uses flat files with sample data. A target mock connector reports the information about processed source data to Mass Ingestion Databases user interface, but it does not write any data to the target.

The sample connections appear in the source and target connection lists in Mass Ingestion Databases if you have the MockConnector license.

The following table lists mock connections that you can use for Mass Ingestion Databases sources and targets:

Connection name	Source or Target
Sample Oracle Connection	Source
Sample SQL Server Connection	Source
Sample S3 Connection	Target
Sample ADLS Gen2 Connection	Target

Note: You must use sample connections for both the source and target databases. You cannot use a sample connection for only one of them, for example, for the source but not for the target.

Source data

The source data for sample connections is stored in CVS files in the following directory:

```
Secure_Agent_installation/downloads/package-MockConnector.version/package/sampleData/source/database_type/
```

Each file represents a single table. A mock table name matches the file name. The first line in a file determines column headers, and the subsequent lines contain row data.

Mass Ingestion Files connectors

Before you define connections for file ingestion tasks, ensure that you have a license for the connectors that Mass Ingestion Files requires for the source and target types.

The following table lists the connectors that a file ingestion task supports based on the source and target types:

Source or target name	Connector	Source or target type
Local folder	No connector required	Source and Target
Advanced FTP	Advanced FTP V2 (add-on)	Source and Target
Advanced FTPS	Advanced FTPS V2 (add-on)	Source and Target
Advanced SFTP	Advanced SFTP V2 (add-on)	Source and Target
Amazon S3	Amazon S3 V2 (add-on)	Source and Target
Amazon Redshift	Amazon Redshift V2 (add-on)	target
Databricks Delta	Databricks Delta (add-on)	Source and Target
Google BigQuery	Google BigQuery V2 (add-on)	Target
Google Cloud Storage	Google Cloud Storage V2 (add-on)	Source and Target
Hadoop Files	Hadoop Files V2 (add-on)	Source and Target

Source or target name	Connector	Source or target type
Microsoft Azure Blob Storage	Microsoft Azure Blob Storage V3 (add-on)	Source and Target
Microsoft Azure Data Lake Store	Microsoft Azure Data Lake Store Gen2 (add-on)	Source and Target
Microsoft Azure Data Lake Store	Microsoft Azure Data Lake Store V3 (add-on)	Source and Target
Microsoft Azure Synapse SQL	Microsoft Azure Synapse SQL (add-on)	Target
Snowflake	Snowflake Cloud Data Warehouse V2 (add-on)	Target

Mass Ingestion Streaming connectors

Before you define connections for the streaming ingestion tasks, ensure that you have a license for the required connectors for your source and target types.

The following table lists the connectors that a streaming ingestion task supports based on the source and target type:

Source or target name	Connector	Source or target type
Amazon Kinesis Data Firehose	Amazon Kinesis (add-on)	Target
Amazon Kinesis Data Streams	Amazon Kinesis (add-on)	Source and Target
Amazon Managed Streaming for Apache Kafka (Amazon MSK)	Kafka (add-on)	Source and Target
Amazon S3	Amazon S3 V2 (add-on)	Target
AMQP	AMQP (add-on)	Source
Apache Kafka	Kafka (add-on)	Source and Target
Azure Event Hubs Kafka	Kafka (add-on)	Source
Confluent Kafka	Kafka (add-on)	Source and Target
Databricks Delta	Databricks Delta (add-on)	Target
Flat file	No connector required	Source and Target
Google BigQuery V2	Google BigQuery V2 (add-on)	Target
Google Cloud Storage	Google Cloud Storage V2 (add-on)	Target
Google PubSub	Google PubSub (add-on)	Source and Target

Source or target name	Connector	Source or target type
JDBC V2	JDBC V2 (add-on)	Target
JMS	JMS (add-on)	Source
Microsoft Azure Data Lake Storage	Azure Data Lake Store Gen2 (add-on)	Target
Microsoft Azure Event Hub	Azure Event Hubs (add-on)	Target
MQTT	MQTT (add-on)	Source
OPC UA	OPCUA (add-on)	Source
REST V2	REST V2 (add-on)	Source

Note: While importing a streaming ingestion task, both read and write connection types appear in the drop-down list on the **Import Review** page. You can also see connections to connectors that are not supported by Mass Ingestion Streaming.

Mass Ingestion connection properties

Connections provide access to data in cloud and on-premises applications, platforms, databases, and flat files. Connection definitions include the location of the source or target, the runtime environment, and the other properties specific to the connection type.

Before you can create a connection, ensure that the correct connectors for your sources and targets are available in Informatica Intelligent Cloud Services. The supported connectors vary by type of ingestion task.

To create a connection or search for an existing connection, use the Administrator service.

After you configure connection properties, the connection becomes available for use within the organization.

Configuring a connection

Configure a source or target connection on the **Connections** page in Administrator.

1. In Administrator, select **Connections**.
2. On the **Connections** page, click **New Connection**.

3. Configure the following connection details:

Property	Description
Connection Name	The name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Maximum length is 100 characters. Connection names are not case sensitive.
Description	A description of the connection.
Type	The type of connection, such as Amazon S3.

   After you select the connection type, additional properties that are specific to that type appear.

4. Configure the connection-specific properties.

   For example, if you are configuring an Amazon S3 connection, enter the Amazon S3 connection properties. Click the Help icon for a description of each connection property.

5. To test the connection, click **Test Connection**.

   The results of the test are displayed at the top of the page.

   If a connection fails, contact the database administrator, or recheck your settings and verify that the selected runtime environment has the status of Up and Running.

6. Click **Save** to save the connection.

Adobe Analytics Mass Ingestion connection properties

When you set up an Adobe Analytics Mass Ingestion connection, you must configure the connection properties.

Adobe Analytics uses a JSON Web Token (JWT) to authenticate the Adobe Analytics Mass Ingestion connection. To use an Adobe Analytics Mass Ingestion connection, you must create a Service Account Integration on Adobe Developer Console and then specify the service integration details in the connection properties. For more information about creating a Service Account Integration on Adobe Developer Console, see the Adobe documentation.

The following table describes the connection properties for an Adobe Analytics Mass Ingestion connection:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Client ID	Client ID of the Service Account that you created on Adobe Developer Console.
Client Secret	Client secret of the Service Account that you created on Adobe Developer Console.
Technical Account ID	Technical account ID of the Service Account.
Organization ID	Organization ID of the Service Account.

Connection property	Description
Private Key	Private key that is generated when you create the Service Account Integration. The private key is required to generate the JWT.
IMS Host	Base URL of Adobe Identity Management System (IMS). The default value is: `ims-na1.adobelogin.com`
IMS Exchange	Exchange URL of IMS. The connection use the JWT to obtain an access token from Adobe by making a POST request to the exchange URL. The default value is: `https://ims-na1.adobelogin.com/ims/exchange/jwt`

Advanced FTP V2 connection properties

When you set up an Advanced FTP V2 connection, you must configure the connection properties.

The following table describes the Advanced FTP V2 connection properties:

Connection property	Description
Connection Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. The description of the connection. The description cannot exceed 4,000 characters.
Type	Select the **Advanced FTP V2** connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. Specify a Secure Agent.
Host	The host name or IP address of the FTP server.
Port	The port number to use for connecting to the FTP server. If left blank, the default port number 21 is used.
Username	User name to connect to the FTP server.
Password	Password to connect to the FTP server.
Folder Path	The directory to use after connecting to the FTP server.

Connection property	Description
Use passive mode	Indicates whether the connection uses **Passive** or **Active** mode. Specify **Yes** to use **Passive** mode. Specify **No** to use **Active** mode. The default value is **Yes**. In Passive mode, the server does not need to connect back to a port on the connection client, which is a firewall-friendly mode. If you have problems with connecting to the server, you might want to change the mode to Passive by selecting **Yes** for this option. In Passive mode, depending on the FTP server, the connection may require high port range based on the port availability to transfer data. In Active mode, the server attempts to connect back to a port on the connection client to perform the data transfer.
Data Connection Start Port	The starting port number to use for the data connection.
Data Connection End Port	The ending port number to use for the data connection.
Timeout	The number of seconds to wait when attempting to connect to the server. A timeout occurs if the connection cannot be established in the specified amount of time. If left blank, the default value of 120 seconds is used.
Connection Retry Attempts	The number of times to connect to retry the FTP connection if a connection cannot be established. This setting is used for both the initial connection and any reconnect attempts due to lost connections. If left blank, no retries will be attempted.
Connection Retry Interval	The number of seconds to wait between each connection retry attempt. **Note:** For instance, if you want to retry to connect up to 10 times with a five second delay between retries, then specify **10** for the **Connection Retry Attempts** and **5** for the **Connection Retry** Interval.
Control Encoding	If left blank, the connection uses the ISO standard ISO-8859-1. If supported by the server, other encodings such as UTF-8 can be specified to support international characters.
List Parser	The list parser to use for the server connection. If the field is left blank, the Advanced FTP V2 Connector attempts to use the MLSD parser. If the MLSD parser is not supported by the server, the UNIX parser is used. If you experience problems listing directories, select a different list parser.
Date Format	This date format is applied if the server returns a date that is different from the selected list parser default. If your location requires a different date format (for example, d MMM yyyy), specify the date format in this field. Not all list parsers support the date format setting. List parsers that do not support the date format setting ignores any user specified value.
Recent Date Format	Specify the date format to use when parsing the recent last modified date for each file. The recent date format applies in UNIX-based systems and appears on entries less than a year old. If your location requires a specific date format (for example, d MMM HH:mm), specify that pattern in this field. Not all list parsers support the recent date format setting. List parsers that do not support the recent date format setting ignores any user-specified value.

Advanced FTPS V2 connection properties

When you set up an Advanced FTPS V2 connection, you must configure the connection properties.

The following table describes the Advanced FTPS V2 connection properties:

Connection property	Description
Connection Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. The description of the connection. The description cannot exceed 4,000 characters.
Type	Select the **Advanced FTPS V2** connection type.
Runtime Environment	The name of the runtime environment that contains the Secure Agent that you want to run the tasks.
Host	The host name or IP address of the server.
Port	The port number to use for connecting to the server. If left blank, the default port number is 21.
Username	User name to connect to the FTPS server.
Password	Password to connect to the FTPS server.
Folder Path	The directory to use after connecting to the server.
Use passive mode	Indicates whether the connection uses **Passive** or **Active** mode. Specify **Yes** to use **Passive** mode. Specify **No** to use **Active** mode. The default value is **Yes**. In Passive mode, the server does not need to connect back to a port on the connection client, which is a firewall-friendly mode. If you have problems with connecting to the server, you might want to change the mode to Passive by selecting **Yes** for this option. In Passive mode, depending on the FTPS server, the connection may require high port range based on the port availability to transfer data. In Active mode, the server attempts to connect back to a port on the connection client to perform the data transfer.
Data Connection Start Port	The starting port number to use for the data connection.
Data Connection End Port	The ending port number to use for the data connection.
Timeout	The number of seconds to wait when attempting to connect to the server. A timeout occurswill if the connection cannot be established in the specified amount of time. If left blank, the default value of 120 seconds is used.
Connection Retry Attempts	The number of times to connect to retry the Advanced FTP V2 connection if a connection cannot be established. This setting is used for both the initial connection and any reconnect attempts due to lost connections. If left blank, no retries will be attempted.

Connection property	Description
Connection Retry Interval	The number of seconds to wait between each connection retry attempt. **Note:** For instance, if you want to retry to connect up to 10 times with a five second delay between retries, then specify **10** for the **Connection Retry Attempts** and **5** for the **Connection Retry** Interval.
Control Encoding	If left blank, the connection uses the ISO standard ISO-8859-1. If supported by the server, other encodings like UTF-8 can be specified to support international characters.
Trusted Server	Specify whether the FTPS server is a trusted server. The Advanced FTP V2 Connector only supports a trusted server.
List Parser	The list parser to use for the server connection. If the field is empty, the Advanced FTP V2 Connector tries to use the MLSD parser. If the server does not support the MLSD parser, the connector uses the UNIX parser. If you experience problems listing directories, select a different list parser.
Date Format	This date format is applied if the server returns a date that is different from the selected list parser default. If your location requires a different date format (for example, d MMM yyyy), specify the date format in this field. Not all list parsers support the date format setting. List parsers that do not support the date format setting ignores any user specified values.
Recent Date Format	Specify the date format to use when parsing the recent last modified date for each file. The recent date format applies in UNIX-based systems and appears on entries less than a year old. If your location requires a specific date format (for example, d MMM HH:mm), specify that pattern in this field. Not all list parsers support the recent date format setting. List parsers that do not support the recent date format setting ignores any user-specified values.
Connection Type	Indicates if the connection type is IMPLICIT_SSL or EXPLICIT_SSL. - IMPLICIT_SSL. The connection automatically starts as an SSL connection. - EXPLICIT_SSL. After initial authentication with the FTPS server, the connection is encrypted with SSL or TLS depending on the security protocol you select. Default is IMPLICIT_SSL.
SecurityProtocol	Indicates whether SSL or TLS is used for EXPLICIT_SSL connections. Default is SSL.
Key Store File	The path and file name of the keystore file. The keystore file contains the certificates to authenticate the FTPS server.
Key Store Password	The password for the keystore file required to access the Trusted Server Certificate Store.
Key Alias	The alias of the individual key.
Key Store Type	Indicates if the type of the keystore is Java KeyStore (JKS) or Public Key Cryptology Standard (PKCS12). Default is JKS.

Advanced SFTP V2 connection properties

When you set up an Advanced SFTP V2 connection, you must configure the connection properties.

The following table describes the Advanced SFTP V2 connection properties:

Connection property	Description
Connection Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. The description of the connection. The description cannot exceed 4,000 characters.
Type	Select the **Advanced SFTP V2** connection type.
Runtime Environment	The name of the runtime environment that contains the Secure Agent that you want to run the tasks.
Host	The host name or IP address of the server.
Port	The port number to use for connecting to the server. If left blank, the default port number is 21.
Username	User name to connect to the SFTP server.
Password	Password to connect to the SFTP server.
Folder Path	The directory to use after connecting to the server.
Timeout	The number of seconds to wait when attempting to connect to the server. A timeout occurs if the connection cannot be established in the specified amount of time. If left blank, the default value of 120 seconds is used.
Connection Retry Attempts	The number of times to connect to retry the SFTP connection if a connection cannot be established. This setting is used for both the initial connection and any reconnect attempts due to lost connections. If left blank, no retries will be attempted.
Connection Retry Interval	The number of seconds to wait between each connection retry attempt. **Note:** For instance, if you want to retry to connect up to 10 times with a five second delay between retries, then specify **10** for the **Connection Retry Attempts** and **5** for the **Connection Retry** Interval.
Private Key File	The name of the SSH private key file along with the path where the file is stored. Ensure that the file path is on the machine that hosts the Secure Agent. For example, `C:/SSH/my_keys/key.ppk`
Private Key Passphrase	Specify the passphrase to encrypt the SSH private key.
Use Curve Kex Algorithm	Enable additional key exchange algorithms such as curve, and keyed-hash algorithm such as, -hmac-sha2-512, and -hmac-sha2-256.

Connection property	Description
Use File Integration Proxy Server	The connector connects to the SFTP server through the file integration proxy server. **Note:** - You must have the File Integration Service license to use this option. - You must define a proxy server in File Servers. - If you don't have the File Integration Service proxy, you need to use the agent proxy through the proxy.ini file.
Proxy Server Host Name	Host name or IP address of the outgoing File Integration Service proxy server.
Proxy Server Port	Port number of the outgoing File Integration Service proxy server.

Amazon Kinesis connection properties

The Amazon Kinesis connection is a messaging connection. Use the Amazon Kinesis connection to access Amazon Kinesis Data Streams or Amazon Kinesis Data Firehose as targets.

Amazon Kinesis Firehose connection properties

When you set up an Amazon Kinesis Firehose connection, you must configure the connection properties.

The following table describes the Amazon Kinesis Firehose connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. Description that you can use to identity the connection. The description cannot exceed 4,000 characters.
Type	The Amazon Kinesis connection type. If you do not see the Amazon Kinesis connection type, go to the **Add-On Connectors** page to enable the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Service	The type of Kinesis Service that you want to use. Select **Kinesis Firehose**.
AWS Access Key ID	The access key ID of the Amazon AWS user account.
AWS Secret Access Key	The secret access key for the Amazon AWS user account.

Property	Description
Region	Region where the endpoint for your service is available. You can select one of the following values: - us-east-2. Indicates the US East (Ohio) region. - us-east-1. Indicates the US East (N. Virginia) region. - us-west-1. Indicates the US West (N. California) region. - us-west-2. Indicates the US West (Oregon) region. - ap-northeast-1. Indicates the Asia Pacific (Tokyo) region. - ap-northeast-2. Indicates the Asia Pacific (Seoul) region. - ap-northeast-3. Indicates the Asia Pacific (Osaka-Local) region. - ap-south-1. Indicates the Asia Pacific (Mumbai) region. - ap-southeast-1. Indicates the Asia Pacific (Singapore) region. - ap-southeast-2. Indicates the Asia Pacific (Sydney) region. - ca-central-1. Indicates the Canada (Central) region. - cn-north-1. Indicates the China (Beijing) region. - cn-northwest-1. Indicates the China (Ningxia) region. - eu-central-1. Indicates the EU (Frankfurt) region. - eu-west-1. Indicates the EU (Ireland) region. - eu-west-2. Indicates the EU (London) region. - eu-west-3. Indicates the EU (Paris) region. - sa-east-1. Indicates the South America (São Paulo) region. - us-gov-west-1. Indicates AWS GovCloud (US-West) region. - us-gov-east-1. Indicates AWS GovCloud (US-East) region. A streaming ingestion task does not support ap-northeast-3 region.
Connection TimeOut (ms)	Optional. Number of milliseconds that the Mass Ingestion service waits to establish a connection to the Kinesis Firehose after which it times out. Default is 10,000 milliseconds.
AWS Credential Profile Name	An AWS credential profile defined in the credentials file. A mapping accesses the AWS credentials through the profile name at run time. If you do not provide an AWS credential profile name, the mapping uses the access key ID and secret access key that you specify when you create the connection.
ARN of IAM Role	The Amazon Resource Name specifying the role of an IAM user. Applies to Cross-Account IAM Roles authentication.
External ID	The external ID for an IAM role is an additional restriction that you can use in an IAM role trust policy to designate who can assume the IAM role. Applies to Cross-Account IAM Roles authentication.

Amazon Kinesis Streams connection properties

When you set up an Amazon Kinesis Streams connection, you must configure the connection properties.

The following table describes the Amazon Kinesis Streams connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. Description that you can use to identity the connection. The description cannot exceed 4,000 characters.
Type	The Amazon Kinesis connection type. If you do not see the Amazon Kinesis connection type, go to the **Add-On Connectors** page to install the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Service	The type of Kinesis Service that you want to use. Select **Kinesis Streams**.
AWS Access Key ID	The access key ID of the Amazon AWS user account.
AWS Secret Access Key	The secret access key for your Amazon AWS user account.
Region	Region where the endpoint for your service is available. You can select one of the following values: - us-east-2. Indicates the US East (Ohio) region. - us-east-1. Indicates the US East (N. Virginia) region. - us-west-1. Indicates the US West (N. California) region. - us-west-2. Indicates the US West (Oregon) region. - ap-northeast-1. Indicates the Asia Pacific (Tokyo) region. - ap-northeast-2. Indicates the Asia Pacific (Seoul) region. - ap-northeast-3. Indicates the Asia Pacific (Osaka-Local) region. - ap-south-1. Indicates the Asia Pacific (Mumbai) region. - ap-southeast-1. Indicates the Asia Pacific (Singapore) region. - ap-southeast-2. Indicates the Asia Pacific (Sydney) region. - ca-central-1. Indicates the Canada (Central) region. - cn-north-1. Indicates the China (Beijing) region. - cn-northwest-1. Indicates the China (Ningxia) region. - eu-central-1. Indicates the EU (Frankfurt) region. - eu-west-1. Indicates the EU (Ireland) region. - eu-west-2. Indicates the EU (London) region. - eu-west-3. Indicates the EU (Paris) region. - sa-east-1. Indicates the South America (São Paulo) region. - us-gov-west-1. Indicates AWS GovCloud (US-West) region. - us-gov-east-1. Indicates AWS GovCloud (US-East) region. A streaming ingestion task does not support ap-northeast-3 region.

