Hortonworks DataFlow

MiNiFi Quick Start

(Feb 24, 2017)

docs.hortonworks.com

Hortonworks DataFlow: MiNiFi Quick Start

Hortonworks DataFlow (HDF) is powered by Apache NiFi. A version of this documentation originally appeared on the Apache NiFi website.

HDF is the first integrated platform that solves the real time challenges of collecting and transporting data from a multitude of sources and provides interactive command and control of live flows with full and automated data provenance. HDF is a single combined platform that provides the data acquisition, simple event processing, transport and delivery mechanism designed to accommodate the diverse dataflows generated by a world of connected people, systems and things.

Unlike other providers of platforms built using Apache Hadoop, Hortonworks contributes 100% of our code back to the Apache Software Foundation. Hortonworks DataFlow is Apache-licensed and completely open source. We sell only expert technical support, training and partner-enablement services. All of our technology is, and will remain free and open source.

Please visit the Hortonworks page for more information on Hortonworks technology. For more information on Hortonworks services, please visit either the Support or Training page. Feel free to Contact Us directly to discuss your specific needs.

Table of Contents

1. MiNiFi Java Agent Quick Start

This guide is intended to to help you install and start using MiNiFi Java Agent quickly. For additional details, see the Administration Guide.

1.1. Overview

Apache NiFi, MiNiFi is an Apache NiFi project, designed to collect data at its source. MiNiFi was developed with the following objectives in mind:

- Small and lightweight footprint
- Central agent management
- Data provenance generation
- NiFi integration for follow-on dataflow management and chain of custody information

1.2. Before You Begin

MiNiFi is supported on the following operating systems:

- Red Hat Enterprise Linux / CentOS 6 (64-bit)
- Red Hat Enterprise Linux / CentOS 7 (64-bit)
- Ubuntu Precise (12.04) (64-bit)
- Ubuntu Trusty (14.04) (64-bit)
- Debian 7
- SUSE Linux Enterprise Server (SLES) 11 SP3 (64-bit)

You can find download links for the following MiNiFi software in the HDF Release Notes.

- MiNiFi Java Agent
- MiNiFi C++
- MiNiFi Toolkit

1.3. Installing and Starting MiNiFi

You have several options for installing and starting MiNiFi.

1.3.1. Installing MiNiFi

To install MiNiFi:

1. Download MiNiFi.
2. Extract the file to the location from which you want to run the application.

1.3.2. Installing MiNiFi as a Service

You can also install MiNiFi as a service:

1. Navigate to the MiNiFi installation directory.
2. Enter:

```
bin/minifi.sh install
```

You can also specify a custom name for your MiNiFi installation, by specifying that name during your install command. For example, to install MiNiFi as a service and named dataflow, enter:

```
bin/minifi.sh install dataflow
```

1.3.3. Starting MiNiFi

Once you have downloaded and installed MiNiFi, you need to start MiNiFi

You can start NiFi in the foreground, background, or as a service.

Launching MiNiFi in the foreground:

1. From a terminal window, navigate to the MiNiFi installation directory.
2. Enter:

```
bin/minifi.sh run
```

Launching MiNiFi in the background:

1. From a terminal window, navigate to the MiNiFi installation directory.
2. Enter:

```
bin/minifi.sh start
```

Launching MiNiFi as a service:

1. From a terminal window, enter:

```
sudo service minifi start
```

1.4. Working with Dataflows

When you are working with a MiNiFi dataflow, you should design it, add any additional configuration your environment or use case requires, and then deploy your dataflow. MiNiFi is not designed to accommodate substantial mid-dataflow configuration.

1.4.1. Setting up Your Dataflow

Before you begin, you should be aware that the following NiFi components are not supported in MiNiFi dataflows:

- Funnels
- Multiple source relationships for a single connection
- Process groups

Additionally, each processor requires a distinct name.

You can use the MiNiFi Toolkit, located in your MiNiFi installation directory, and any NiFi instance to set up the dataflow you want MiNiFi to run:

1. Launch NiFi
2. Create a dataflow.
3. Convert your dataflow into a template.
4. Download your template as an `.xml` file.

   For more information on working with templates, see the Templates section in the *User Guide*.
5. From the MiNiFi Toolkit, run the following command to turn your `.xml` file into a `.yml` file:

   ```
   config.sh transform input_file output_file
   ```
6. Move your new `.yml` file to `minifi/conf`.
7. Rename your `.yml` file `config.yml`.

Note

You can use one template at a time, per MiNiFi instance.

Result: Once you have your `config.yml` file in the `minifi/conf` directory, launch that instance of MiNiFi and your dataflow begins automatically.

