Cloudera Flow Management 4.12.0

Integrations

Date published: 2019-06-26
Date modified: 2026-03-31

https://docs.cloudera.com/

Legal Notice

Contents

Integrations

This section provides information on integrating NiFi and NiFi Registry with other components.

Integrating NiFi and Atlas

You can integrate NiFi with Apache Atlas to take advantage of robust dataset and application lineage support.

Manually integrating with Atlas when Auto-TLS is not enabled

If Cloudera Flow Management or the Cloudera Base on premises cluster does not have Auto-TLS enabled and you want to Atlas, then you must manually integrate with Atlas by creating the `ReportLineageToAtlas` reporting task.

About this task

Perform this task if:

- Cloudera Flow Management does not have TLS enabled; AND
- The Cloudera Base on premises cluster does not have auto-TLS enabled; AND
- You do not want to enable auto-TLS; AND
- You want Atlas as part of Cloudera Flow Management on your Cloudera Base on premises deployment.

Procedure

1. From the Global Menu located in NiFi's upper right corner, select Controller Services and click the Reporting Tasks tab.
2. Click the Add (+) icon to launch the Add Reporting Task dialog.
3. Select `ReportLineageToAtlas` and click Add.
4. Click the Edit icon to launch the Configure Reporting Task dialog. The following properties are required:
    - Atlas URLs – a comma-separated list of Atlas Server URLs. Once you have started reporting, you cannot modify an existing Reporting Task to add a new Atlas Server. When you need to add a new Atlas Server, you must create a new reporting task.
    - Atlas Configuration Directory - This specifies where the atlas-applications.properties is created.

      The directory must:

        - Be located and accessible/writable by the user running the NiFi process.
        - Be available on each NiFi node.
        - Pre-exist. It will not be created by the reporting task.
        - Not be in the /tmp directory.
    - Create Atlas Configuration File – Set to True. When set to True, the atlas-application-properties file and the Atlas Configuration Directory are automatically created when the Reporting Task starts.
    - Lineage Strategy – Specifies the level of granularity for your NiFi dataflow reporting to Atlas. Once you have started reporting, you should not switch between simple and complete lineage reporting strategies.
    - Provenance Record Start Position – Specifies where in the Provenance Events stream the Reporting Task should start.
    - Provenance Record Batch Size – Specifies how many records you want to send in a single batch
    - NiFi URL for Atlas – Specifies the NiFi cluster URL.
    - Atlas Authentication Method – Specifies how to authenticate the Reporting Task to the Atlas Server. Basic authentication is the default.

- Kafka Security Protocol – Specifies the protocol used to communicate with Kafka brokers to send Atlas hook notification messages. This value should match Kafka's `security.protocol` property value.

Manually integrating with Atlas when Auto-TLS is enabled

You must perform some manual steps to integrate with Atlas when auto-TLS is enabled on your Cloudera Base on premises cluster.

About this task

You must perform these steps if:

- You want CFM to integrate with Atlas; AND
- The Cloudera Base on premises cluster has auto-TLS enabled

Procedure

1. Select the Atlas integration checkbox.
2. Restart NiFi.
3. Click Create required NiFi object in the Cloudera Manager Actions menu.

Integrating NiFi and NiFi Registry with Knox

Integrate NiFi and NiFi Registry with Knox to securely access NiFi and NiFi Registry nodes.

Apache Knox Gateway (Knox) provides the following benefits:

- Centralized access to all services in the cluster.
- Authentication with single sign-on.
- Service-level authorization to the cluster.
- Does not expose the service endpoints such as URLs, ports, IP addresses.

When you integrate NiFi and NiFi Registry with Knox, you can use the Knox URL as a single entry point to securely access all NiFi nodes and switch nodes if one fails.

For information more information on Knox, see *Apache Knox Overview*.

For information on how to select Knox during the NiFi and NiFi Registry installation, see *Cloudera Flow Management Deployment*.

Related Information

Apache Knox Overview
Cloudera Flow Management Deployment

Customizing properties in Cloudera Manager

You can customize NiFi and NiFi Registry beyond what the customization page in Cloudera Manager allows. To make any changes, use the dot notation to represent the actual schema for a given property file.

About this task

The following steps show how to enhance or overwrite xml based properties in Cloudera Manager using dot notation.

Procedure

Use the following structure:

```
xml.<properties-type>.<entity>.<identifier>.class
xml.<properties-type>.<entity>.<identifier>.property.<property-value>
```

Where:

- <properties-type> for NiFi can be authorizers and loginIdentityProviders
- <properties-type> for NiFi Registry can be authorizers and identityProviders.

The following property key/value example creates a user group provider entry into the authorizers file for NiFi:

```
Name: xml.authorizers.userGroupProvider.file-user-group-provider.class
Value: org.apache.nifi.authorization.FileUserGroupProvider

Name: xml.authorizers.userGroupProvider.file-user-group-provider.property
.Initial User Identity 2
Value: CN=localhost, OU=NIFI
```

This translates to the following entry in the generated authorizers.xml file:

```
<authorizers>
…...
    <userGroupProvider>
        <identifier>file-user-group-provider</identifier>
        <class>org.apache.nifi.authorization.FileUserGroupProvider</class>
        <property name="Initial User Identity 2">CN=localhost, OU=NIFI</prop
erty>
    </userGroupProvider>
…
...
</authorizers>
```

Properties names that have spaces are supported and do not need to be escaped.

Example

For an example, see *Pairing LDAP with a Composite Group Provider*.