Property	Description
Connection TimeOut (ms)	Optional. Number of milliseconds that the Mass Ingestion service waits to establish a connection to the Kinesis Streams after which it times out. Default is 10,000 milliseconds.
AWS Credential Profile Name	An AWS credential profile defined in the credentials file. A mapping accesses the AWS credentials through the profile name at run time. If you do not provide an AWS credential profile name, the mapping uses the access key ID and secret access key that you specify when you create the connection.
ARN of IAM Role	The Amazon Resource Name specifying the role of an IAM user. Applies to Cross-Account IAM Roles authentication.
External ID	The external ID for an IAM role is an additional restriction that you can use in an IAM role trust policy to designate who can assume the IAM role. Applies to Cross-Account IAM Roles authentication.

AWS Credential Profile

You can define AWS credential profiles in the credentials file. Each credential profile consists of secret access key and access key ID.

Users can use the AWS credential profile names to use different AWS credentials at run time than the AWS credentials that they specify when they create an Amazon Kinesis connection with an Amazon Kinesis Streams as a source and target and Amazon Kinesis Firehose as a target.

Create AWS credentials for the users, such as access key ID and secret access key. Users can select an authentication type while creating an Amazon Kinesis connection, such as AWS credential profile. The default authentication type is AWS credential profile.

Generate an Access Key ID and Secret Access Key for the users in AWS.

Amazon Redshift V2 connection properties

When you set up an Amazon Redshift V2 connection, configure the connection properties.

The following table describes the Amazon Redshift V2 connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Amazon Redshift V2 connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks. You cannot run an application ingestion task, database ingestion task, file ingestion task, or streaming ingestion task on a Hosted Agent or serverless runtime environment.

Property	Description
Username	User name of the Amazon Redshift account.
Password	Password for the Amazon Redshift account.
Access Key ID	Access key to access the Amazon S3 staging bucket. Enter the access key value based on the following authentication methods: - Basic authentication. Enter the actual access key value. - IAM authentication. Do not enter the access key value. - Temporary security credentials using assume role. Enter access key of an IAM user with no permissions to access the Amazon S3 staging bucket. - Assume role for EC2. Do not enter the access key value. **Note:** If you use the connection for application ingestion or database ingestion tasks that use key-based authentication, provide the access key value.
Secret Access Key	Secret access key to access the Amazon S3 staging bucket. The secret key is associated with the access key and uniquely identifies the account. Enter the access key value based on the following authentication methods: - Basic authentication. Enter the actual access secret value. - IAM authentication. Do not enter the access secret value. - Temporary security credentials using assume role. Enter access secret of an IAM user with no permissions to access Amazon S3 staging bucket. - Assume role for EC2. Do not enter the access secret value. **Note:** If you use the connection for application ingestion or database ingestion tasks that use key-based authentication, provide the access key value.
IAM Role ARN	The Amazon Resource Number (ARN) of the IAM role assumed by the user to use the dynamically generated temporary security credentials. Set the value of this property if you want to use the temporary security credentials to access the Amazon S3 staging bucket. For more information about how to get the ARN of the IAM role, see the AWS documentation. **Note:** If you use the connection for application ingestion or database ingestion tasks that uses role-based authentication, but not the default role for the AWS cluster, specify an IAM role ARN. If you use the default role, leave this field blank.
External Id	The external ID for a more secure access to the Amazon S3 bucket when the Amazon S3 staging bucket is in a different AWS account. Does not apply to application ingestion and database ingestion tasks.
Use EC2 Role to Assume Role	Optional. Select the check box to enable the EC2 role to assume another IAM role specified in the IAM Role ARN option. **Note:** The EC2 role must have a policy attached with a permission to assume an IAM role from the same or different account. Does not apply to application ingestion and database ingestion tasks. By default, this check box is not selected.
Master Symmetric Key	A 256-bit AES encryption key in the Base64 format when you enable client-side encryption. You can generate a key using a third-party tool. Does not apply to application ingestion and database ingestion tasks.
JDBC URL	The URL of the Amazon Redshift V2 connection. Enter the JDBC URL in the following format: `jdbc:redshift://<amazon_redshift_host>:<port_number>/<database_name>`

Property	Description
Cluster Region	The AWS cluster region in which the bucket you want to access resides. Select a cluster region if you choose to provide a custom JDBC URL that does not contain a cluster region name in the **JDBC URL** connection property. If you select a cluster region in both **Cluster Region** and **JDBC URL** connection properties, the agent ignores the cluster region that you specify in the **JDBC URL** connection property. To use the cluster region name that you specify in the **JDBC URL** connection property, select **None** as the cluster region in this property. You can only read data from or write data to the cluster regions supported by AWS SDK. Select one of the following cluster regions: - None - Asia Pacific(Mumbai) - Asia Pacific(Seoul) - Asia Pacific(Singapore) - Asia Pacific(Sydney) - Asia Pacific(Tokyo) - Asia Pacific(Hong Kong) - AWS GovCloud (US) - AWS GovCloud (US-East) - Canada(Central) - China(Bejing) - China(Ningxia) - EU(Ireland) - EU(Frankfurt) - EU(Paris) - EU(Stockholm) - South America(Sao Paulo) - Middle East(Bahrain) - US East(N. Virginia) - US East(Ohio) - US West(N. California) - US West(Oregon) Default is **None**.
Customer Master Key ID	The customer master key ID generated by AWS Key Management Service (AWS KMS) or the ARN of your custom key for cross-account access. You must generate the customer master key ID for the same region where your Amazon S3 staging bucket resides. You can either enter the customer-generated customer master key ID or the default customer master key ID. You can use a cross account KMS key for the same regions when you run mappings in advanced mode. Doesn't apply to application ingestion and database ingestion tasks.

Amazon S3 V2 connection properties

When you set up an Amazon S3 V2 connection, configure the connection properties.

The following table describes the Amazon S3 V2 connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Amazon S3 V2 connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks. You cannot run an application ingestion task or a database ingestion task on a Hosted Agent or serverless runtime environment.
Access Key	Access key to access the Amazon S3 bucket. Enter the access key value based on the following authentication methods: - Basic authentication. Enter the actual access key value. - IAM authentication. Don't enter the access key value. - Temporary security credentials using assume role. Enter the secret access key of an IAM user with no permissions to access Amazon S3 bucket. - Assume role for EC2. Don't enter the access key value. - Credential profile file authentication. Don't enter the access key value. - Federated user single sign-on. Don't enter the secret access key value.
Secret Key	Secret access key to access the Amazon S3 bucket. The secret key is associated with the access key and uniquely identifies the account. Enter the secret access key value based on the following authentication methods: - Basic authentication. Enter the actual access secret value. - IAM authentication. Don't enter the access secret value. - Temporary security credentials using assume role. Enter access secret of an IAM user with no permissions to access Amazon S3 bucket. - Assume role for EC2. Don't enter the access key value. - Credential profile file authentication. Don't enter the access secret value. - Federated user single sign-on. Don't enter the access secret value.
IAM Role ARN	The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role assumed by the user to use the dynamically generated temporary security credentials. Enter the value of this property if you want to use the temporary security credentials to access the AWS resources. This property is not applicable to an application ingestion task. **Note:** Even if you remove the IAM role that enables the agent to access the Amazon S3 bucket and create a connection, the test connection is successful. For more information about how to get the ARN of the IAM role, see the AWS documentation.
External Id	Provides a more secure access to the Amazon S3 bucket when the Amazon S3 bucket is in a different AWS account.

Property	Description
Use EC2 Role to Assume Role	Enables the EC2 role to assume another IAM role specified in the IAM Role ARN option. **Note:** The EC2 role must have a policy attached with a permission to assume an IAM role from the same or different account. By default, the Use EC2 Role to Assume Role check box is not selected. **Note:** Enter a value for the IAM Role ARN property when you enable this property for a streaming ingestion task.
Folder Path	Bucket name or complete folder path to the Amazon S3 objects. Don't use a slash at the end of the folder path. For example, `<bucket name>/<my folder name>`.
Master Symmetric Key	A 256-bit AES encryption key in the Base64 format when you use client-side encryption. You can generate a key using a third-party tool. Not applicable for an application ingestion task, database ingestion task, or streaming ingestion task.
Customer Master Key ID	The customer master key ID or alias name generated by AWS Key Management Service (AWS KMS) or the Amazon Resource Name (ARN) of your custom key for cross-account access. You must generate the customer master key for the same region where the Amazon S3 bucket resides. You can specify the following master keys: - Customer generated customer master key. Enables client-side or server-side encryption. - Default customer master key. Enables client-side or server-side encryption. Only the administrator user of the account can use the default customer master key ID to enable client-side encryption. Not applicable for an application ingestion task, database ingestion task, or streaming ingestion task.
S3 Account Type	The type of the Amazon S3 account. Select from the following options: - Amazon S3 Storage. Enables you to use the Amazon S3 services. - S3 Compatible Storage. Enables you to use the endpoint for a third-party storage provider such as Scality RING or MinIO. Default is Amazon S3 storage.
REST Endpoint	The S3 storage endpoint required for S3 compatible storage. Enter the S3 storage endpoint in HTTP or HTTPs format. For example, http://s3.isv.scality.com.

Property	Description
Region Name	The AWS region of the bucket that you want to access. Select one of the following regions: - Asia Pacific(Mumbai) - Asis Pacific(Jakarta) - Asia Pacific (Osaka) - Asia Pacific(Seoul) - Asia Pacific(Singapore) - Asia Pacific(Sydney) - Asia Pacific(Tokyo) - Asia Pacific(Hong Kong) - AWS GovCloud (US) - AWS GovCloud (US-East) - Canada(Central) - China(Bejing) - China(Ningxia) - EU(Ireland) - EU(Frankfurt) - EU (London) - EU (Milan) - EU(Paris) - EU(Stockholm) - South America(Sao Paulo) - Middle East(Bahrain) - US East(N. Virginia) - US East(Ohio) - US ISO East - US ISOB East (Ohio) - US ISO West - US West(N. California) - US West(Oregon) Default is US East (N. Virginia).
Federated SSO IdP	SAML 2.0-enabled identity provider for the federated user single sign-on to use with the AWS account. Amazon S3 V2 connector supports only the ADFS 3.0 identity provider. Select **None** if you don't want to use federated user single sign-on. **Note:** Federated user single sign-on is not applicable to application ingestion tasks, database ingestion tasks, and streaming ingestion tasks.
Other Authentication Type	Select one the following authentication types: - NONE - Credential Profile File Authentication Select the Credential Profile File Authentication option to access the Amazon S3 credentials from a credential file that contains the access key and secret key. Enter the credential profile file path and the profile name to establish the connection with Amazon S3. You can use permanent IAM credentials or temporary session tokens when you configure the Credential Profile File Authentication. Default is NONE.

Property	Description
Credential Profile File Path	Specifies the credential profile file path. If you don't enter the credential profile path, the Secure Agent uses the credential profile file present in the following default location in your home directory: `~/.aws/credentials` **Note:** Mass Ingestion Databases has not been certified with the **Credential Profile File Path** and **Profile Name** connection properties. Mass Ingestion Databases finds AWS credentials by using the default credential provider chain that is implemented by the DefaultAWSCredentialsProviderChain class, which includes the credential profile file.
Profile Name	Name of the profile in the credential profile file used to get the credentials. If you don't enter the profile name, the credentials from the default profile in the credential profile file are used.
S3 VPC Endpoint Type	The VPC endpoint type for Amazon S3. You can enable private communication with Amazon S3 by selecting a VPC endpoint. Select one of the following VPC endpoint types: - None - Gateway Endpoint - Interface Endpoint Default is None. This property is not applicable to an application ingestion task and database ingestion task.
Endpoint DNS Name for Amazon S3	The DNS name for the Amazon S3 interface endpoint. Enter the DNS name in the following format: `bucket.<DNS name of the interface endpoint>` This property is not applicable to an application ingestion task and database ingestion task.
STS VPC Endpoint Type	Applicable when you select the S3 VPC interface endpoint. The VPC endpoint type for AWS STS. When you select **IAM Role ARN** or **Federated SSO IdP**, configure the STS VPC endpoint. This property is not applicable to an application ingestion task and database ingestion task.
Endpoint DNS Name for AWS STS service	The DNS name for the AWS STS interface endpoint. This property is not applicable to an application ingestion task and database ingestion task.
KMS VPC Endpoint Type	Applicable when you select the interface endpoint. The VPC endpoint type for the AWS KMS. When you select **Customer Master Key ID**, configure the KMS VPC endpoint. This property is not applicable to an application ingestion task and database ingestion task.
Endpoint DNS Name for AWS KMS service	The DNS name for the AWS KMS interface endpoint. This property is not applicable to an application ingestion task and database ingestion task.

Federated user single sign-on connection properties

Configure the following properties when you select ADFS 3.0 in Federated SSO IdP:

Property	Description
Federated User Name	User name of the federated user to access the AWS account through the identity provider.
Federated User Password	Password for the federated user to access the AWS account through the identity provider.
IdP SSO URL	Single sign-on URL of the identity provider for AWS. Not applicable for a streaming ingestion task.
SAML Identity Provider ARN	ARN of the SAML identity provider that the AWS administrator created to register the identity provider as a trusted provider.
Role ARN	ARN of the IAM role assumed by the federated user.

Credential Profile File Authentication

You can provide the credentials required to establish the connection with Amazon S3 through the credential profile file that contains an access key and secret key. The credential profile file contains an access key, a secret key, and a session token when you use temporary security credentials.

You can use permanent IAM credentials or temporary security credentials with a session token when you use credential profile file authentication.

If you do not specify the credential profile file path, the default credential file path is used. If you do not specify the profile name, the credentials are used from the default profile in the credential file.

Consider the following rules for a credential profile file:

- The credential file must be on the same machine where you installed the Secure Agent.
- The credential profile file name must end with `.credentials`.
- If you do not specify the credential profile path, the Secure Agent uses the credential profile file present in the following default location in your home directory:
  `~/.aws/credentials`

  Note: On Windows, you can refer to your home directory by using the environment variable `%UserProfile%`. On Unix-like systems, you can use the environment variable `$HOME`.

A sample credential profile file:

```
[default]
aws_access_key_id = 1233333
aws_secret_access_key = abcabcabc

[test-profile]
aws_access_key_id = 1233333
aws_secret_access_key = abcabcabc
aws_session_token = jahaheieomdrtflmlioerp
```

The `aws_access_key_id` and `aws_secret_access_key` specify the AWS access key and secret key used as part of credentials to authenticate the user.

The `aws_session_token` specifies an AWS session token used as part of the credentials to authenticate the user. A session token is required only if you specify temporary security credentials.

AMQP connection properties

When you set up an AMQP connection, you must configure the connection properties.

The following table describes the AMQP connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
Description	Optional. Description that you can use to identity the connection. The description cannot exceed 4,000 characters.
Type	The AMQP connection type. If you do not see the connection type, go to the **Add-On Connectors** page to install the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Host Name	Network address of the AMQP broker.
Port	Port number of the AMQP broker to which the underlying TCP connection is made. Default is 5672.
Virtual Host	Virtual host name that identifies the AMQP system. Use the virtual host name for enhanced security.
Username	Username for the AMQP broker.
Password	Password for the AMQP broker.
Use SSL	Enable this option to use SSL for secure transmission. If you enable the SSL authentication, ensure that you provide both keystore and truststore details for using the AMQP connection in a streaming ingestion task.
Keystore File Name	Contains the keys and certificates required for secure communication.
Keystore Password	Password for the keystore filename.
Keystore Type	Type of keystore that you want to use. Keystore type defines the storage and data format of the keystore information and the algorithms used to protect private keys in the keystore. Use one of the following types: - JKS. Stores private keys and certificates. - PKCS12. Stores private keys, secret keys, and certificates.

Property	Description
Truststore File Name	Name of the truststore file.
Truststore Password	Password for the truststore file.
Truststore Type	Type of truststore that you want to use. Use one of the following types: - JKS - PKCS 12
TLS Protocol	Transport protocols that you want to use. Use one of the following types: - SSL - SSLv2Hello - SSLv3 - TLS - TLSv1 - TLSv1.1 - TLSv1.2
Client Authentication	Client authentication policy when connecting to the secured AMQP broker. Use one of the following property values when you define and enable an SSL context. - WANT - REQUIRED - NONE

Db2 for i Database Ingestion connection properties

When you define a Db2 for i Database Ingestion connection, you must configure connection properties. You can use this connection type in database ingestion tasks, which you configure in the Mass Ingestion service.

The following table describes the connection properties:

Property	Description
Connection Name	A name for the connection. This name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Spaces at the beginning or end of the name are trimmed and are not saved as part of the name. Maximum length is 100 characters. Connection names are not case sensitive.
Description	An optional description for the connection. Maximum length is 255 characters.
Type	The type of connection. Ensure that the type is Db2 for i Database Ingestion.
Runtime Environment	The name of the runtime environment where you want to run database ingestion tasks. You define runtime environments in Administrator.
User Name	The user name to use for connecting to the Db2 for i instance.
Password	The password to use for connecting to the Db2 for i instance.
Host	The name of the machine that hosts the database server.

Property	Description
Port	The network port number used to connect to the database server.
Location Name	The name of the Db2 for i location that you want to access. Your system administrator can determine the name of the Db2 location by using the WRKRDBDIRE command. In the output, find the name of the database that is listed as *LOCAL and then use that value as the value of this property.
JDBC Driver	The type of JDBC driver. Select one of the following options: - Data Direct - JTOpen Default is Data Direct.
Code Page for Bit Data	The code page that Mass Ingestion Databases uses to read character data that is stored as bit data. This value must be a canonical name for the java.io API and java.lang API. For more information, see the supported encodings in the Oracle Java documentation. Specify this property if you have FOR BIT DATA source columns.
Advanced Connection Properties	Advanced properties for the JDBC driver which is used to connect to the Db2 for i source. If you specify more than one *property=value* entry, separate them with a semicolon (;). For information about the DataDirect JDBC driver connection properties, see Progress DataDirect documentation. For example, you can set the ConnectionRetryCount property to control the number of times the driver retries attempts to connect to the primary database server. For information about the JTOpen JDBC driver connection properties, see IBM Toolbox for Java JDBC properties.
Encryption Method	The data encryption method for the JTOpen JDBC Driver. Select one of the following options: - No Encryption - SSL Default is No Encryption. If you select SSL, you must add the required certificates to the Informatica Cloud Secure Agent JRE cacerts keystore in one of the following locations: For Linux: *Secure Agent Directory*\jdk\jre\lib\security\cacerts For Windows: *Secure Agent Directory*\apps\jdk*LatestVersion*\jre

Db2 for LUW Database Ingestion connection properties

When you define a Db2 for LUW Database Ingestion connection, you must configure connection properties. You can use this connection type in database ingestion tasks, which you configure in the Mass Ingestion service.

The following table describes the connection properties:

Property	Description
Connection Name	A name for the connection. This name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Spaces at the beginning or end of the name are trimmed and are not saved as part of the name. Maximum length is 100 characters. Connection names are not case sensitive.
Description	An optional description for the connection. Maximum length is 255 characters.
Type	The type of connection. Ensure that the type is Db2 for LUW Database Ingestion.
Runtime Environment	The name of the runtime environment where you want to run database ingestion tasks. You define runtime environments in Administrator.
User Name	The user name to use for connecting to the Db2 for LUW instance.
Password	The password to use for connecting to the Db2 for LUW instance.
Host	The name of the machine that hosts the database server.
Port	The network port number used to connect to the database server.
Database Name	The name of the Db2 for LUW database that you want to access.
Advanced Connection Properties	Advanced properties for the Progress DataDirect JDBC DB2 driver, which is used to connect to the Db2 for LUW source. If you specify more than one *property=value* entry, separate them with a semicolon (;). The driver properties that you can enter in this field are described in the Progress DataDirect documentation at https://docs.progress.com/bundle/datadirect-connect-jdbc-51/page/Connection-Properties_10.html. For example, you can set the EncryptionMethod property to control whether data is encrypted and decrypted when transmitted over the network between the driver and database server.

Db2 for zOS Database Ingestion connection properties

When you define a Db2 for zOS Database Ingestion connection, you must configure connection properties. You can use this connection type in database ingestion tasks, which you configure in the Mass Ingestion service.

The following table describes the connection properties:

Property	Description
Connection Name	A name for the connection. This name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Spaces at the beginning or end of the name are trimmed and are not saved as part of the name. Maximum length is 100 characters. Connection names are not case sensitive.
Description	An optional description for the connection. Maximum length is 255 characters.
Type	The type of connection. Ensure that the type is Db2 for zOS Database Ingestion.
Runtime Environment	The name of the runtime environment where you want to run database ingestion tasks. You define runtime environments in Administrator.
User Name	The user name to use for connecting to the Db2 for zOS instance.
Password	The password to use for connecting to the Db2 for zOS instance.
Host	The name of the machine that hosts the database server.
Port	The network port number used to connect to the database server.
Location Name	The name of the Db2 for zOS location that you want to access. For DB2 for z/OS, your system administrator can determine the name of your DB2 location using the command DISPLAY DDF.
Code Page for Bit Data	The code page that Mass Ingestion Databases uses to read character data that is stored as bit data. This value must be a canonical name for the java.io API and java.lang API. For more information, see the supported encodings in the Oracle Java documentation. Specify this property if you have FOR BIT DATA source columns.
CDC Stored Procedure Schema	For incremental change data capture processing, the name of the schema for the z/OS stored procedure that is required to collect change data from the Db2 log. This value is specified in the #STPINST data set that you customized when setting up the stored procedure on z/OS. No default value is provided.
CDC Stored Procedure Name	For incremental change data capture processing, the name of the z/OS stored procedure that is required to collect change data from the Db2 log. This value is specified in the #STPINST data set that you customized when setting up the stored procedure on z/OS. The default value is INFALOG.
Advanced Connection Properties	Advanced properties for the Progress DataDirect JDBC DB2 driver, which is used to connect to the Db2 for z/OS source. If you specify more than one *property=value* entry, separate them with a semicolon (;). The driver properties that you can enter in this field are described in the Progress DataDirect documentation at https://docs.progress.com/bundle/datadirect-connect-jdbc-51/page/Connection-Properties_10.html. For example, you can set the ConnectionRetryCount property to control the number of times the driver retries attempts to connect to the primary database server.

Databricks Delta connection properties

When you set up a Databricks Delta connection, configure the connection properties.

The following table describes the Databricks Delta connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Databricks Delta connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks. You cannot run an application ingestion, database ingestion, or streaming ingestion task on a Hosted Agent or serverless runtime environment.
Databricks Host	The host name of the endpoint the Databricks account belongs to. Use the following syntax: `jdbc:spark://<Databricks Host>:443/default;transportMode=http;ssl=1;httpPath=sql/protocolv1/o/<Org Id>/<Cluster ID>;AuthMech=3;UID=token;PWD=<personal-access-token>` **Note:** You can get the URL from the Advanced Options of JDBC or ODBC in the Databricks Delta analytics cluster or all purpose cluster. The value of PWD in Databricks Host, Org Id, and Cluster ID is always `<personal-access-token>`.
Cluster ID	The ID of the Databricks analytics cluster. You can get the cluster ID from the JDBC URL. Use the following syntax: `jdbc:spark://<Databricks Host>:443/default;transportMode=http;ssl=1;httpPath=sql/protocolv1/o/<Org Id>/<Cluster ID>;AuthMech=3;UID=token;PWD=<personal-access-token>`
Organization Id	The unique organization ID for the workspace in Databricks. Use the following syntax: `jdbc:spark://<Databricks Host>:443/default;transportMode=http;ssl=1;httpPath=sql/protocolv1/o/<Org Id>/<Cluster ID>;AuthMech=3;UID=token;PWD=<personal-access-token>`
Databricks Token	Personal access token to access Databricks. Ensure that you have permissions to attach to the cluster identified in the **Cluster ID** property.

Property	Description

SQL Endpoint JDBC URL	Databricks SQL endpoint JDBC connection URL.
	Use the following syntax:
	`jdbc:spark://<Databricks Host>:443/default;transportMode=http;ssl=1;AuthMech=3;httpPath=/sql/1.0/endpoints/<SQL endpoint cluster ID>;`
	For application ingestion and database ingestion tasks, begin the URL with the prefix jdbc:databricks://, as follows:
	`jdbc:databricks://<Databricks Host>:443/default;transportMode=http;ssl=1;AuthMech=3;httpPath=/sql/1.0/endpoints/<SQL endpoint cluster ID>;`
	This field is required to connect to the Databricks SQL endpoint.
	Note: The Databricks Host, Organization ID, and Cluster ID properties are not considered if you configure the SQL Endpoint JDBC URL property.

Database	The database in Databricks Delta that you want to connect to.
JDBC Driver Class Name	The name of the JDBC driver class. For application ingestion and database ingestion tasks, specify the driver class name as: `com.databricks.client.jdbc.Driver`
Cluster Environment	The cloud provider where the Databricks cluster is deployed. Choose from the following options: - AWS - Azure Default is AWS. The connection attributes depend on the cluster environment you select. For more information, see the AWS cluster properties and Azure cluster properties sections.
Min Workers	The minimum number of worker nodes to be used for the Spark job.
Max Workers	The maximum number of worker nodes to be used for the Spark job. If you don't want to autoscale, set Max Workers = Min Workers or don't set Max Workers.
DB Runtime Version	The Databricks runtime version. Select 7.3 LTS from the list.
Worker Node Type	The worker node instance type that is used to run the Spark job. For example, the worker node type for AWS can be i3.2xlarge. The worker node type for Azure can be Standard_DS3_v2.
Driver Node Type	The driver node instance type that is used to collect data from the Spark workers. For example, the driver node type for AWS can be i3.2xlarge. The driver node type for Azure can be Standard_DS3_v2. If you don't specify the driver node type, Databricks uses the value you specify in the worker node type field.
Instance Pool ID	The instance pool ID used for the Spark cluster.
Enable Elastic Disk	Enables the cluster to get additional disk space. Enable this option if the Spark workers are running low on disk space.

Property	Description

Spark Configuration	The Spark configuration to use in the Databricks cluster.
	The configuration must be in the following format:
	`"key1"="value1";"key2"="value2";....`
	For example:
	`"spark.executor.userClassPathFirst"="False"`

Spark Environment Variables	The environment variables to export before launching the Spark driver and workers.
	The variables must be in the following format:
	`"key1"="value1";"key2"="value2";....`
	For example:
	`"MY_ENVIRONMENT_VARIABLE"="true"`

Flat file connection properties

The following table describes the flat file connection properties:

Connection Property	Description
Runtime Environment	Runtime environment that contains the Secure Agent to use to access the flat files. **Note:** Do not select a runtime environment with Secure Agents that run on NTT. A flat file connection cannot use a Secure Agent that runs on NTT.
Directory	Directory where the flat file is stored. Must be accessible by all Secure Agents in the selected runtime environment. Enter the full directory or click **Browse** to locate and select the directory. When you use the connection, you can select a file that's contained in the directory or in any of its subdirectories. Maximum length is 100 characters. Directory names can contain alphanumeric characters, spaces, and the following special characters: `/ \ : _ ~` The directory is the service URL for this connection type. **Note:** On Windows, the **Browse for Directory** dialog box does not display mapped drives. You can browse My Network Places to locate the directory or enter the directory name in the following format: `\\<server_name>\<directory_path>`. If network directories do not display, you can configure a login for the Secure Agent service. Do not include the name of the flat file. You specify the file name when you create the task.
Browse button	Use to locate and select the directory where flat files are stored.

Connection Property	Description
Date Format	Date format for date fields in the flat file. Default date format is: MM/dd/yyyy HH:mm:ss
Code Page	The code page of the system that hosts the flat file. Select one of the following code pages: - MS Windows Latin 1. Select for ISO 8859-1 Western European data. - UTF-8. Select for Unicode data. - UTF-16 encoding of Unicode (Big Endian). - UTF-16 encoding of Unicode (Lower Endian). - Shift-JIS. Select for double-byte character data. - ISO 8859-15 Latin 9 (Western European). - ISO 8859-2 Eastern European. - ISO 8859-3 Southeast European. - ISO 8859-5 Cyrillic. - ISO 8859-9 Latin 5 (Turkish). - IBM EBCDIC International Latin-1. - Japanese EUC (with \ <-> Yen mapping - IBM EBCDIC Japanese - IBM EBCDIC Japanese CP939 - PC Japanese SJIS-78 syntax (IBM-942) - PC Japanese SJIS-90 (IBM-943) - MS Windows Traditional Chinese, superset of Big 5 - Taiwan Big-5 (w/o euro update) - Chinese EUC - ISO 8859-8 Hebrew - PC Hebrew (old) - PC Hebrew (w/o euro update) - EBCDIC Hebrew (updated with new sheqel, control characters) - IBM EBCDIC US English IBM037 In elastic mappings, flat file objects in cloud storage connections must use UTF-8 encoding. If the file contains supplementary characters with UTF-16 encoding, the task fails. **Note:** When you use a flat file connection with the Shift-JIS code page and a UTF data object, be sure to install fonts that fully support Unicode.

Google Analytics Mass Ingestion connection properties

When you set up a Google Analytics Mass Ingestion connection, you must configure the connection properties.

The following table describes the connection properties for a Google Analytics Mass Ingestion connection:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Service Account ID	Specifies the client_email value present in the JSON file that you download after you create a service account.
Service Account Key	Specifies the private_key value present in the JSON file that you download after you create a service account.

Google BigQuery V2 connection properties

When you create a Google BigQuery V2 connection, configure the connection properties.

The following table describes the Google BigQuery V2 connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Google BigQuery V2 connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks. You cannot run a database ingestion task on a Hosted Agent or serverless runtime environment.
Service Account ID	The client_email value in the JSON file that you download after you create a service account.
Service Account Key	The private_key value in the JSON file that you download after you create a service account.
Project ID	The project_id value in the JSON file that you download after you create a service account. If you have created multiple projects with the same service account, enter the ID of the project that contains the dataset that you want to connect to.
Storage Path	Path in Google Cloud Storage where the agent creates a local stage file to store the data temporarily. Applies to tasks that read or write large volumes of data. Use this property when you read data in staging mode or write data in bulk mode. You can either enter the bucket name or the bucket name and folder name. Use one of the following formats: - `gs://<bucket name>` - `gs://<bucket name>/<folder_name>`
Connection mode	The mode that you want to use to read data from or write data to Google BigQuery. Select one of the following connection modes: - Simple. Flattens each field within the Record data type field as a separate field in the mapping. - Hybrid[1]. Displays all the top-level fields in the Google BigQuery table including Record data type fields. Google BigQuery V2 Connector displays the top-level Record data type field as a single field of the String data type in the mapping. - "Complex[1]. Displays all the columns in the Google BigQuery table as a single field of the String data type in the mapping. Default is Simple.

Property	Description
Schema Definition File Path[1]	Directory on the Secure Agent machine where the Secure Agent must create a JSON file with the sample schema of the Google BigQuery table. The JSON file name is the same as the Google BigQuery table name. Alternatively, you can specify a storage path in Google Cloud Storage where the Secure Agent must create a JSON file with the sample schema of the Google BigQuery table. You can download the JSON file from the specified storage path in Google Cloud Storage to a local machine. The schema definition file is required if you configure complex connection mode in the following scenarios: - You add a Hierarchy Builder transformation in a mapping to read data from relational sources and write data to a Google BigQuery target. - You add a Hierarchy Parser transformation in a mapping to read data from a Google BigQuery source and write data to relational targets.
Use Legacy SQL For Custom Query[1]	Select this option to use a legacy SQL to define a custom query. If you clear this option, you must use a standard SQL to define a custom query. **Note:** Not applicable when you configure the Google BigQuery V2 connection in hybrid or complex mode.
Dataset Name for Custom Query[1]	When you define a custom query, you must specify a Google BigQuery dataset.
Region Id	The region name where the Google BigQuery dataset that you want to access resides. **Note:** You must ensure that you specify a bucket name or the bucket name and folder name in the **Storage Path** property that resides in the specified region. For more information about the regions supported by Google BigQuery, see Dataset locations.
Optional Properties[1]	Specifies whether you can configure source and target functionality through custom properties. You can select one of the following options: - None. If you do not want to configure any custom properties, select None. - Required. If you want to specify custom properties to configure the source and target functionalities. Default is None.
Provide Optional Properties[1]	Comma-separated key-value pairs of custom properties in the Google BigQuery V2 connection to configure certain source and target functionalities. Appears when you select **Required** in the Optional Properties. For more information about the list of custom properties that you can specify, see the Informatica Knowledge Base article: https://kb.informatica.com/faq/7/Pages/26/632722.aspx

Note: Ensure that you specify valid credentials in the connection properties. The test connection is successful even if you specify incorrect credentials in the connection properties.

Google Cloud Storage V2 connection properties

When you create a Google Cloud Storage V2 connection, configure the connection properties.

The following table describes the Google Cloud Storage connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Google Cloud Storage V2 connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks. You cannot run a database ingestion task or streaming ingestion task on a Hosted Agent or serverless runtime environment.
Service Account ID	The client_email value in the JSON file that you download after you create a service account.
Service Account Key	The private_key value in the JSON file that you download after you create a service account.
Project ID	The project_id value in the JSON file that you download after you create a service account. If you created multiple projects with the same service account, enter the ID of the project that contains the bucket that you want to connect to.
Private Key ID	The `private_key_id` value in the JSON file that you download after you create a service account. This property applies only to a database ingestion or streaming ingestion task.
Client ID	The `client_id` value in the JSON file that you download after you create a service account. This property applies only to a database ingestion or streaming ingestion task.
Bucket Name	The Google Cloud Storage bucket name that you want to connect to. When you select a source object , the Package Explorer lists files and folder available in the specified Google Cloud Storage bucket. If you do not specify a bucket name, you can select a bucket from the Package Explorer to select a source.

Configuring the proxy settings on Windows

If your organization uses an outgoing proxy server to connect to the Internet, the Secure Agent connects to Informatica Intelligent Cloud Services using the proxy server.
To configure the proxy server settings for the Secure Agent on a Windows machine, you must configure the proxy server settings through the Secure Agent Manager and the JVM options of the Secure Agent.

Restriction: These steps do not work for Mass Ingestion Databases.

Contact your network administrator for the proxy settings.

1. Click **Start** > **All Programs** > **Informatica Cloud Secure Agent** > **Informatica Cloud Secure Agent** to launch the Secure Agent Manager.

   The **Secure Agent Manager** displays the **Secure Agent** status.
2. Click **Proxy** on the Secure Agent Manager page.
3. Click **Use a Proxy Server** to enter the proxy server settings.
4. Configure the following proxy server details:

Field	Description
Proxy Host	Host name of the outgoing proxy server that the Secure Agent uses.
Proxy Port	Port number of the outgoing proxy server.
User Name	User name to connect to the outgoing proxy server.
Password	Password to connect to the outgoing proxy server.

5. Click **OK**.
6. Log in to Informatica Intelligent Cloud Services.
7. Open Administrator and select **Runtime Environments**.
8. Select the Secure Agent for which you want to configure a proxy server.
9. On the upper-right corner of the page, click **Edit**.
10. In the **System Configuration Details** section, select the **Type** as **Agent** for the CMI Streaming Agent Service.
11. To use a proxy server, add the following parameters in any **JVMOption** field and specify appropriate values for each parameter:

Parameter	Description
-Dproxy.host=	Host name of the outgoing HTTPS proxy server.
-Dproxy.port=	Port number of the outgoing HTTPS proxy server.
-Dproxy.user=	User name for the HTTPS proxy server.
-Dproxy.password=	Password for the HTTPS proxy server.

Note: You must specify the parameter and the value for the parameter enclosed in single quotation marks.

For example,

```
JVMOption1='-Dproxy.host=INPQ8583WI29'
JVMOption2='-Dproxy.port=8081'
JVMOption3='-Dproxy.user=adminuser'
JVMOption4='-Dproxy.password=password'
```

Note: You can configure only five **JVMOption** fields in the **System Configuration Details** section. To configure the remaining parameters, you must add the **JVMOption** fields in the **Custom Configuration Details** section. In the **Custom Configuration Details** section, select the **Type** as **Agent** for the CMI Streaming Agent Service, add the **JVMOption** fields, and specify the remaining parameters and appropriate values for each parameter.

12. Click **Save**.

    The Secure Agent restarts to apply the settings.

    Note: The session log does not record the proxy server details even if you have configured a proxy server.

Configuring the proxy settings on Linux

If your organization uses an outgoing proxy server to connect to the Internet, the Secure Agent connects to Informatica Intelligent Cloud Services through the proxy server.
You can update the proxy server settings defined for the Secure Agent from the command line. To configure the proxy server settings for the Secure Agent on a Linux machine, you must update the `proxy.ini` file and configure the JVM options of the Secure Agent.

Restriction: These steps do not work for Mass Ingestion Databases.

Contact your network administrator for the proxy settings.

1. Navigate to the following directory:

   ```
   <Secure Agent installation directory>/apps/agentcore/conf
   ```

2. To update the `proxy.ini` file, add the following parameters and specify appropriate values for each parameter:

   ```
   InfaAgent.ProxyHost=<proxy_server_hostname>
              InfaAgent.ProxyPort=<proxy_server_port>
              InfaAgent.ProxyUser=<user_name>
              InfaAgent.ProxyPassword=<password>
              InfaAgent.ProxyPasswordEncrypted=false
   ```

   For example,

   ```
   InfaAgent.ProxyHost=INW2PF0MT01V
              InfaAgent.ProxyPort=808
              InfaAgent.ProxyUser=user06
              InfaAgent.ProxyPassword=user06
              InfaAgent.ProxyPasswordEncrypted=false
   ```

3. Log in to Informatica Intelligent Cloud Services.
4. Open Administrator and select **Runtime Environments**.
5. Select the Secure Agent for which you want to configure a proxy server.
6. On the upper-right corner of the page, click **Edit**.
7. In the **System Configuration Details** section, select the **Type** as **Agent** for the CMI Streaming Agent Service.

8. To use a proxy server, add the following parameters in any **JVMOption** field and specify appropriate values for each parameter:

Parameter	Description
-Dproxy.host=	Host name of the outgoing HTTPS proxy server.
-Dproxy.port=	Port number of the outgoing HTTPS proxy server.
-Dproxy.user=	User name for the HTTPS proxy server.
-Dproxy.password=	Password for the HTTPS proxy server.

Note: You must specify the parameter and the value for the parameter enclosed in single quotation marks.

For example,

```
JVMOption1='-Dproxy.host=INPQ8583WI29'
JVMOption2='-Dproxy.port=8081'
JVMOption3='-Dproxy.user=adminuser'
JVMOption4='-Dproxy.password=password'
```

Note: You can configure only five **JVMOption** fields in the **System Configuration Details** section. To configure the remaining parameters, you must add the **JVMOption** fields in the **Custom Configuration Details** section. In the **Custom Configuration Details** section, select the **Type** as **Agent** for the CMI Streaming Agent Service, add the **JVMOption** fields, and specify the remaining parameters and appropriate values for each parameter.

9. Click **Save**.

   The Secure Agent restarts to apply the settings.

   Note: The session log does not record the proxy server details even if you have configured a proxy server.

Google PubSub - Mass Ingestion Streaming connection properties

When you define a Google PubSub Mass Ingestion Streaming connection, you must configure connection properties. You can use this connection type in streaming ingestion tasks, which you configure in the Mass Ingestion service.

The following table describes the Google PubSub connection properties:

Property	Description
Connection Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. Description that you use to identity the connection. The description must not exceed 4,000 characters.
Type	The **Google PubSub** connection type.

Property	Description
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Client Email	The `client_email` value available in the JSON file that you download after you create a service account.
Client ID	The `client_id` value available in the JSON file that you download after you create a service account.
Private Key ID	The `private_key_id` value available in the JSON file that you download after you create a service account.
Private Key	The `private_key` value available in the JSON file that you download after you create a service account.
Project ID	The `project_id` value available in the JSON file that you download after you create a service account.

Note: The test connection for the Google PubSub connector does not fail even if you enter incorrect values for **Client ID** and **Private Key ID**.

Hadoop Files V2 connection properties

When you set up a Hadoop Files V2 connection, you must configure the connection properties.

The following table describes the Hadoop Files V2 connection properties:

Connection property	Description
Connection Name	Name of the Hadoop Files V2 connection.
Description	Description of the connection. The description cannot exceed 765 characters.
Type	Type of connection. Select **Hadoop Files V2**.
Runtime Environment	The name of the runtime environment where you want to run the tasks.
User Name	Required to read data from HDFS. Enter a user name that has access to the single-node HDFS location to read data from or write data to.

Connection property	Description

NameNode URI	The URI to access HDFS.
	Use the following format to specify the name node URI in Cloudera, Amazon EMR, and Hortonworks distributions:
	`hdfs://<namenode>:<port>/`
	Where
	- `<namenode>` is the host name or IP address of the name node.
	- `<port>` is the port that the name node listens for remote procedure calls (RPC).
	If the Hadoop cluster is configured for high availability, you must copy the `fs.defaultFS` value in the `core-site.xml` file and append `/` to specify the name node URI.
	For example, the following snippet shows the `fs.defaultFS` value in a sample `core-site.xml` file:
	`<property>`
	`<name>fs.defaultFS</name>`
	`<value>hdfs://nameservice1</value>`
	`<source>core-site.xml</source>`
	`</property>`
	In the above snippet, the `fs.defaultFS` value is
	`hdfs://nameservice1`
	and the corresponding name node URI is
	`hdfs://nameservice1/`
	Note: Specify either the name node URI or the local path. Do not specify the name node URI if you want to read data from or write data to a local file system path.

Local Path	A local file system path to read and write data. Read the following conditions to specify the local path: - You must enter **NA** in local path if you specify the name node URI. If the local path does not contain **NA**, the name node URI does not work. - If you specify the name node URI and local path, the local path takes the preference. The connection uses the local path to run all tasks. - If you leave the local path blank, the agent configures the root directory (/) in the connection. The connection uses the local path to run all tasks. - If the file or directory is in the local system, enter the fully qualified path of the file or directory. For example, `/user/testdir` specifies the location of a directory in the local system. Default value for Local Path is NA.
Configuration Files Path	The directory that contains the Hadoop configuration files. **Note:** Copy the core-site.xml, hdfs-site.xml, and hive-site.xmlfrom the Hadoop cluster and add them to a folder in Linux Box.
Keytab File	The file that contains encrypted keys and Kerberos principals to authenticate the machine.
Principal Name	Users assigned to the superuser privilege can perform all the tasks that a user with the administrator privilege can perform.
Impersonation Username	You can enable different users to run mappings in a Hadoop cluster that uses Kerberos authentication or connect to sources and targets that use Kerberos authentication. To enable different users to run mappings or connect to big data sources and targets, you must configure user impersonation.

Note: When you read from or write to remote files, the **Name Node URI** and **Configuration Files Path** fields are mandatory. When you read from or write to local files only **Local Path** field is required.

JDBC V2 connection properties

When you set up a JDBC V2 connection, configure the connection properties.

The following table describes the JDBC V2 connection properties:

Property	Description
Connection Name	Name of the connection.
Description	Description of the connection.
Type	Type of connection. Select JDBC V2 from the list.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You can specify a Secure Agent or serverless runtime environment.
User Name	The user name to connect to the database.
Password	The password for the database user name.
Schema Name	Optional. The schema name. If you don't specify the schema name, all the schemas available in the database are listed.
JDBC Driver Class Name	Name of the JDBC driver class. To connect to Aurora PostgreSQL, specify the following driver class name: org.postgresql.Driver For more information about which driver class to use with specific databases, see the corresponding third-party vendor documentation.
Connection String	Connection string to connect to the database. Use the following format to specify the connection string: `jdbc:<subprotocol>:<subname>` For example, the connection string for the Aurora PostgreSQL database type is `jdbc:postgresql://<host>:<port>[/dbname]`. For more information about the connection string to use with specific drivers, see the corresponding third-party vendor documentation.
Additional Security Properties	Masks sensitive and confidential data of the connection string that you don't want to display in the session log. Specify the part of the connection string that you want to mask. When you create a connection, the string you enter in this field appends to the string that you specified in the **Connection String** field.
Database Type	The database type to which you want to connect. You can select one of the following database types: - PostgreSQL. Connect to the Aurora PostgreSQL database hosted in the Amazon Web Services or the Microsoft Azure environment. - Azure SQL Database. Connect to Azure SQL Database hosted in the Microsoft Azure environment. - Others. Connect to any database that supports the Type 4 JDBC driver.

Property	Description
Support Mixed-Case Identifiers	Indicates whether the database supports case-sensitive identifiers. When enabled, the Secure Agent encloses all identifiers within the character selected for the SQL Identifier Character property.
SQL Identifier Character	Type of character that the database uses to enclose delimited identifiers in SQL queries. The available characters depend on the database type. Select **None** if the database uses regular identifiers. When the Secure Agent generates SQL queries, it does not place delimited characters around any identifiers. Select a character if the database uses delimited identifiers. When the Secure Agent generates SQL queries, it encloses delimited identifiers within this character.

JMS connection properties

When you set up a JMS connection, you must configure the connection properties.

The following table describes the connection properties for the JMS connection:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
Description	Optional. Description that you can use to identity the connection. The description cannot exceed 4,000 characters.
Type	The JMS connection type. If you do not see the connection type, go to the **Add-On Connectors** page to install the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Connection URL	URL of the JNDI naming provider. For example, in IBM MQ it is the directory location that contains the .bindings file.
JNDI User Name	Optional. User name to connect to the JNDI context factory.
JNDI Password	Optional. The password of the user account that you use to connect to the JNDI context factory.
JNDI Context Factory	The JMS provider specific initial JNDI context factory implementation for connecting to the JNDI service. This value is a fully qualified class name of the Initial Context Factory. For example, the class name of the Initial Context Factory for ActiveMQ is `org.apache.activemq.jndi.ActiveMQInitialContextFactory` For more information, see the documentation of the JMS provider.

Property	Description
JNDI Package Prefixes	A colon-delimited list of package prefixes to use when loading URL context factories. These are the package prefixes for the name of the factory class that will create a URL context factory. For more information about the values, see the documentation of the JMS provider.
JMS Connection Factory	The name of the object in the JNDI server that enables the JMS Client to create JMS connections. For example, `jms/QCF` or `jmsSalesSystem`.
JMS Connection User Name	Optional. User name to connect to the JMS connection factory.
JMS Connection Password	Optional. The password of the user account that you use to connect to the JMS connection factory.

Note: Ensure to copy the external JMS `JAR` files to the following location:

```
<Secure_Agent_home>/ext/connectors/thirdparty/infa.jms
```

After copying the external JMS `JAR` files, restart the Secure Agent.

Kafka connection properties

When you set up a Kafka connection, you must configure the connection properties.

The following table describes the Kafka connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive. It must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
Description	Optional. Description that you use to identity the connection. The description cannot exceed 4,000 characters.
Type	The Kafka connection type. If you do not see the connection type, go to the **Add-On Connectors** page in Administrator to install the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Kafka Broker List	Comma-separated list of the Kafka brokers. To list a Kafka broker, use the following format: `<HostName>:<PortNumber>` **Note:** When you connect to a Kafka broker over SSL, you must specify the fully qualified domain name for the host name. Otherwise, the test connection fails with SSL handshake error.

Property	Description
Retry Timeout	Optional. Number of seconds after which the Secure Agent attempts to reconnect to the Kafka broker to read or write data. Default is 180 seconds. This property is not used by Mass Ingestion Databases. You can specify an equivalent Kafka property in **Additional Connection Properties**.
Kafka Broker Version	Kafka message broker version. The only valid value is Apache 0.10.1.1 and above. Optional for a streaming ingestion task.
Additional Connection Properties	Optional. Comma-separated list of additional configuration properties of the Kafka producer or consumer. For a streaming ingestion task, ensure that you set the `<kerberos name>` property if you configure `<Security Protocol>` as `SASL_PLAINTEXT` or `SASL_SSL`.

Schema Registry URL	Location and port of the Confluent schema registry service to access Avro sources and targets in Kafka.
	To list a schema registry URL, use the following format:
	`<https>://<HostName or IP>:<PortNumber>`
	or
	`<http>://<HostName or IP>:<PortNumber>`
	Example for the schema registry URL:
	`https://kafkarnd.informatica.com:8082`
	or
	`http://10.65.146.181:8084`
	Applies only when you import a Kafka topic in Avro format that uses the Confluent schema registry to store the metadata.
	This property is not used by Mass Ingestion Databases. You can specify an equivalent Kafka property in **Additional Connection Properties**.

SSL Mode	Required. Determines the encryption type to use for the connection. You can choose a mode from the following SSL modes: - Disabled. Establishes an unencrypted connection to the Kafka broker. - One-way. Establishes an encrypted connection to the Kafka broker using truststore file and truststore password. - Two-way. Establishes an encrypted connection to the Kafka broker using truststore file, truststore password, keystore file, and keystore password. This property is not used by Mass Ingestion Databases. You can specify an equivalent Kafka property in **Additional Connection Properties**.
SSL TrustStore File Path	Required when you use the one-way or two-way SSL mode. Absolute path and file name of the SSL truststore file that contains the SSL certificate to connect to the Kafka broker.
SSL TrustStore Password	Required when you use the one-way or two-way SSL mode. Password for the SSL truststore.
SSL KeyStore File Path	Required when you use the two-way SSL mode. Absolute path and file name of the SSL keystore file that contains private keys and certificates to connect to the Kafka broker.

Property	Description
SSL KeyStore Password	Required when you use the two-way SSL mode. Password for the SSL keystore.
Additional Security Properties	Optional. Comma-separated list of additional configuration properties to connect to the Kafka broker in a secure way. If you specify two different values for the same property in **Additional Connection Properties** and **Additional Security Properties**, the value in **Additional Security Properties** overrides the value in **Additional Connection Properties**. This property is not used by Mass Ingestion Databases.

Schema Registry Security Configuration Properties

When you configure the **Schema Registry URL** connection property, you can configure the schema registry security configuration properties. You can configure one-way SSL, two-way SSL, and basic authentication to connect to the Confluent schema registry in a secure way.

The following table describes the security properties for the Kafka connection when you use the Confluent schema registry:

Property	Description
SSL Mode Schema Registry[1]	Required. Determines the encryption type to use for the connection. You can choose a mode from the following SSL modes: - Disabled. Establishes an unencrypted connection to the Confluent schema registry. - One-way. Establishes an encrypted connection to the Confluent schema registry using truststore file and truststore password. - Two-way. Establishes an encrypted connection to the Confluent schema registry using truststore file, truststore password, keystore file, and keystore password. This property is not used by Mass Ingestion Databases. You can specify an equivalent Kafka property in **Additional Connection Properties**.
SSL TrustStore File Path Schema Registry[1]	Required when you use the one-way or two-way SSL mode. Absolute path and file name of the SSL truststore file that contains the SSL certificate to connect to the Confluent schema registry.
SSL TrustStore Password Schema Registry[1]	Required when you use the one-way or two-way SSL mode. Password for the SSL truststore.
SSL KeyStore File Path Schema Registry[1]	Required when you use the two-way SSL mode. Absolute path and file name of the SSL keystore file that contains private keys and certificates to connect to the Confluent schema registry.
SSL KeyStore Password Schema Registry[1]	Required when you use the two-way SSL mode. Password for the SSL keystore.

Property	Description
Additional Security Properties Schema Registry	Optional. Comma-separated list of additional security properties to connect to the Confluent schema registry in a secure way. For example, when you configure basic authentication to establish a secure communication with Confluent schema registry, specify the following value: `basic.auth.credentials.source=USER_INFO,basic.auth.user.info=<username>:<password>` If you specify two different values for the same property in **Additional Connection Properties** and **Additional Security Properties Schema Registry**, the value in **Additional Security Properties Schema Registry** overrides the value in **Additional Connection Properties**. This property is not used by Mass Ingestion Databases.
[1] Does not apply to mappings.	

Configuring the krb5.conf file to read data from or write to a Kerberised Kafka cluster

To read from or write to a Kerberised Kafka cluster, configure the default realm, KDC, and Kafka advanced source or target properties.

You can configure Kerberos authentication for a Kafka client by placing the required Kerberos configuration files on the Secure Agent machine and specifying the required JAAS configuration in the Kafka connection. The JAAS configuration defines the keytab and principal details that the Kafka broker must use to authenticate the Kafka client.

Note: This topic is not applicable to Mass Ingestion Applications and Mass Ingestion Databases. Mass Ingestion Applications and Mass Ingestion Databases does not yet support this functionality.

Before you read from or write to a Kerberised Kafka cluster, perform the following tasks:

1. Ensure that you have the `krb5.conf` file for the Kerberised Kafka cluster.
2. Configure the default realm and KDC. If the default `/etc/krb5.conf` file is not configured or you want to change the configuration, add the following lines to the `/etc/krb5.conf` file:

```
[libdefaults]
default_realm = <REALM NAME>
dns_lookup_realm = false
dns_lookup_kdc = false
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true

[realms]
<REALM NAME> = {
kdc = <Location where KDC is installed>
admin_server = <Location where KDC is installed>
                                }
 [domain_realm]
.<domain name or hostname> = <KERBEROS DOMAIN NAME>
<domain name or hostname> = <KERBEROS DOMAIN NAME>
```

3. To pass a static JAAS configuration file into the JVM using the `java.security.auth.login.config` property at runtime, perform the following tasks:
   a. Ensure that you have JAAS configuration file.

      For information about creating JAAS configuration and configuring keytab for Kafka clients, see the Apache Kafka documentation at https://kafka.apache.org/0101/documentation/#security

For example, the JAAS configuration file can contain the following lines of configuration:

```
//Kafka Client Authentication. Used for client to kafka broker connection
KafkaClient {
com.sun.security.auth.module.Krb5LoginModule required
doNotPrompt=true
useKeyTab=true
storeKey=true
keyTab="<path to Kafka keytab  file>/<Kafka keytab file name>"
principal="<principal name>"
client=true
};
```

b. Place the JAAS config file and keytab file in the same location on all the secure agents.

Informatica recommends that you place the files in a location that is accessible by all the secure agents in the runtime environment. For example, `/etc` or `/temp`.

c. Configure the following properties:

Kafka connection

Configure the **Additional Connection Properties** in a Kafka connection and specify the value in the following format:

```
security.protocol=SASL_PLAINTEXT,sasl.kerberos.service.name=kafka,sasl.mechan
ism=GSSAPI
```

Sources

Configure the **Consumer Configuration Properties** in the advanced source properties to override the value specified in **Additional Connection Properties** in a Kafka connection. Specify the value in the following format:

```
security.protocol=SASL_PLAINTEXT,sasl.kerberos.service.name=kafka,sasl.mechan
ism=GSSAPI
```

Targets

Configure the **Producer Configuration Properties** in the advanced target properties to override the value specified in **Additional Connection Properties** in a Kafka connection. Specify the value in the following format:

```
security.protocol=SASL_PLAINTEXT,sasl.kerberos.service.name=kafka,sasl.mechan
ism=GSSAPI
```

4. To embed the JAAS configuration in the `sasl.jaas.config` configuration property, configure the following properties:

Kafka connection

Configure the **Additional Connection Properties** in a Kafka connection and specify the value in the following format:

```
security.protocol=SASL_PLAINTEXT,sasl.kerberos.service.name=kafka,sasl.mechanism=
GSSAPI,
sasl.jaas.config=com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
storeKey=true doNotPrompt=true serviceName="<service_name>" keyTab="<location of
keytab file>"
client=true principal="<principal_name>";
```

Sources

Configure the **Consumer Configuration Properties** in the advanced source properties to override the value specified in **Kerberos Configuration Properties** in a Kafka connection. Specify the value in the following format:

```
security.protocol=SASL_PLAINTEXT,sasl.kerberos.service.name=kafka,sasl.mechanism=
GSSAPI,
```

```
sasl.jaas.config=com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
storeKey=true doNotPrompt=true serviceName="<service_name>" keyTab="<location of
keytab file>"
client=true principal="<principal_name>";
```

Targets

Configure the **Producer Configuration Properties** in the advanced target properties to override the value specified in **Kerberos Configuration Properties** in a Kafka connection. Specify the value in the following format:

```
security.protocol=SASL_PLAINTEXT,sasl.kerberos.service.name=kafka,sasl.mechanism=
GSSAPI,
sasl.jaas.config=com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
storeKey=true doNotPrompt=true serviceName="<service_name>" keyTab="<location of
keytab file>"
client=true principal="<principal_name>";
```

Configuring SASL PLAIN authentication for a Kafka cluster

In the Kafka connection, you can configure PLAIN security for the Kafka broker to connect to a Kafka broker. To read data from or write data to a Kafka broker with SASL PLAIN authentication, configure the Kafka connection properties. To override the properties defined in the Kafka connection, you can configure the advanced source or target properties.

You can configure SASL PLAIN authentication so that the Kafka broker can authenticate the Kafka producer and the Kafka consumer. Kafka uses the Java Authentication and Authorization Service (JAAS) for SASL PLAIN authentication. To enable SASL PLAIN authentication, you must specify the SASL mechanism as PLAIN. You must also provide the formatted JAAS configuration that the Kafka broker must use for authentication. The JAAS configuration defines the username, password, that the Kafka broker must use to authenticate the Kafka client.

Note: This topic is not applicable to Mass Ingestion Applications and Mass Ingestion Databases. Mass Ingestion Applications and Mass Ingestion Databases does not yet support this functionality.

Configure the following properties:

Kafka connection

Configure the **Additional Connection Properties** or **Additional Security Properties** property in the Kafka connection and specify the value in the following format:

```
security.protocol=SASL_SSL,sasl.mechanism=PLAIN,sasl.jaas.config=org.apache.kafka.com
mon.security.plain.PlainLoginModule required username=<username> password=<password>
```

In the **Security Configuration Section**, select **One-Way** as the **SSL Mode** and specify the SSL TrustStore File Path and SSL TrustStore Password.

Sources

Configure the **Consumer Configuration Properties** property in the advanced source properties to override the value that you specified in the **Additional Connection Properties** property in the Kafka connection. Specify the value in the following format:

```
security.protocol=SASL_SSL,sasl.mechanism=PLAIN,sasl.jaas.config=org.apache.kafka.com
mon.security.plain.PlainLoginModule required username=<username> password=<password>
```

Targets

Configure the **Producer Configuration Properties** property in the advanced target properties to override the value that you specified in the **Additional Connection Properties** property in the Kafka connection. Specify the value in the following format:

```
security.protocol=SASL_SSL,sasl.mechanism=PLAIN,sasl.jaas.config=org.apache.kafka.com
mon.security.plain.PlainLoginModule required username=<username> password=<password>
```

Configuring SASL PLAIN authentication for an Azure Event Hub Kafka broker

In the Kafka connection, you can configure PLAIN security for the Kafka broker to connect to an Azure Event Hub Kafka broker. When you connect to an Azure Event Hub Kakfa broker, the password defines the endpoint URL that contains the fully qualified domain name (FQDN) of the Event Hub namespace, shared access key name, and shared access key required to connect to an Azure Event Hub Kafka broker. Configure the SSL Mode as One-Way and provide the path to a trusted root certificate on your file system for SSL TrustStore File Path.

To connect to an Azure Event Hub Kafka broker, configure any of the above properties and specify the value in the following format:

```
security.protocol=SASL_SSL,sasl.mechanism=PLAIN,sasl.kerberos.service.name=Kafka,sasl.jaa
s.config=org.apache.kafka.common.security.plain.PlainLoginModule required
username="$ConnectionString" password="Endpoint=sb://<FQDN>/;SharedAccessKeyName=<key
name>;SharedAccessKey=<shared access key>=";
```

Configuring SASL_SSL authentication for a Cloud Confluent Kafka cluster

In the Kafka connection, you can configure SSL security for encryption and authentication while connecting to a Kafka broker. To read data from or write data to a Confluent Kafka broker with `SASL_SSL` authentication, configure the Kafka connection properties. To override the properties defined in the Kafka connection, you can configure the advanced source or target properties.

Note: This topic is not applicable to Mass Ingestion Applications and Mass Ingestion Databases. Mass Ingestion Applications and Mass Ingestion Databases does not yet support this functionality.

Configure the following properties:

Property	Values
Additional Connection Properties	`security.protocol=SASL_SSL,sasl.kerberos.service.name=kafka,ssl.endpoint.identification.algorit` `required username=<> password=<>`
SSL Mode	One-way
SSL TrustStore File Path	Use `cacert` file of agent JDK. For example: `/root/staging/infaagent/jdk/jre/lib/security/cacerts`
SSL TrustStore Password	Password for the SSL truststore.

Marketo V3 connection properties

When you set up a Marketo V3 connection, configure the connection properties.

The following table describes the Marketo V3 connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Marketo V3 connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
client_ID	The client ID of the custom service required to generate a valid access token.
client_secret	The client secret of the Marketo custom service required to generate a valid access token.
grant_type	Marketo supports only the client_credentials grant type.
REST API URL	The URL has the following format: https://<Host name of the Marketo Rest API Server>. Contact the Marketo Administrator for the REST API URL.
Bypass Proxy	**Note:** This property is not applicable to connections configured for application ingestion tasks.

Microsoft Azure Blob Storage V3 connection properties

When you set up a Microsoft Azure Blob Storage V3 connection, configure the connection properties.

The following table describes the Microsoft Azure Blob Storage V3 connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Microsoft Azure Blob Storage V3 connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. Specify a Secure Agent, Hosted Agent, or serverless runtime environment.
Account Name	Microsoft Azure Blob Storage account name.

Property	Description
Authentication Type	Authentication type to access the Microsoft Azure Blob Storage account. Select one of the following options: - Shared Key Authentication. Uses the account key to connect to Microsoft Azure Blob Storage. - Shared Access Signature. Uses the SAS token to connect to Microsoft Azure Blob Storage. Use the SAS token to grant access to the resources in the storage account or container for a specific time range without sharing the account key.
Account Key	Applies to shared key authentication. The account key for the Microsoft Azure Blob Storage account.
SAS Token	Applies to shared access signature. The shared access signature token generated in the Azure portal.
Container Name	Microsoft Azure Blob Storage container name.
Endpoint Suffix	Type of Microsoft Azure endpoints. Select one of the following options: - core.windows.net. Connects to Azure endpoints. - core.usgovcloudapi.net. Connects to Azure Government endpoints. - core.chinacloudapi.cn. Not applicable. Default is core.windows.net.

Microsoft Azure Data Lake Storage Gen2 connection properties

When you set up a Microsoft Azure Data Lake Storage Gen2 connection, configure the connection properties.

The following table describes the Microsoft Azure Data Lake Storage Gen2 connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Microsoft Azure Data Lake Storage Gen2 connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run a database ingestion or streaming ingestion task on a Hosted Agent or serverless runtime environment.
Account Name	Microsoft Azure Data Lake Storage Gen2 account name or the service name.

Property	Description
Authentication Type	Authentication type to access the Microsoft Azure Data Lake Storage Gen2 account. Select one of the following options: - Service Principal Authentication. Uses the client ID, client secret, and tenant ID to connect to Microsoft Azure Data Lake Storage Gen2. - Shared Key Authentication. Uses the account key to connect to Microsoft Azure Data Lake Storage Gen2. - Managed Identity Authentication. Select to authenticate using identities that are assigned to applications in Azure to access Azure resources in Microsoft Azure Data Lake Storage Gen2. **Note:** Mass Ingestion Streaming does not support shared key authentication or managed identity authentication.
Client ID	Applies to Service Principal Authentication and Managed Identity Authentication. The client ID of your application. To use service principal authentication, specify the application ID or client ID for your application registered in the Azure Active Directory. To use managed identity authentication, specify the client ID for the user-assigned managed identity. If the permission is provided by system-assigned managed identity, leave the field empty. If there is no system-assigned identity but only a single user-assigned managed identity, you may also leave the field empty.
Client Secret	Applies to Service Principal Authentication. The client secret key to complete the OAuth authentication in the Azure Active Directory.
Tenant ID	Applies to Service Principal Authentication. The directory ID of the Azure Active Directory.
Account Key	Applies to Shared Key Authentication. The account key for the Microsoft Azure Data Lake Storage Gen2 account.
File System Name	The name of the file system in the Microsoft Azure Data Lake Storage Gen2 account.
Directory Path	The path of an existing directory without the file system name. You can select one of the following syntax: - `/` for root directory - `/dir1` - `dir1/dir2` There is no default directory.
Adls Gen2 End-point	The type of Microsoft Azure endpoints. Select one of the following endpoints: - core.windows.net. Connects to Azure endpoints. - core.usgovcloudapi.net. Connects to US government Microsoft Azure Data Lake storage Gen2 endpoints. - core.chinacloudapi.cn. Connects to Microsoft Azure Data Lake storage Gen2 endpoints in the China region. Default is core.windows.net.

Microsoft Azure Event Hub connection properties

When you set up an Azure Event Hub connection, you must configure the connection properties.

The following table describes the Azure Event Hub connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive. It must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. Description that you use to identity the connection. The description cannot exceed 4,000 characters.
Type	The Azure Event Hub connection type. If you do not see the connection type, go to the **Add-On Connectors** page in Administrator to install the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Tenant ID	The ID of the tenant that the data belongs to. This ID is the Directory ID of the Azure Active Directory.
Subscription ID	The ID of the Azure subscription.
Resource Group Name	The name of the Azure resource group associated with the Event Hub namespace.
Client Application ID	The ID of the application created under the Azure Active Directory.
Client Secret Key	The secret key generated for the application.
Event Hub Namespace	The name of the Event Hub namespace that is associated with the resource group name.
Shared Access Policy Name	Optional. The name of the Event Hub Namespace Shared Access Policy. The policy must apply to all data objects that are associated with this connection. To read from Event Hubs, you must have Listen permission. To write to an Event Hub, the policy must have Send permission.
Shared Access Policy Primary Key	Optional. The primary key of the Event Hub Namespace Shared Access Policy.

Microsoft Azure Synapse Analytics Database Ingestion connection properties

When you define a Microsoft Azure Synapse Analytics Database Ingestion connection, you must configure connection properties. You can use this connection type in application ingestion tasks and database ingestion tasks, which you configure in the Mass Ingestion service.

Note: Some properties are for Microsoft Azure Data Lake Storage Gen2. Mass Ingestion Applications and Mass Ingestion Databases use Microsoft Azure Data Lake Storage Gen2 to stage data in files before sending the data to the Microsoft Azure Synapse Analytics target tables.

The following table describes the connection properties:

Property	Description
Connection Name	A name for the connection. This name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Spaces at the beginning or end of the name are trimmed and are not saved as part of the name. Maximum length is 100 characters. Connection names are not case sensitive.
Description	An optional description for the connection. Maximum length is 255 characters.
Type	The type of connection. Ensure that the type is for Microsoft Azure Synapse Analytics - Database Ingestion.
Runtime Environment	The name of the runtime environment where you want to run the application ingestion tasks or database ingestion tasks. You define runtime environments in Administrator. **Note:** You cannot run application ingestion tasks and database ingestion tasks on a Hosted Agent or serverless runtime environment.

Azure Synapse Analytics JDBC URL	The Microsoft Azure Synapse Analytics (formerly SQL Data Warehouse) JDBC connection string.
	Example connection string for Microsoft SQL Server authentication:
	`jdbc:sqlserver://server.database.windows.net:1433;database=database`
	Example connection string for Azure Active Directory (AAD) authentication:
	`jdbc:sqlserver://server.database.windows.net:1433;database=database;encrypt=true;trustServerCertificate=false;hostNameInCertificate=*.database.windows.net;loginTimeout=30;Authentication=ActiveDirectoryPassword;`
	Note: The default authentication type is Microsoft SQL Server authentication.

Azure Synapse Analytics JDBC Username	The user name to use for connecting to the Microsoft Azure Synapse Analytics account. Provide the AAD user name for AAD authentication.
Azure Synapse Analytics JDBC Password	The password to use for connecting to the Microsoft Azure Synapse Analytics account.
Azure Synapse Analytics Schema Name	The name of the schema in the Microsoft Azure Synapse Analytics target.
ADLS Gen2 Account Name	The name of the Microsoft Azure Data Lake Storage Gen2 account.
Client Id	The ID of your client application for completing the OAuth Authentication in the Active Directory.
Client Secret	The client secret key for completing the OAuth Authentication in the Active Directory.
Directory	The Microsoft Azure Data Lake Storage Gen2 directory that Mass Ingestion Applications and Mass Ingestion Databases uses to stage data in files. The default is the root directory.
Filesystem Name	The name of an existing file system in the Microsoft Azure Data Lake Storage Gen2 account.
Tenant ID	The Directory ID of the Azure Active Directory.

Microsoft Azure Synapse SQL connection properties

When you set up a Microsoft Azure Synapse SQL connection, configure the connection properties.

The following table describes the Microsoft Azure Synapse SQL connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Microsoft Azure Synapse SQL connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. Select a Secure Agent, Hosted Agent, or serverless runtime environment.

Azure DW JDBC URL	The Microsoft Azure Synapse SQL JDBC connection string.
	Enter the connection string in the following format for Microsoft SQL Server authentication:
	`jdbc:sqlserver://<Server>.database.windows.net:1433;database=<Database>`
	Enter the connection string in the following format for Azure Active Directory (AAD) authentication:
	`jdbc:sqlserver://<Server>.database.windows.net:1433;database=<Database>;encrypt=true;trustServerCertificate=false;hostNameInCertificate=*.database.windows.net;loginTimeout=30;Authentication=ActiveDirectoryPassword;`
	Default is Microsoft SQL Server authentication.

Azure DW JDBC Username	User name to connect to the Microsoft Azure Synapse SQL account. Provide AAD user name for AAD authentication.
Azure DW JDBC Password	Password to connect to the Microsoft Azure Synapse SQL account.
Azure DW Schema Name	Name of the schema in Microsoft Azure Synapse SQL.
Azure Storage Type	Type of Azure storage to stage the files. Select one of the following storage types: - Azure Blob. Uses Microsoft Azure Blob Storage to stage the files. - ADLS Gen2. Uses Microsoft Azure Data Lake Storage Gen2 to stage the files. Default is Azure Blob.

Property	Description
Authentication Type	Authentication type to connect to Azure storage to stage the files. Select one of the following options: - Shared Key Authentication. Uses the account name and account key to connect to Microsoft Azure Blob Storage or Microsoft Azure Data Lake Storage Gen2. - Service Principal Authentication. Applies to Microsoft Azure Data Lake Storage Gen2. Uses the client ID, client secret, and tenant ID to connect to Microsoft Azure Data Lake Storage Gen2. To use Service Principal authentication, register an application in the Azure Active Directory, generate a client secret, and then assign the Storage Blob Contributor role to the application. - Managed Identity Authentication. Applies to Microsoft Azure Data Lake Storage Gen2. Select to authenticate using identities that are assigned to applications in Azure to access Azure resources in Microsoft Azure Data Lake Storage Gen2. In a file ingestion task, if you select Microsoft Azure Synapse SQL with Managed Identity authentication type as the target, then you must select Microsoft Azure Data Lake Storage Gen2 as the source.
Azure Blob Account Name	Applies to Shared Key Authentication for Microsoft Azure Blob Storage. Name of the Microsoft Azure Blob Storage account to stage the files.
Azure Blob Account Key	Applies to Shared Key Authentication for Microsoft Azure Blob Storage. The Microsoft Azure Blob Storage access key to stage the files.
Container Name	Applies to Microsoft Azure Blob Storage. The name of the container in the Azure Blob Storage account.
ADLS Gen2 Storage Account Name	Applies to Shared Key Authentication and Service Principal Authentication for Microsoft Azure Data Lake Storage Gen2. Name of the Microsoft Azure Data Lake Storage Gen2 account to stage the files.
ADLS Gen2 Account Key	Applies to Shared Key Authentication for Microsoft Azure Data Lake Storage Gen2. The Microsoft Azure Data Lake Storage Gen2 access key to stage the files.
Client ID	Applies to Service Principal Authentication and Managed Identity Authentication for Microsoft Azure Data Lake Storage Gen2. The client ID of your application. To use service principal authentication, enter the application ID or client ID for your application registered in the Azure Active Directory. To use managed identity authentication, enter the client ID for the user-assigned managed identity. If the managed identity is system-assigned, leave the field empty.
Client Secret	Applies to Service Principal Authentication for Microsoft Azure Data Lake Storage Gen2. The client secret for your application.
Tenant ID	Applies to Service Principal Authentication for Microsoft Azure Data Lake Storage Gen2. The directory ID or tenant ID for your application.
File System Name	Applies to Microsoft Azure Data Lake Storage Gen2. The name of the file system in the Microsoft Azure Data Lake Storage Gen2 account.

Property	Description
Blob End-point	Type of Microsoft Azure endpoints. Select one of the following endpoints: - core.windows.net. Connects to Azure endpoints. - core.usgovcloudapi.net. Connects to US Government Microsoft Azure Synapse SQL endpoints. - core.chinacloudapi.cn. Connects to Microsoft Azure Synapse SQL endpoints in the China region. Default is core.windows.net.
VNet Rule	Enable to connect to a Microsoft Azure Synapse SQL endpoint residing in a virtual network (VNet). When you use a serverless runtime environment, you cannot connect to a Microsoft Azure Synapse SQL endpoint residing in a virtual network.

Microsoft Dynamics 365 Mass Ingestion connection properties

When you set up a Microsoft Dynamics 365 Mass Ingestion connection, you must configure the connection properties.

The Microsoft Dynamics 365 Mass Ingestion connection requires a native application that is registered in Azure Active Directory (Azure AD) to access the Microsoft Dynamics 365 data. Before you configure the connection, you must register an application in Azure AD to allow the connection to access the Microsoft Dynamics 365 data. For more information about registering an application in Azure AD, see the Microsoft documentation.

The properties of a Microsoft Dynamics 365 Mass Ingestion connection vary based on the authentication method that you specify for the connection. When you create a connection, you can select one of the following authentication methods:

- **OAuth 2.0 Username-Password Flow**: Authenticates the connection by using the Microsoft Dynamics 365 account login credentials and the client ID of the application registered in Azure AD.
- **OAuth 2.0 Client Secret Flow**: Authenticates the connection by using the client ID and client secret of the application registered in Azure AD.
- **OAuth 2.0 JWT Bearer Flow**: Authenticates the connection by using a X509 Public Key Infrastructure (PKI) certificate and a JSON Web Token (JWT). Use this authentication method to gain secured access to Microsoft Dynamics 365 without sharing sensitive information, such as client secret and Microsoft Dynamics 365 account login credentials.

Connection properties for OAuth 2.0 Username-Password Flow authentication

The following table describes the connection properties for a Microsoft Dynamics 365 Mass Ingestion connection configured with OAuth 2.0 Username-Password Flow authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
User Name	User name of the Microsoft Dynamics 365 account.
Password	Password for the Microsoft Dynamics 365 account.

Connection property	Description
Client ID	Client ID of the application registered in Azure AD.
Resource URL	URL of the Microsoft Dynamics 365 organization. You must enter the resource URL in the following format: `https://<Microsoft_Dynamics_365_org_name>.api.crm8.dynamics.com`
OAuth Token URL	OAuth 2.0 token endpoint of the Microsoft Dynamics 365 organization. The application that is registered in Azure AD sends the access token requests to this endpoint. You must enter the following value in this field: `https://login.windows.net/common/oauth2/token`

Note: For more information about the OAuth 2.0 Username-Password Flow authentication method, see the Microsoft Dynamics 365 documentation.

Connection properties for OAuth 2.0 Client Secret Flow authentication

The following table describes the connection properties for a Microsoft Dynamics 365 Mass Ingestion connection configured with OAuth 2.0 Client Secret Flow authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Client ID	Client ID of the application registered in Azure AD.
Client Secret	Client secret of the application registered in Azure AD.
Resource URL	URL of the Microsoft Dynamics 365 organization. You must enter the resource URL in the following format: `https://<Microsoft_Dynamics_365_org_name>.api.crm8.dynamics.com`
OAuth Token URL	OAuth 2.0 token endpoint of the Microsoft Dynamics 365 organization. The application that is registered in Azure AD sends the access token requests to this endpoint. You must enter the following value in this field: `https://login.microsoftonline.com/<tentant_id>/oauth2/token`

Note: For more information about the OAuth 2.0 Client Secret Flow authentication method, see the Microsoft Dynamics 365 documentation.

Connection properties for OAuth 2.0 JWT Bearer Flow authentication

The following table describes the connection properties for a Microsoft Dynamics 365 Mass Ingestion connection configured with OAuth 2.0 JWT Bearer Flow authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Client ID	Client ID of the application registered in Azure AD.
Certificate Signature	Base64URL string that encodes the hexadecimal value which represents the SHA-1 thumbprint of the X509 certificate.
Keystore Path	Absolute path to the keystore file that contains the X509 certificate required to validate a JSON Web Token (JWT) and establish a secure connection with Microsoft Dynamics 365. The keystore file must be in the Java KeyStore (JKS) format.
Keystore Password	Password for the keystore file.
Private Key Alias	Alias name of the private key used to sign the JWT.
Private Key Password	Password for the private key.
Audience for JWT	URL of the Microsoft Dynamics 365 resource server to which the application that is registered in Azure AD sends the JWT for validation. You must enter the address in the following format: `https://login.microsoftonline.com/<tentant_id>/oauth2/token`
Resource URL	URL of the Microsoft Dynamics 365 organization. You must enter the resource URL in the following format: `https://<Microsoft_Dynamics_365_org_name>.api.crm8.dynamics.com`
OAuth Token URL	OAuth 2.0 token endpoint of the Microsoft Dynamics 365 organization. The application that is registered in Azure AD sends the access token requests to this endpoint. You must enter the following value in this field: `https://login.microsoftonline.com/<tentant_id>/oauth2/token`

Note: For more information about the OAuth 2.0 Client Secret Flow authentication method, see the Microsoft Dynamics 365 documentation.

Microsoft SQL Server connection properties

When you set up a Microsoft SQL Server connection, configure the connection properties.

The following table describes the Microsoft SQL Server connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	Type of connection. Select Microsoft SQL Server from the list.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run a database ingestion task on a Hosted Agent or serverless runtime environment.
SQL Server Version	Microsoft SQL Server database version.
Authentication Mode	Authentication method to access Microsoft SQL Server. Select one of the following methods: - SQL Server Authentication. Uses your Microsoft SQL Server user name and password to access Microsoft SQL Server. - Windows Authentication (Deprecated). Uses the Microsoft Windows authentication to access Microsoft SQL Server. This option is available when you access Data Integration or Mass Ingestion by using Microsoft Windows. When you choose this option, you don't need to enter credentials to access Microsoft SQL Server and ensure that the user account that starts the Secure Agent service is available in the Microsoft SQL Server database. If you use Mass Ingestion Databases and want to use Windows authentication, select this option. **Note:** Windows authentication is not certified for Microsoft SQL Server 2017 version hosted on Linux. You can't configure Windows Authentication when you use a serverless runtime environment. - Active Directory Password. Uses the Azure Active Directory user name and password to authenticate and access the Microsoft Azure SQL Database. - Windows Authentication v2. Uses this authentication method to access Microsoft SQL Server from Data Integration using the agent hosted on a Linux or Windows machine. When you choose this option, enter your domain name and Microsoft Windows credentials to access Microsoft SQL Server and ensure that the user account that starts the Secure Agent service is available in the Microsoft SQL Server database when you use the Windows agent.
Domain	Applies to Windows Authentication v2. The domain name of the Windows user.

Property	Description
User Name	User name for the database login. The user name can't contain a semicolon. To connect to Microsoft Azure SQL Database, specify the user name in the following format: `username@host` For Windows Authentication v2, specify the Windows NT user name. **Note:** This property is not applicable if you use the Windows Authentication mode to access Microsoft SQL Server.
Password	Password for the database login. The password can't contain a semicolon. For Windows Authentication v2, specify the Windows NT password. **Note:** This property is not applicable if you use the Windows Authentication mode to access Microsoft SQL Server.
Host	Name of the machine hosting the database server. To connect to Microsoft Azure SQL Database, specify the fully qualified host name. For example, `vmjcmwxsfboheng.westus.cloudapp.azure.com`.
Port	Network port number used to connect to the database server. Default is 1433.
Instance Name	Instance name of the Microsoft SQL Server database.
Database Name	Database name for the Microsoft SQL Server target. Database name is case-sensitive if the database is case-sensitive. Maximum length is 100 characters. Database names can include alphanumeric and underscore characters.
Schema	Schema used for the target connection.
Code Page	The code page of the database server.
Encryption Method	The method that the Secure Agent uses to encrypt the data sent between the driver and the database server. You can use the encryption method to connect to Microsoft Azure SQL Database.
Crypto Protocol Version	Cryptographic protocols to use when you enable SSL encryption.
Validate Server Certificate	When set to True, Secure Agent validates the certificate that is sent by the database server. If you specify the HostNameInCertificate parameter, Secure Agent also validates the host name in the certificate. When set to false, the Secure Agent doesn't validate the certificate that is sent by the database server.
Trust Store	The location and name of the truststore file. The truststore file contains a list of Certificate Authorities (CAs) that the driver uses for SSL server authentication.
Trust Store Password	The password to access the contents of the truststore file.
Host Name in Certificate	Host name of the machine that hosts the secure database. If you specify a host name, the Secure Agent validates the host name included in the connection with the host name in the SSL certificate.

Property	Description
Metadata Advanced Connection Properties	Additional properties for the JDBC driver to fetch the metadata. If you specify more than one property, separate each key-value pair with a semicolon.
Runtime Advanced Connection Properties	Additional properties for the ODBC driver to run mappings. If you specify more than one property, separate each key-value pair with a semicolon.

MongoDB Mass Ingestion connection properties

When you set up a MongoDB Mass Ingestion connection, you must configure the connection properties.

The following table describes MongoDB Mass Ingestion connection properties:

Connection property	Description
Connection Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 255 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Description	Optional. A description of the connection. The description cannot exceed 4,000 characters.
Type	Type of connection. You must select **MongoDB Mass Ingestion**.
Runtime Environment	The name of the runtime environment where you want to run the tasks.
Host and Port	An SRV record or a comma-separated list of *host_name*:*port* pairs. **Note:** If you are using the MongoDB replica set mode, you can enter multiple host names for resilience. If one host is not available, another specified host will be used.
SRV	Select this check box if you specified an SRV record in **Host and Port** property.
User Name	User name for logging in to the database.
Password	Password for the specified database user.
Authentication Database	The name of the authentication database associated with the specified user.

Connection property	Description
Replica Set Name	The name of the replica set that is composed of the MongoDB servers with replicas of the source data. This field is relevant if you are using the MongoDB replica set mode.
Additional Connection Properties	One or more additional MongoDB connection string options that you want to use. Specify the properties as key-value pairs. If you specify more than one property, separate them with the ampersand symbol (&). The connection properties are case sensitive. Example: `authSource=admin&replicaSet=rsprimary` For more information about the MongoDB connection string options, refer to: https://www.mongodb.com/docs/v5.2/reference/connection-string/#connection-string-options

MQTT connection properties

When you set up an MQ Telemetry Transport (MQTT) connection, you must configure the connection properties.

The following table describes the MQTT connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
Description	Optional. Description that you can use to identity the connection. The description cannot exceed 4,000 characters.
Type	The MQTT connection type. If you do not see the connection type, go to the **Add-On Connectors** page to install the connector.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Broker URI	The connection URL of the MQTT broker. If specified, this value overrides the URL specified in the main portion of the URL. Sample URL: `tcp://<IP Address>:<port>`
Client Id	Client identifier of your MQTT client. If this value is left blank, the MQTT server assigns a unique value. This property value must be unique for each MQTT client connecting to a specific MQTT server. Sharing projects without changing the Client ID can lead to connection issues, including disconnections and missing updates.
Username	Username to use when connecting to the broker.
Password	Password to use when connecting to the broker.

Property	Description
Connection Timeout	Maximum time interval the client will wait for the connection to the MQTT server to be established. Default timeout is 30 seconds. A value of 0 disables timeout processing. That is, the client waits until the network connection is made successfully or fails.
Use SSL	Enable this option to use SSL for secure transmission. If you enable the SSL authentication, ensure to provide both keystore and truststore details for using the MQTT connection in a streaming ingestion task.
Keystore Filename	Contains the keys and certificates required for secure communication.
Keystore Password	Password for the keystore filename.
Keystore Type	Type of keystore to use. Keystore type defines the storage and data format of the keystore information and the algorithms used to protect private keys in the keystore. Use one of the following types: - JKS. Stores private keys and certificates. - PKCS12. Stores private keys, secret keys. and certificates.
Truststore Filename	File name of the truststore file.
Truststore Password	Password for the truststore file name.
Truststore Type	Type of truststore to use. Use one of the following types: - JKS - PKCS 12
TLS Protocol	Transport protocols to use. Use one of the following types: - SSL - SSLv3 - TLS - TLSv1 - TLSv1.1 - TLSv1.2

MySQL connection properties

When you set up a MySQL connection, configure the connection properties.

The following table describes the MySQL connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	Type of connection. Select MySQL from the list.
Runtime Environment	The name of the runtime environment where you want to run the tasks. **Note:** You cannot run a database ingestion task on a Hosted Agent or serverless runtime environment.
User Name	User name for the database login. The user name can't contain a semicolon.
Password	Password for the database login. The password can't contain a semicolon.
Host	Name of the machine that hosts the database server.
Port	Network port number used to connect to the database server. Default is 3306.
Database Name	Name of the MySQL database that you want to connect to. **Note:** The database name is case-sensitive. Maximum length is 64 characters. Database name can contain alphanumeric and underscore characters.
Code Page	The code page of the database server.
Metadata Advanced Connection Properties	Additional properties for the JDBC driver to fetch the metadata. If you specify more than one property, separate each key-value pair with a semicolon.
Runtime Advanced Connection Properties	Additional properties for the ODBC driver to run ingestion jobs. If you specify more than one property, separate each key-value pair with a semicolon.

Netezza connection properties

When you set up a Netezza connection, you must configure the connection properties.

The following table describes the Netezza connection properties:

Connection property	Description
Runtime Environment	The name of the runtime environment where you want to run the tasks.
Database	The name of the Netezza database.
Schemaname	The schema used for the Netezza source or target. Schema name is case sensitive.
Servername	The Netezza database host name.
Port	Network port number used to connect to the database server. Default is 1521.
Driver	The Netezza ODBC driver name, NetezzaSQL, used to connect to the Netezza database.
Runtime Additional Connection Configuration	Additional run-time attributes required to fetch data. For example, `securityLevel=preferredUnSecured;caCertFile =`
Metadata Additional Connection Configuration	The values to set the optional properties of the JDBC driver to fetch the metadata.
Username	Database user name with the appropriate read and write database permissions to access the database.
Password	Password for the database user name.

NetSuite Mass Ingestion connection properties

When you set up a NetSuite Mass Ingestion connection, you must configure the connection properties.

Note: Before you configure the connection properties, install the SuiteAnalytics Connect JDBC driver and copy the NQjc.jar file to the following directory: `<Secure_Agent>\ext\connectors\thirdparty\informatica.netsuiteami`

For more information about installing the SuiteAnalytics Connect JDBC driver, see the SuiteAnalytics Connect documentation.

The following table describes the connection properties for a NetSuite Mass Ingestion connection:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Email ID	User name of the NetSuite account. The user name is an email address.
Password	Password for the NetSuite account.
Service Host	Name of the SuiteAnalytics Connect Service host. The value in this field must match the value specified in the **Service Host** field under the **Your Configuration** section of the **SuiteAnalytics Connect Driver Download** page in NetSuite. To access the **SuiteAnalytics Connect Driver Download** page, log in to NetSuite and click the Set Up SuiteAnalytics Connect link in the Settings portlet.
Service Port	TCP/IP port on which the SuiteAnalytics Connect server is listening. Default is 1708.
Service Datasource	Data source that you want to use to access NetSuite data. You can select one of the following data sources: - NetSuite.com - NetSuite2.com Default is NetSuite2.com. **Note:** - In connections configured before the August 2022 release, the default value for this field is NetSuite.com. - To use a NetSuite2.com data source, the NetSuite user account must be configured with some specific roles and permissions. For more information about the roles and permissions required to access NetSuite2.com data sources, see the NetSuite documentation.
Account ID	NetSuite account ID. To find your account ID, log in to NetSuite and click **Setup** > **Integration** > **Web Services Preferences**. If you cannot access the **Setup** menu, navigate to **Support** > **Go to Suite Answers** > **Contact support by phone**. The page displays your account ID.
Role ID	Role ID associated with the NetSuite account.
Additional Connection Properties	Additional properties for the SuiteAnalytics Connect Driver that is used to connect to the NetSuite service data source. Specify the properties in `<property>=<value>` format. If you want to specify multiple properties, separate each property-value pair with a semicolon (;). You can specify the following connection properties in this field: - **ValidateServerCertificate**: Determines whether the driver validates the certificate sent by the SuiteAnalytics Connect server. During SSL server authentication, the SuiteAnalytics Connect server sends a certificate issued by a trusted Certificate Authority (CA). The required CAs are usually included in the Java truststore but you can also specify them using the TrustStore property. Valid values for the ValidateServerCertificate property are *true* and *false*. - **TrustStore**: Contains the path to a valid truststore containing the security certificates to be used for server authentication. The TrustStore property is ignored if the ValidateServerCertificate property is set to *false*. **Note:** For more information about the additional connection properties, see the NetSuite documentation.

OPC UA connection properties

When you set up an OPC UA connection, you must configure the connection properties.

The following table describes the OPC UA connection properties:

Property	Description
Connection Name	Name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:`~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
Description	Optional. Description of the connection. The description cannot exceed 4,000 characters.
Type	The OPC UA connection type.
Runtime Environment	Name of the runtime environment where you want to run the tasks.
Endpoint URL	The unique URL to connect to the OPC UA server. The endpoint URL identifies the specific instance of a server and a security policy type. A valid endpoint URL consists of the endpoint type (opc.tcp), the endpoint host name (IP address, URL, or DSN), and the endpoint port number. For example, `opc.tcp://opcuaserver.com:48010`
Security Policy	The security policy used to connect to the OPC UA server. The security policy parameters specify the security algorithms that the OPC UA server supports. You can choose one of the following security policies: - None. No security provided. - Basic128Rsa15 - Basic256 - Basic256Sha256 - Aes128_Sha256_RsaOaep - Aes256_Sha256_RsaPss **Note:** The OPC Foundation deprecated the security policies, Basic128Rsa15 and Basic256 as of OPC UA specification version 1.04. The encryption provided by these policies is less secure. Use these security policies only to provide backward compatibility.
Security Mode	The security mode used to connect to the OPC UA server. The security mode is valid only when security policy is not set to None. You can choose one of the following security policies: - Sign. Transfer unencrypted data, but with digital signatures that allow verification of data integrity. - SignAndEncrypt. Transfer signed and encrypted data.
Application URI	Optional. A unique identifier that the OPC UA application can use to connect to the OPC UA server. Enter a unique ID in the following format: `urn:aaa:bbb` For example, `urn:nifi:opcua` The unique identifier must match the URI of the Subject Alternative Name of your OPC UA client certificate.

Property	Description
Client Keystore Location	Optional. Absolute path and file name of the keystore file that contains private keys and certificates for the OPC UA server. Enter the path in the following format: `/root/opcua/client.jks` The keystore must contain only one keypair entry of private key and certificate. If multiple keypair entries exist, the first entry is used.
Client Keystore Password	Optional. Password for the client keystore.
Require server authentication	Optional. Enable if you require server authentication of client certificates, client authentication of server certificates, or both.
Trust store Location	Optional. The absolute path of the truststore file that contains the trusted certificate. Enter the path in the following format: `/root/opcua/trust.jks`
Trust store Password	Password for the truststore file.
Authentication Policy	Authentication settings required to establish the connections. You can choose one of the following authentication policies: - Anon. Anonymous authentication. Anonymous tokens are associated with servers that do not require user authentication. - UserName. User name and password tokens are associated with servers with any password based system, such as Windows.
User Name	User name to access the OPC UA server if you choose authentication policy as **UserName**.
Password	Password to access the OPC UA server if you choose authentication policy as **UserName**.

Oracle Database Ingestion connection properties

When you define an Oracle Database Ingestion connection for a database ingestion task, you must configure connection properties.

The following table describes the connection properties:

Property	Description
Connection Name	A name for the connection. This name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Spaces at the beginning or end of the name are trimmed and are not saved as part of the name. Maximum length is 100 characters. Connection names are not case sensitive.
Description	An optional description for the connection. Maximum length is 255 characters.

Property	Description
Type	The type of connection. For an Oracle Database Ingestion connection, the type must be **Oracle Database Ingestion**.
Runtime Environment	The name of the runtime environment where you want to run database ingestion tasks. You define runtime environments in Administrator.
User Name	User name for the Oracle database login. The user name cannot contain a semicolon.
Password	Password for the Oracle database login. The password cannot contain a semicolon.
Host	Host name of the database server.
Port	Network port number used to connect to the database server. Default is 1521.
Service Name	Service name or System ID (SID) that uniquely identifies the Oracle database. Specify the SID in the following format to connect to Oracle databases: `SID:<ORACLE_SID>`
Schema	Schema used for the Oracle connection.
Code Page	The code page of the database server. Database ingestion tasks use the UTF-8 code page. Default is UTF-8.
Encryption Method	For initial load jobs, determines whether the data exchanged between the Secure Agent and the Oracle database server is encrypted: Options are: - **SSL**. Establishes a secure connection using SSL for data encryption. If the Oracle database server cannot configure SSL, the connection fails. - **No Encryption**. Establishes a connection without using SSL. Data is not encrypted. Default is No Encryption.
Crypto Protocol Version	If you selected SSL as the encryption method, you must specify a cryptographic protocol or a list of cryptographic protocols supported by your server to use with an encrypted connection. Options are: - SSLv2 - SSLv3 - TLSv1.2 Default is TLSv1.2.

Property	Description
Validate Server Certificate	If you selected SSL as the encryption method, controls whether the Secure Agent validates the server certificate that is sent by the Oracle database server. - **True**. Validate the server certificate. - **False**. Do not validate the server certificate. Default is False. If you also specify the **Host Name in Certificate** property, the Secure Agent also validates the host name in the certificate.
Trust Store	If you selected SSL as the encryption method and enabled validation of the server certificate, specify the path and name of the truststore file, which contains the list of the Certificate Authorities (CAs) that the client trusts for SSL authentication.
Trust Store Password	If you selected SSL as the encryption method and enabled validation of the server certificate, specify a password for accessing the contents of the truststore file.
Host Name in Certificate	If you selected SSL as the encryption method and enabled validation of the server certificate, specify the host name of the machine that hosts the Oracle database to provide for additional security. The Secure Agent validates the host name included the connection with the host name in the SSL certificate.
Key Store	If you selected SSL as the encryption method and client authentication is enabled on the Oracle database server, specify the path and name of the keystore file. The keystore file contains the certificates that the client sends to the Oracle server in response to the server's certificate request.
Key Store Password	If you selected SSL as the encryption method and client authentication is enabled on the Oracle database server, specify the password for the keystore file.
Key Password	If you selected SSL as the encryption method and client authentication is enabled on the Oracle database server, specify the password for the keys in the keystore file. Use this property when the keys have a different password than the keystore file.
Database Connect String	An Oracle connection string, defined in TNS, that database ingestion tasks use to connect to the Oracle database.

Property	Description
TDE Wallet Directory	The path and file name for the Oracle wallet file that is used for Oracle Transparent Data Encryption (TDE). Specify this property value only if you capture change data from TDE-encrypted tablespaces and one of the following conditions are true: - The Oracle wallet is not available to the database. - The Oracle database is running on a server that is remote from Oracle redo logs. - The wallet directory is not in the default location on the database host or the wallet name is not the default name of ewallet.p12. - The wallet directory is not available to the Secure Agent host.
TDE Wallet Password	A clear text password that is required to access the Oracle TDE wallet and get the master key. This property value is required if you need to read and decrypt data from TDE-encrypted tablespaces in the Oracle source database.
Directory Substitution	A local path prefix to substitute for the server path prefix of the redo logs on the Oracle server. This substitute local path is required when the log reader runs on a system other than the Oracle server and uses a different mapping to access the redo log files. Use this property in the following situations: - The redo logs reside on shared disk. - The redo logs have been copied to a system other than the Oracle system. - The archived redo logs are accessed by using a different NFS mount. **Note:** Do not use this statement if you use Oracle Automatic Storage Management (ASM) to manage the redo logs. You can define one or more substitutions. Use the following format: `server_path_prefix,local_path_prefix;server_path_prefix,local_path_prefix;...`
Reader Active Log Mask	A mask that the log reader uses for selecting active redo logs when the Oracle database uses multiplexing of redo logs. The log reader compares the mask against the member names in an active redo log group to determine which log to read. In the mask, you can use the asterisk (*) wildcard to represent zero or more characters. The mask can be up to 128 characters in length. It is case-sensitive on Linux or UNIX systems but not on Windows systems.

Property	Description
Reader Archive Destination 1	The primary log destination from which the log reader reads archived logs, when Oracle is configured to write more than one copy of each archived redo log. Enter a number that corresponds to a *n* value in an Oracle LOG_ARCHIVE_DEST_*n* initialization parameter, where *n* is a value from 1 to 10. If you set only one of the Reader Archive Destination 1 and Destination 2 properties, the log reader uses that property setting. If you specify neither property, the archive log queries are not filtered by the log destination.
Reader Archive Destination 2	The secondary log destination from which the log reader reads archived logs when the primary destination becomes unavailable or when the logs at the primary destination cannot be read. For example, logs might have been corrupted or deleted. Enter a number that corresponds to the *n* value in an Oracle LOG_ARCHIVE_DEST_*n* initialization parameter, where *n* is a value from 1 to 10. Usually, this value is a number greater than 1.
Reader ASM Connect String	In an Oracle ASM environment, the Oracle connection string, defined in TNS, that the log reader uses to connect to the ASM instance that manages storage of active and archived redo logs for the source database.
Reader ASM User Name	In an Oracle ASM environment, an Oracle user ID that the log reader uses to connect to the ASM instance that manages storage of active and archived redo logs for the source database. This user ID must have SYSDBA or SYSASM authority. To use SYSASM authority, set the **Reader ASM Connect As SYSASM** property to Y.
Reader ASM Password	In an Oracle ASM environment, a clear text password for the user that is specified in the **Reader ASM User Name** property. The log reader uses this password and the ASM user name to connect to the ASM instance that manages storage of active and archived redo logs for the source database.
Reader ASM Connect As SYSASM	If you use Oracle 11*g* ASM or later and want the log reader to use a user ID that has SYSASM authority to connect to the ASM instance, select this check box. Also specify a user ID that has SYSASM authority in the **Reader ASM User Name** property. To use a user ID that has SYSDBA authority, clear this check box. By default, this check box is cleared.

Property	Description

Reader Mode	Indicates the source of and types of Oracle redo logs that the log reader reads. Valid options are:
	- **ACTIVE**. Read active and archived redo logs from the Oracle online system. Optionally, you can use the **Reader Active Log Mask** property to filter the active redo logs and use the **Reader Archive Destination 1** and **Reader Archive Destination 2** properties to limit the archived log destinations from which to read archived logs.
	- **ARCHIVEONLY**. Read only archived redo logs. Optionally, you can use the **Reader Archive Destination 1** and **Reader Archive Destination 2** properties to limit the archived log destinations from which to read archived logs.
	- **ARCHIVECOPY**. Read archived redo logs that have been copied to an alternate file system. Use this option in the following situations:
	  - You do not have the authority to access the Oracle archived redo logs directly.
	  - The archived redo logs are written to ASM, but you do not have access to ASM.
	  - The archived log retention policy for the database server causes the archived logs to not be retained long enough.
	With this option, the **Reader Archive Destination 1** and **Reader Archive Destination 2** properties are ignored.
	Default is **ACTIVE**.

Reader Standby Log Mask	A mask that the log reader uses for selecting redo logs for an Oracle physical standby database when the database uses multiplexing of redo logs. The log reader compares the mask against the member names in an redo log group to determine which log to read. In the mask, you can use the asterisk (*) wildcard to represent zero or more characters. The mask can be up to 128 characters in length. It is case-sensitive on Linux or UNIX systems but not on Windows systems.
Standby Connect String	An Oracle connection string, defined in TNS, that the log reader uses to connect to the Oracle physical standby database for change capture when the database is not open with read only access.
Standby User Name	A user ID that the log reader uses to connect to the Oracle physical standby database for change capture. This user ID must have SYSDBA authority.
Standby Password	A password that the log reader uses to connect to the Oracle physical standby database for change capture.

Property	Description
RAC Members	The maximum number of active redo log threads, or *members*, in an Oracle Real Application Cluster (RAC) that can be tracked. For a Data Guard physical standby database that supports a primary database in a RAC environment, this value is the number of active threads for the primary database. Valid values are 1 to 100. Default is 0, which causes an appropriate number of log threads to be determined automatically. If this value is not appropriate for your environment, set this property to a value greater than 0.
BFILE Access	Select this check box in the following circumstances: - You use BFILE access to redo logs in physical directories on the local Oracle server file system. BFILE access uses Oracle directory objects to remotely access the redo logs in the file system. This method is an alternative to other log access methods such as ASM or NFS mounts. - You have an Amazon Relational Database Service (RDS) for Oracle source. In this case, this option enables access to the redo logs of a cloud-based database instance deployed in RDS. By default, this check box is cleared.

Oracle Fusion Cloud Mass Ingestion connection properties

When you set up an Oracle Fusion Cloud Mass Ingestion connection, you must configure the connection properties.

Note: Oracle Fusion Cloud Mass Ingestion connections can access the data of only Enterprise Resource Planning (ERP) and Oracle Supply Chain and Manufacturing (SCM) modules of Oracle Fusion Cloud Applications Suite.

The following table describes the connection properties for an Oracle Fusion Cloud Mass Ingestion connection:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Authentication	Authentication method of the connection. By default, the connection uses the Basic authentication method.
User Name	User name of the Oracle Cloud account.
Password	Password for the Oracle Cloud account.
Server URL	URL of the Oracle Cloud service that you want to access.
API Version	Version of the Oracle Cloud REST API that you want to use for the connection.

PostgreSQL connection properties

When you set up a PostgreSQL connection, configure the connection properties.

The following table describes the PostgreSQL connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	Type of connection. Select PostgreSQL from the list.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run a database ingestion task on a Hosted Agent or in a serverless runtime environment.
Host Name	Host name of the PostgreSQL server to which you want to connect.
Port	Port number for the PostgreSQL server to which you want to connect. Default is 5432.
Schema	The schema name. If you don't specify the schema name, all the schemas available in the database are listed while importing the source object in Data Integration.
Database	The PostgreSQL database name.
User Name	User name to access the PostgreSQL database.
Password	Password for the PostgreSQL database user name.
Encryption Method	Determines whether the data exchanged between the Secure Agent and the PostgreSQL database server is encrypted. Select one of the following encryption methods: - noEncryption. Establishes a connection without using SSL. Data is not encrypted. - SSL. Establishes a connection using SSL. Data is encrypted using SSL. If the PostgreSQL database server can't configure SSL, the connection fails. - requestSSL. Attempts to establish a connection using SSL. If the PostgreSQL database server can't configure SSL, the Secure Agent establishes an unencrypted connection. Default is noEncryption.
Validate Server Certificate	Applicable if you select SSL or requestSSL as the encryption method. Select the Validate Server Certificate option so that the Secure Agent validates the server certificate that is sent by the PostgreSQL database server. If you specify the Host Name In Certificate property, the Secure Agent also validates the host name in the certificate.
TrustStore	Applicable if you select SSL or requestSSL as the encryption method and the Validate Server Certificate option. The path and name of the truststore file, which contains the list of the Certificate Authorities (CAs) that the PostgreSQL client trusts.

Property	Description
TrustStore Password	Applicable if you select SSL or requestSSL as the encryption method and the Validate Server Certificate option. The password to access the truststore file that contains the SSL certificate.
Host Name In Certificate	Optional when you select SSL or requestSSL as the encryption method and the Validate Server Certificate option. A host name for providing additional security. The Secure Agent validates the host name included in the connection with the host name in the SSL certificate.
KeyStore	Applicable if you select SSL as the encryption method and when client authentication is enabled on the PostgreSQL database server. The path and the file name of the key store. The keystore file contains the certificates that the PostgreSQL client sends to the PostgreSQL server in response to the server's certificate request.
KeyStore Password	Applicable if you select SSL as the encryption method and when client authentication is enabled on the PostgreSQL database server. The password for the keystore file required for secure communication.
Key Password	Applicable if you select SSL as the encryption method and when client authentication is enabled on the PostgreSQL database server. Required when individual keys in the keystore file have a different password than the keystore file.
Additional Connection Properties	Additional connection parameters that you want to use. Provide the connection parameters as semicolon-separated key-value pairs.
Crypto Protocol Versions	Required if you select SSL or requestSSL as the encryption method. A cryptographic protocol or a list of cryptographic protocols to use with an encrypted connection. You can select one of the following protocols: - SSLv3 - TLSv1_2 Default is TLSv1_2.

Salesforce Marketing Cloud connection properties

When you set up a Salesforce Marketing Cloud connection, configure the connection properties.

The following table describes the Salesforce Marketing Cloud connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Salesforce Marketing Cloud connection type.

Property	Description
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run an application ingestion task on a Hosted Agent or serverless runtime environment.
Salesforce Marketing Cloud Url	The URL that the agent uses to connect to the Salesforce Marketing Cloud WSDL.
Username	Applies to basic authentication. The user name of the Salesforce Marketing Cloud account. **Note:** This property is not applicable to connections configured for application ingestion tasks.
Password	Applies to basic authentication. The password for the Salesforce Marketing Cloud account. **Note:** This property is not applicable to connections configured for application ingestion tasks.
Client ID	The client ID of Salesforce Marketing Cloud required to generate a valid access token.
Client Secret	The client secret of Salesforce Marketing Cloud required to generate a valid access token.
Use Proxy Server	Connects to Salesforce Marketing Cloud though proxy. **Note:** This property is not applicable to connections configured for application ingestion tasks.
Enable Logging	Enables logging for the task. When you enable logging, you can view the session log for the log details. **Note:** This property is not applicable to connections configured for application ingestion tasks.
UTC offset	Uses the UTC offset connection property to read data from and write data to Salesforce Marketing Cloud in the UTC offset time zone. **Note:** This property is not applicable to connections configured for application ingestion tasks.
Batch Size	Number of rows that the agent writes in a batch to the target. When you insert or update data and specify the contact key, the data associated with the specified contact ID is inserted or updated in a batch to Salesforce Marketing Cloud. When you upsert data to Salesforce Marketing Cloud, do not specify the contact key. **Note:** This property is not applicable to connections configured for application ingestion tasks.
Enable Multiple BU	Uses the Salesforce Marketing Cloud connection to access data across all business units. Select this option if there are multiple business units in your Salesforce Marketing Cloud account. **Note:** This property is not applicable to connections configured for application ingestion tasks.

Salesforce Mass Ingestion connection properties

When you set up a Salesforce Mass Ingestion connection, you must configure the connection properties.

The Salesforce Mass Ingestion connection uses a connected app to access the Salesforce data. Before you configure the connection, you must configure a connected app in Salesforce to allow the connection to access the Salesforce data.
Note: For more information about configuring a connected app, see the Knowledge Base article 000172095.

The properties of a Salesforce Mass Ingestion connection vary based on the authentication method that you specify for the connection. When you create a connection, you can select one of the following authentication methods:

- **OAuth 2.0 Username-Password Flow**: Authenticates the connection by using the Salesforce account login credentials and the consumer key and consumer secret that Salesforce generates for the connected app.
- **OAuth 2.0 JWT Bearer Flow**: Authenticates the connection by using the Salesforce account user name, private key alias, private key password, and the consumer key that Salesforce generates for the connected app. Informatica recommends that you use this authentication method because this method provides secured access to Salesforce without sharing sensitive information, such as consumer secret and Salesforce account password.

Connection properties for OAuth 2.0 Username-Password Flow authentication

The following table describes the connection properties for a Salesforce Mass Ingestion connection configured with OAuth 2.0 Username-Password Flow authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
User Name	User name of the Salesforce account.
Password	Password for the Salesforce account.
Security Token	Security token associated with the Salesforce account. You can configure the connection without specifying the security token if there are no IP restrictions specified for the connected app. However, you must specify the security token if IP restrictions are enforced for the connected app and if the Secure Agent is not running on the trusted IP range specified for your Salesforce organization. **Note:** If you do not have the security token, reset the security token in Salesforce. For more information about resetting the security token, see the Salesforce documentation.
Consumer Key	Consumer key that Salesforce generates when you enable OAuth 2.0 authentication for the connected app.
Consumer Secret	Consumer secret that Salesforce generates when you enable OAuth 2.0 authentication for the connected app.

Connection property	Description
API Version	Version of the Salesforce API that you want to use to access the source data. Default is 51.0. **Note:** You cannot use a version older than 51.0.

OAuth token URL	OAuth 2.0 token endpoint of the Salesforce organization. The connected app sends access token requests to this endpoint.
	Default value is:
	`https://login.salesforce.com/services/oauth2/token`
	This default URL is used for all Salesforce instances.
	Alternatively, you can enter an instance-specific URL:
	`https://<instance domain URL>/services/oauth2/token`
	An instance-specific URL can establish a more direct and faster connection to the Salesforce host server. If the load on the common default endpoint is heavy and ingestion jobs fail with an authentication error when using it, use this alternative URL instead.

Note: For more information about the OAuth 2.0 Username-Password Flow authentication method, see the Salesforce documentation.

Connection properties for OAuth 2.0 JWT Bearer Flow authentication

The following table describes the connection properties for a Salesforce Mass Ingestion connection configured with OAuth 2.0 JWT Bearer Flow authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
User Name	User name of the Salesforce account.
Consumer Key	Consumer key that Salesforce generates when you enable OAuth 2.0 authentication for the connected app.
Keystore Path	Absolute path to the keystore file that contains the X509 certificate required to validate a JSON Web Token (JWT) and establish a secure connection with Salesforce. The keystore file must be in the Java KeyStore (JKS) format.
Keystore Password	Password for the keystore file.
Private Key Alias	Alias name of the private key used to sign the JWT.
Private Key Password	Password for the private key.

Connection property	Description
API Version	Version of the Salesforce API that you want to use to access the source data. Default is 51.0. **Note:** You cannot use a version older than 51.0.

OAuth token URL	OAuth 2.0 token endpoint of the Salesforce organization. The connected app sends access token requests to this endpoint.
	Default value is:
	`https://login.salesforce.com/services/oauth2/token`
	This default URL is used for all Salesforce instances.
	Alternatively, you can enter an instance-specific URL:
	`https://<instance domain URL>/services/oauth2/token`
	An instance-specific URL can establish a more direct and faster connection to the Salesforce host server. If the load on the common default endpoint is heavy and ingestion jobs fail with an authentication error when using it, use this alternative URL instead.

Note: For more information about the OAuth 2.0 JWT Bearer Flow authentication method, see the Salesforce documentation.

SAP HANA Database Ingestion connection properties

When you set up an SAP HANA connection, you must configure the connection properties.

The following table describes the SAP HANA connection properties:

Connection property	Description
Connection Name	A name for the connection. This name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + - Spaces at the beginning or end of the name are trimmed and are not saved as part of the name. Maximum length is 100 characters. Connection names are not case sensitive.
Description	An optional description for the connection. Maximum length is 255 characters.
Type	The type of connection. Ensure that the type is **SAP HANA Database Ingestion**.
Runtime Environment	The name of the runtime environment where you want to run database ingestion tasks. You define runtime environments in Administrator.
User Name	The user name to use for connecting to the SAP HANA instance.
Password	The password to use for connecting to the SAP HANA instance.
Host	The name of the machine that hosts the SAP HANA database server.
Port	The port number for the SAP HANA server to which you want to connect. Default is 30015.
Database Name	The SAP HANA source database name.

Connection property	Description
Advanced Connection Properties	Advanced properties for the SAP HANA JDBC driver, which is used to connect to the SAP HANA source. If you specify more than one *property=value* entry, separate them with an ampersand (&). The JDBC connection properties that you can enter in this field are described in the SAP JDBC Connection Properties documentation. For example: encrypt=true.
Log Clear	Required for incremental loads. The time interval, in days, after which the PKLOG table entries and shadow _CDC table entries are purged. The purging occurs only while an incremental load job is running. Valid values for a database ingestion job are 0 to 366. Any positive value in this range cause automatic housekeeping to run while the incremental job is running. Default is 14. A value of 0 means that the table entries are not purged. For manual housekeeping, enter 0 and use your in-house process. Any value outside the range of 0 to 366, including a negative number or non-numeric value, causes database ingestion jobs that use the connection to fail with the following error: `LogClear contains a non numeric number. Caused by: LogClear contains a non numeric number.`
Trigger Prefix	Adds a prefix to the names of the AFTER DELETE, AFTER INSERT, and AFTER UPDATE triggers that the CDC script generates for each source table to get before images and after images of the DML changes. Enter any prefix value up to 16 characters in length. An underscore (_) follows the prefix in the trigger name, for example, **TX_**SAP_DEMO_TABLE_DBMI_USER_t_d. You can use the prefix to comply with your site's trigger naming conventions.

Note: If you test the connection and the test fails, check that the SAP HANA JDBC driver file, ngdbc.jar, has been installed at *Secure Agent installation directory*>/ext/connectors/thirdparty/informatica.hanami.

SAP Mass Ingestion connection properties

When you set up a SAP Mass Ingestion connection, you must configure the connection properties.

The following table describes the connection properties for a SAP Mass Ingestion connection:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
User Name	User name of the SAP instance.
Password	Password for the SAP instance.
Language Code	Language code that corresponds to the SAP language.
System Number	System number of the SAP server.
Client Number	Client number of the SAP server.
Port Range	HTTP port range to run the Netty server.

Connection property	Description
Connection Type	Type of connection to access the ABAP application server. Options are: - **Direct Connection**: Accesses a single ABAP application server using the server host. - **Load Balancing Connection**: Accesses a group of ABAP application servers through the message server.
Application Server	Name of the SAP application server host. **Note:** This field appears only for the Direct Connection type.
Message Server	IP address or name of the SAP message server. **Note:** This field appears only for the Load Balancing Connection type.
SAP Logon Group	Name of the group of servers that belong to the SAP system you want to access. **Note:** This field appears only for the Load Balancing Connection type.
SAP System ID	ID of the SAP system that you want to access. **Note:** This field appears only for the Load Balancing Connection type.
Message Server Port	Port number on which the SAP message server is listening. **Note:** This field appears only for the Load Balancing Connection type.

SAP ODP Extractor connection properties

Select the **SAP ODP Extractor** connection type and configure the connection properties.

The following table describes the SAP ODP Extractor connection properties:

Connection property	Description
Runtime Environment	Runtime environment that contains the Secure Agent that you want to use to access SAP S/4HANA or SAP ECC.

SAP Server Connection Type	The SAP server connection type to use.
	Select from the following options:
	- **Application Server Connection**. Connect to an SAP Application Server using the SAP user name and password.
	- **Application Server SNC Connection**. Connect to an SAP Application Server using the secured network connection:
	  - With X.509 Certificate. You do not need to specify the SAP user name and password explicitly. You must provide the path of the x.509 certificate file.
	  - Without X.509 Certificate. You must provide the SAP user name.
	- **Load Balancing Server Connection**. Connect to an SAP Application Server with the least load at run time.
	- **Load Balancing Server SNC Connection**. Connect to an SAP Application Server using SNC with the least load at run time.
	Note: Before you use an SNC connection, you must verify that SNC is configured both on the SAP Server and the machine where the Secure Agent runs.

The following table describes the properties that must configure when you select **Application Server Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Application Server	The host name of the SAP Application Server.
SAP System Number	The system number of the SAP Server to connect.
SAP Username	The SAP user name with the appropriate user authorization.
SAP Password	The SAP password.
Subscriber Name	A name which defines the Secure Agent as a unique subscriber in the SAP system. SAP uses this name to define unique operational delta queue (ODQ) in case of delta read from ODP.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

The following table describes the properties that must configure when you select **Load Balancing Server Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Message Server	Host name of the SAP Message Server.
SAP System ID	The system ID of the SAP Message Server.
SAP Group	The login group name, for example, PUBLIC.
SAP Username	The SAP user name with the appropriate user authorization.
SAP Password	The SAP password.

Connection property	Description
Subscriber Name	A name which defines the Secure Agent as a unique subscriber in the SAP system. SAP uses this name to define unique operational delta queue (ODQ) in case of delta read from ODP.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

The following table describes the properties that must configure when you select **Application Server SNC Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Application Server	The host name of the SAP Application Server.
SAP System Number	The system number of the SAP Server to connect.
SNC My Name	Optional. The Informatica client Personal Security Environment (PSE) or certificate name. Default length is 256.
SNC Partner Name	The Informatica client PSE or certificate name. Default length is 256.
SNC Quality of Protection (QoP)	Specifies the SAP PSE or certificate name. You can select from the following options: - 1 - Apply authentication only. - 2 - Apply integrity protection (authentication). - 3 - Apply privacy protection (integrity and authentication). - 8 - Apply the default protection. - 9 - Apply the maximum protection. Default is *3 - Apply privacy protection (integrity and authentication)*.
SAP Cryptographic Library Path	The path to the cryptographic library. Specify `sapcrypto.dll` for Windows or `libsapcrypto.so` for Linux.
Use X509 Certificate	Specifies the quality of protection. Select to use X509 Certificate based SNC connection.

Connection property	Description
X509 Certificate Path or SAP Username	The path to the X509 certificate file. If you select to use the X509 certificate, specify the path to the X509 certificate file with .crt extension. You do not need to specify the SAP user name and password. If you do not want to use the X509 certificate, specify the SAP user name for which SNC is configured in SAP Server.
Subscriber Name	A name which defines the Informatica Secure Agent as a unique subscriber in the SAP system. SAP uses this name to define unique operational delta queue (ODQ) when the Secure Agent reads delta data from ODP.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

The following table describes the properties that must configure when you select **Load Balancing Server SNC Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Message Server	Host name of the SAP Message Server.
SAP System ID	The system ID of the SAP Message Server.
SAP Group	The login group name, for example, PUBLIC.
SNC My Name	Optional. The Informatica client PSE or certificate name generated on the Secure Agent machine. Default length is 256.
SNC Partner Name	The Informatica client PSE or certificate name generated on the SAP Server. Default length is 256.

Connection property	Description
SNC Quality of Protection (QoP)	Specifies the SAP PSE or certificate name. You can select from the following options: - 1 - Apply authentication only. - 2 - Apply integrity protection (authentication). - 3 - Apply privacy protection (integrity and authentication). - 8 - Apply the default protection. - 9 - Apply the maximum protection. Default is *3 - Apply privacy protection (integrity and authentication)*.
SAP Cryptographic Library Path	The path to the cryptographic library. Specify `sapcrypto.dll` for Windows or `libsapcrypto.so` for Linux.
Use X509 Certificate	Specifies the quality of protection. Select to use X509 Certificate based SNC connection.
X509 Certificate Path or SAP Username	The path to the X509 certificate file. If you select to use the X509 certificate, specify the path to the X509 certificate file with .crt extension. You do not need to specify the SAP user name and password. If you do not want to use the X509 certificate, specify the SAP user name for which SNC is configured in SAP Server.
Subscriber Name	A name which defines the Informatica Secure Agent as a unique subscriber in the SAP system. SAP uses this name to define unique operational delta queue (ODQ) when the Secure Agent reads delta data from ODP.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

ServiceNow Mass Ingestion connection properties

When you set up a ServiceNow Mass Ingestion connection, you must configure the connection properties.

The properties of a ServiceNow Mass Ingestion connection vary based on the authentication method that you specify for the connection. When you create a connection, you can select one of the following authentication methods:

- **OAuth 2.0**: Authenticates the connection by using the details of the OAuth API endpoint that is created for the connection in ServiceNow. To use this method, you must create OAuth API endpoint in ServiceNow and then specify the client ID and client secret of the API endpoint in the connection properties. For more information about creating an OAuth API endpoint in ServiceNow, see the ServiceNow documentation.
- **Basic**: Authenticates the connection by validating the login credentials of the ServiceNow account.

Connection properties for OAuth 2.0 authentication

The following table describes the connection properties for a ServiceNow Mass Ingestion connection configured with OAuth 2.0 authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
User Name	User name of the ServiceNow account.
Password	Password for the ServiceNow account.
Client Secret	Client secret of the API endpoint created for the connection in ServiceNow.
Client ID	Client ID of the API endpoint created for the connection in ServiceNow.
Base URI	URL of the ServiceNow instance. You must enter the base URI in the following format: `https://{your_servicenow_instance}.service-now.com/`
OAuth Token URL	OAuth token endpoint of the ServiceNow instance. The API client associated with the connection sends the access token requests to this endpoint.

Connection properties for Basic authentication

The following table describes the connection properties for a ServiceNow Mass Ingestion connection configured with Basic authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
User Name	User name of the ServiceNow account.
Password	Password for the ServiceNow account.
Base URI	URL of the ServiceNow instance. You must enter the base URI in the following format: `https://{your_servicenow_instance}.service-now.com/`

Snowflake Data Cloud connection properties

When you set up a Snowflake Data Cloud connection, configure the connection properties.

You can use the following authentication methods to connect to Snowflake:

- Standard. Uses Snowflake account user name and password credentials to connect to Snowflake.

  Note: For application ingestion tasks, you can use only the Standard authentication method.
- Authorization Code. Uses the OAuth 2.0 protocol with Authorization Code grant type to connect to Snowflake. Authorization Code allows authorized access to Snowflake without sharing or storing your login credentials.
- KeyPair. Uses the private key file and private key file password, along with the existing Snowflake account user name to connect to Snowflake.

You create a Snowflake Data Cloud connection on the Connections page. You can then use the connection when you read from or write data to Snowflake.

Standard authentication

When you set up a Snowflake Data Cloud connection, configure the connection properties.

The following table describes the Snowflake Data Cloud connection properties for the Standard authentication mode:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Snowflake Data Cloud connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run application ingestion tasks and database ingestion tasks on a Hosted Agent or serverless runtime environment.
Authentication	The authentication method that the connector must use to log in to Snowflake. Select **Standard**. Default is **Standard**.
Username	The user name to connect to the Snowflake account.
Password	The password to connect to the Snowflake account.
Account	The name of the Snowflake account. For example, if the Snowflake URL is https://<123abc>.us-east-2.aws.snowflakecomputing.com/console/login#/, your account name is the first segment in the URL before snowflakecomputing.com. Here, 123abc.us-east-2.aws is your account name. If you use the Snowsight URL, for example, https://app.snowflake.com/us-east-2.aws/<123abc>/dashboard, your account name is 123abc.us-east-2.aws **Note:** Ensure that the account name doesn't contain underscores. To use an alias name, contact Snowflake Customer Support.

Property	Description
Warehouse	The Snowflake warehouse name.
Role	The Snowflake role assigned to the user.

Additional JDBC URL Parameters	The additional JDBC connection parameters.
	Enter one or more JDBC connection parameters in the following format:
	`<param1>=<value>&<param2>=<value>&<param3>=<value>....`
	For example:
	`user=jon&warehouse=mywh&db=mydb&schema=public`
	Important: Ensure that there is no space before and after the equal sign (=) when you add the parameters.

OAuth 2.0 authorization code authentication

The following table describes the Snowflake Data Cloud connection properties for an OAuth 2.0 - AuthorizationCode type connection:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Snowflake Data Cloud connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run application ingestion tasks and database ingestion tasks on a Hosted Agent or serverless runtime environment.
Authentication	The authentication method that Snowflake Data Cloud Connector must use to log in to Snowflake. Select **AuthorizationCode**.
Account	The name of the Snowflake account. For example, if the Snowflake URL is https://<123abc>.us-east-2.aws.snowflakecomputing.com/console/login#/, your account name is the first segment in the URL before snowflakecomputing.com. Here, 123abc.us-east-2.aws is your account name. If you use the Snowsight URL, for example, https://app.snowflake.com/us-east-2.aws/<123abc>/dashboard, your account name is 123abc.us-east-2.aws **Note:** Ensure that the account name doesn't contain underscores. To use an alias name, contact Snowflake Customer Support.
Warehouse	The Snowflake warehouse name.

Property	Description

Additional JDBC URL Parameters	The additional JDBC connection parameters.
	Enter one or more JDBC connection parameters in the following format:
	`<param1>=<value>&<param2>=<value>&<param3>=<value>....`
	For example:
	`user=jon&warehouse=mywh&db=mydb&schema=public`
	Important: Ensure that there is no space before and after the equal sign (=) when you add the parameters.

Authorization URL	The Snowflake server endpoint that is used to authorize the user request. The authorization URL is https://<account name>.snowflakecomputing.com/oauth/authorize, where <account name> specifies the full name of your account provided by Snowflake. For example, https://<abc>.snowflakecomputing.com/oauth/authorize **Note:** If the account name contains underscores, use the alias name. You can also use the Authorization Code grant type that supports the authorization server in a Virtual Private Cloud network.
Access Token URL	The Snowflake access token endpoint that is used to exchange the authorization code for an access token. The access token URL is https://<account name>.snowflakecomputing.com/oauth/token-request, where <account name> specifies the full name of your account provided by Snowflake. For example, https://<abc>.snowflakecomputing.com/oauth/token-request **Note:** If the account name contains underscores, use the alias name.
Client ID	Client ID of your application that Snowflake provides during the registration process.
Client Secret	Client secret of your application.
Scope	Determines the access control if the API endpoint has defined custom scopes. Enter space-separated scope attributes. For example, specify `session:role:CQA_GCP` as the scope to override the value of the default user role. The value must be one of the roles assigned in Security Integration.
Access Token Parameters	Additional parameters to use with the access token URL. Define the parameters in the JSON format. For example, define the following parameters: `[{"Name":"code_verifier","Value":"5PMddu6Zcg6Tc4sbg"}]`

Authorization Code Parameters	Additional parameters to use with the authorization token URL.
	Define the parameters in the JSON format.
	For example, define the following parameters:
	`[{"Name":"code_challenge","Value":"Ikr-vv52th0UeVRi4"},`
	`{"Name":"code_challenge_method","Value":"S256"}]`

Access Token	The access token value. Enter the populated access token value, or click **Generate Token** to populate the access token value.

Property	Description
Generate Token	Generates the access token and refresh token based on the OAuth attributes you specified.
Refresh Token	The refresh token value. Enter the populated refresh token value, or click **Generate Token** to populate the refresh token value. If the access token is not valid or expires, the agent fetches a new access token with the help of the refresh token. **Note:** If the refresh token expires, provide a valid refresh token or regenerate a new refresh token by clicking **Generate Token**.

Key pair authentication

The following table describes the Snowflake Data Cloud connection properties for the KeyPair authentication type connection:

Connection property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	The Snowflake Data Cloud connection type.
Runtime Environment	The name of the runtime environment where you want to run the tasks. You cannot run application ingestion tasks and database ingestion tasks on a Hosted Agent or serverless runtime environment.
Authentication	The authentication method to log in to Snowflake. Select **KeyPair**.
Username	The user name to connect to the Snowflake account.
Account	The name of the Snowflake account. For example, if the Snowflake URL is https://<123abc>.us-east-2.aws.snowflakecomputing.com/console/login#/, your account name is the first segment in the URL before snowflakecomputing.com. Here, 123abc.us-east-2.aws is your account name. If you use the Snowsight URL, for example, https://app.snowflake.com/us-east-2.aws/<123abc>/dashboard, your account name is 123abc.us-east-2.aws. **Note:** Ensure that the account name doesn't contain underscores. To use an alias name, contact Snowflake Customer Support.
Warehouse	The Snowflake warehouse name.

Connection property	Description

Additional JDBC URL Parameters	Optional. The additional JDBC connection parameters.
	Enter one or more JDBC connection parameters in the following format:
	`<param1>=<value>&<param2>=<value>&<param3>=<value>....`
	For example:
	`user=jon&warehouse=mywh&db=mydb&schema=public`
	Important: Ensure that there is no space before and after the equal sign (=) when you add the parameters.

Private Key File	Path to the private key file, including the private key file name, that the Secure Agent uses to access Snowflake. **Note:** Verify that the keystore is FIPS-certified.
Private Key Password	Password for the private key file.

REST V2 connection properties

When you set up a REST V2 connection, you must configure the connection properties.

The following table describes the REST V2 connection properties for a standard authentication type connection:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the tasks. Specify a Secure Agent, Hosted Agent, or serverless runtime environment. You cannot run a streaming ingestion task on a Hosted Agent or serverless runtime environment.
Authentication Type	If required, select the authentication method that the connector must use to login to the web service application. Default is none.
Auth User ID	The user name to login to the web service application when you select the Basic authentication. Digest authentication is not applicable.
Auth Password	The password associated with the user name when you select the Basic authentication. Digest authentication is not applicable.
OAuth Consumer Key	The client key associated with the web service application. Required only for OAuth authentication type.
OAuth Consumer Secret	The client password to connect to the web service application. Required only for OAuth authentication type.
OAuth Token	The access token to connect to the web service application. Required only for OAuth authentication type.

Connection property	Description
OAuth Token Secret	The password associated with the OAuth token. Required only for OAuth authentication type.
Swagger File Path	The absolute path along with the file name or the hosted URL of the swagger specification file. The hosted URL must return the content of the file without prompting for further authentication and redirection. If you provide the absolute path of the swagger specification file, the swagger specification file must be located on the machine that hosts the Secure Agent. The user must have the read permission for the folder and the specification file. Example: `C:\swagger\sampleSwagger.json` **Note:** In a streaming ingestion task, use only a hosted URL of the swagger specification file as the swagger file path.
TrustStore File Path	The absolute path of the truststore file that contains the TLS certificate to establish a one-way or two-way secure connection with the REST API. Specify a directory path that is available on each Secure Agent machine in the runtime environment. You can also configure the truststore file name and password as a JVM option or import the certificate to the following directory: `<Secure Agent installation directory\jre\lib\security\cacerts.` For the serverless runtime environment, specify the truststore file path in the serverless agent directory. For example: /home/cldagnt/SystemAgent/serverless/configurations/ssl_store/<cert_name>.jks
TrustStore Password	The password for the truststore file that contains the SSL certificate. You can also configure the truststore password as a JVM option.
KeyStore File Name	The absolute path of the keystore file that contains the keys and certificates required to establish a two-way secure communication with the REST API. Specify a directory path that is available on each Secure Agent machine in the runtime environment. You can also configure the keystore file name and location as a JVM option or import the certificate to any directory. For the serverless runtime environment, specify the keystore file path in the serverless agent directory. For example: /home/cldagnt/SystemAgent/serverless/configurations/ssl_store/<cert_name>.jks
KeyStore Password	The password for the keystore file required for secure communication. You can also configure the keystore password as a JVM option.
Proxy Type	Type of proxy. You can select one of the following options: - No Proxy. Bypasses the proxy server configured at the agent or the connection level. - Platform Proxy. Proxy configured at the agent level is considered. - Custom Proxy. Proxy configured at the connection level is considered.

Connection property	Description
Proxy Configuration	The proxy configuration format: `<host>:<port>` You cannot configure an authenticated proxy server.
Advanced Fields	Enter the arguments that the Secure Agent uses when connecting to a REST endpoint. You can specify the following arguments, each separated by a semicolon (`;`): `ConnectionTimeout`. The wait time in milliseconds to get a response from a REST endpoint. The connection ends after the connection timeout is over. Default is the timeout defined in the endpoint API. **Note:** If you define both the REST V2 connection timeout and the endpoint API timeout, the connection ends at the shortest defined timeout. `connectiondelaytime`. The delay time in milliseconds to send a request to a REST endpoint. Default is 10000. `retryattempts`. Number of times the connection is attempted when 400 and 500 series error codes are returned in the response. Default is 3. Specify 0 to disable the retry attempts. `qualifiedSchema`. Specifies if the schema selected is qualified or unqualified. Default is false. Example: `connectiondelaytime:10000;retryattempts:5` **Note:** In a streaming ingestion task, only `ConnectionTimeout` and `retryattempts` are applicable.

Teradata connection properties

When you set up a Teradata connection, you must configure the connection properties.

The following table describes the Teradata connection properties:

Connection property	Description
Connection Name	Name of the connection.
Description	Description of the connection.
Type	The type of connection. Select Teradata.
Runtime Environment	The name of the run-time environment where you want to run the tasks. You cannot use the Hosted Agent for Teradata Connector.
TDPID	The name or IP address of the Teradata database machine.
Tenacity	Amount of time, in hours, that Teradata PT API continues trying to log on when the maximum number of operations runs on the Teradata database. Specify a positive integer. Default is 4.
Database Name	The Teradata database name. If you do not enter a database name, Teradata PT API uses the default login database name.

Connection property	Description
Code Page	Code page associated with the Teradata database. Select one the following code pages: - MS Windows Latin 1. Select for ISO 8859-1 Western European data. - UTF-8. Select for Unicode and non-Unicode data. When you run a task that extracts data from a Teradata source, the code page of the Teradata PT API connection must be the same as the code page of the Teradata source.
Max Sessions	Maximum number of sessions that Teradata PT API establishes with the Teradata database. Specify a positive, non-zero integer. Default is 4.
Min Sessions	Minimum number of Teradata PT API sessions required for the Teradata PT API job to continue. Specify a positive integer between 1 and the Max Sessions value. Default is 1.
Sleep	Amount of time, in minutes, that Teradata PT API pauses before it retries to log on when the maximum number of operations runs on the Teradata database. Specify a positive, non-zero integer. Default is 6.
Data Encryption	Enables full security encryption of SQL requests, responses, and data. Default is disabled.
Block Size	Maximum block size, in bytes. Teradata PT API uses this property to read the data block size from source through the Export operator. Maximum is 16775168 bytes for Teradata Database version 16.20 and higher. If the Teradata Database version is lower than 16.20, then Teradata scales down the block size from 16775168 bytes to the maximum allowed value. The block size 16775168 is not allowed in the Spool mode. For more information, see Teradata logs and verify the Teradata documentation of the same version.
Authentication Type	Method to authenticate the user. Select one of the following authentication types: - Native. Authenticates your user name and password against the Teradata database specified in the connection. - LDAP. Authenticates user credentials against the external LDAP directory service. - KRB5. Authenticates to the Teradata database through Kerberos. Default is Native.
Kerberos Artifacts Directory	Directory that contains the Kerberos configuration files named `krb5.conf` and `IICSTPT.keytab`. Applicable when you select KRB5 as the authentication type.
Metadata Advanced Connection Properties	The values to set the optional properties of the JDBC driver to fetch the metadata. For example, tmode=ANSI.
Enable Metadata Qualification	Select this option to enable the Teradata connection to read reserved words used as table or column names from the Teradata database. By default, the Enable Metadata Qualification checkbox is not selected and the Secure Agent does not read reserved words from Teradata.

Connection property	Description
User Name	Database user name with the appropriate read and write database permissions to access the database. If you select KRB5 as the authentication type, you must specify the Kerberos user name.
Password	Password for the database user name. If you select KRB5 as the authentication type, you do not need to specify the Kerberos user password.

Workday Mass Ingestion connection properties

When you set up a Workday Mass Ingestion connection, you must configure the connection properties.

The properties of a Workday Mass Ingestion connection vary based on the authentication method that you specify for the connection. When you create a connection, you can select one of the following authentication methods:

- **Basic**: Authenticates the connection by validating the login credentials of the Workday account.
- **OAuth 2.0 Refresh Token Flow**: Authenticates the connection by using an application that is registered in Workday. To use this method, you must register an application in Workday and then specify the client ID and client secret of the application in the connection properties. For more information about registering an application in Workday, see the Workday documentation.

Connection properties for Basic authentication

The following table describes the connection properties for a Workday Mass Ingestion connection configured with Basic authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Domain Name	Name of the Workday domain that contains the resources that you want to access.
Tenant Name	Identifier of the Workday tenant that you want to access.
Version	Web Service Description Language (WSDL) version for the endpoints that the connection must use to retrieve Workday data. The list of operations supported for a web service depends on the WSDL version that you specify in this field. **Note:** Informatica recommends that you use WSDL v37.0 because Workday Mass Ingestion connections might not read data from the services that are not part of WSDL v37.0. For more information on the WSDL versions, see the Workday Web Services (WWS) documentation.
User Name	User name of the Workday account.
Password	Password for the Workday account.

Note: If you configure a connection with the Basic authentication method and then test the connection, the test is always successful even if the connection property values that you specified are incorrect. Therefore, ensure that you specify correct values for the connection properties before you save the connection.

Connection properties for OAuth 2.0 Refresh Token Flow authentication

The following table describes the connection properties for a Workday Mass Ingestion connection configured with OAuth 2.0 Refresh Token Flow authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Domain Name	Name of the Workday domain that contains the resources that you want to access.
Tenant Name	Identifier of the Workday tenant that you want to access.
Version	Web Service Description Language (WSDL) version for the endpoints that the connection must use to retrieve Workday data. The list of operations supported for a web service depends on the WSDL version that you specify in this field. **Note:** Informatica recommends that you use WSDL v37.0 because Workday Mass Ingestion connections might not read data from the services that are not part of WSDL v37.0. For more information on the WSDL versions, see the Workday Web Services (WWS) documentation.
User Name	User name of the Workday account.
Client ID	Client ID of the application registered in Workday.
Client Secret	Private key of the application registered in Workday.
Refresh Token	Refresh token string that Workday generates for the registered application.
Token Endpoint	OAuth token endpoint of the Workday instance. The registered application sends the access token requests to this endpoint.

Zendesk Mass Ingestion connection properties

When you set up a Zendesk Mass Ingestion connection, you must configure the connection properties.

The properties of a Zendesk Mass Ingestion connection vary based on the authentication method that you specify for the connection. When you create a connection, you can select one of the following authentication methods:

- **Basic**: Authenticates the connection by using the login credentials and subdomain associated with the Zendesk account. The Basic authentication method does not use any encrypted access token to connect to the data source, which results in quick and easy access to Zendesk data.

  Note: You can use the Basic authentication method only if your Zendesk account is not configured with two-factor authentication. If the account is configured with two-factor authentication, you must use the OAuth 2.0 authentication method for the connection.

- **OAuth 2.0**: Authenticates the connection by using an application that is registered in Zendesk along with the login credentials and subdomain associated with the Zendesk account. To use this method, you must register an application in Zendesk and then specify the client ID and client secret of the application in the connection properties. For more information about registering an application in Zendesk, see the Zendesk documentation.

Connection properties for Basic authentication

The following table describes the connection properties for a Zendesk Mass Ingestion connection configured with Basic authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Email ID	User name of the Zendesk account. The user name is an email address.
Password	Password for the Zendesk account.
Subdomain	URL of the Zendesk help center that you want to access.

Note: For more information about the Basic authentication method, see the Zendesk documentation.

Connection properties for OAuth 2.0 authentication

The following table describes the connection properties for a Zendesk Mass Ingestion connection configured with OAuth 2.0 authentication:

Connection property	Description
Runtime Environment	Name of the runtime environment where you want to run the ingestion tasks. You must specify a Secure Agent as the runtime environment. **Note:** You cannot run application ingestion tasks on a Hosted Agent or serverless runtime environment.
Email ID	User name of the Zendesk account. The user name is an email address.
Password	Password for the Zendesk account.
Subdomain	URL of the Zendesk help center that you want the connection to access.
Client ID	Client ID of the application registered in Zendesk.
Client Secret	Client secret of the application registered in Zendesk.
Grant Type	OAuth 2.0 grant type to be used by the connection. By default, Zendesk Mass Ingestion connections are configured to use the password grant type to exchange user names and passwords for access tokens.

Note: For more information about the OAuth 2.0 authentication method, see the Zendesk documentation.

INDEX