1.4.2. Using Processors Not Packaged with MiNiFi

MiNiFi is able to use the following processors out of the box:

- UpdateAttribute

- AttributesToJSON
- Base64EncodeContent
- CompressContent
- ControlRate
- ConvertCharacterSet
- ConvertJSONToSQL
- DetectDuplicate
- DistributeLoad
- DuplicateFlowFile
- EncryptContent
- EvaluateJsonPath
- EvaluateRegularExpression
- EvaluateXPath
- EvaluateXQuery
- ExecuteProcess
- ExecuteSQL
- ExecuteStreamCommand
- ExtractText
- FetchDistributedMapCache
- FetchFile
- FetchSFTP
- GenerateFlowFile
- GetFTP
- GetFile
- GetHTTP
- GetJMSQueue
- GetJMSTopic
- GetSFTP
- HandleHttpRequest

- HandleHttpResponse
- HashAttribute
- HashContent
- IdentifyMimeType
- InvokeHTTP
- ListFile
- ListSFTP
- ListenHTTP
- ListenRELP
- ListenSyslog
- ListenTCP
- ListenUDP
- LogAttribute
- MergeContent
- ModifyBytes
- MonitorActivity
- ParseSyslog
- PostHTTP
- PutDistributedMapCache
- PutEmail
- PutFTP
- PutFile
- PutJMS
- PutSFTP
- PutSQL
- PutSyslog
- QueryDatabaseTable
- ReplaceText
- ReplaceTextWithMapping

- RouteOnAttribute
- RouteOnContent
- RouteText
- ScanAttribute
- ScanContent
- SegmentContent
- SplitContent
- SplitJson
- SplitText
- SplitXml
- TailFile
- TransformXml
- UnpackContent
- ValidateXml

If you want to create a dataflow with a processor not shipped with MiNiFi, you can do so.

1. Set up your dataflow as described above.
2. Copy the desired NAR file into the MiNiFi lib directory.
3. Restart your MiNiFi instance.

Note

Currently only the StandardSSLContextService is supported as a controller service. It is created automatically if the the "Security Properties" section is set and can be referenced in the processor configuration using the ID "SSL-Context-Service".

1.5. Securing your Dataflow

You can secure your MiNiFi dataflow using keystore or trust store SSL protocols, however, this information is not automatically generated. You will need to generate your security configuration information yourself.

To run a MiNiFi dataflow securely, modify the Security Properties section of your `config.yml` file.

1. Create your dataflow template as discussed above.

2. Move it to minifi.conf and rename config.yml.

3. Manually modify the Security Properties section of `config.yml`.

```
Security Properties:
keystore:
keystore type:
keystore password:
key password:
truststore:
truststore type:
truststore password:
ssl protocol: TLS
Sensitive Props:
key:
algorithm: PBEWITHMD5AND256BITAES-CBC-OPENSSL
provider: BC
```

1.6. Managing MiNiFi

You can also perform some management tasks using MiNiFi

1.6.1. Monitoring Status

You can use the `minifi.sh` flowStatus option to monitor a range of aspects of your MiNiFi operational and dataflow status. You can use the flowStatus option to get information dataflow component health and functionality, a MiNiFi instance, or system diagnostics.

FlowStatus accepts the following flags and options:

- processors
  - health
  - bulletins
  - status
- connections
  - health
  - stats
- remoteProcessGroups
  - health
  - bulletins
  - status
  - authorizationIssues

    - inputPorts
- controllerServices
    - health
    - bulletins
- provenancereporting
    - health
    - bulletins
- instance
    - health
    - bulletins
    - status
- systemdiagnostics
    - heap
    - processorstats
    - contentrepositoryusage
    - flowfilerepositoryusage
    - garbagecollection

For example, this query gets the health, stats, and bulletins for the TailFile processors

```
minifi.sh flowStatus processor:TailFile:health,stats,bulletins
```

Note

Currently the script only accepts one high level option at a time.

Any connections, remote process groups or processors names that contain ":", ";" or "," will cause parsing errors when querying.

For details on the flowStatus option, see the FlowStatus Query Option section of the *Administration Guide*.

1.6.2. Loading a New Dataflow

You can load a new dataflow for a MiNiFi instance to run:

1. Create a new `config.yml` file with the new dataflow.
2. Replace the existing `config.yml` in minifi/conf with the new file.

3. Restart MiNiFi.

1.6.3. Stopping MiNiFi

You can stop MiNiFi at any time.

Stopping MiNiFi:

1. From a terminal window, navigate to the MiNiFi installation directory.
2. Enter:

```
bin/minifi.sh stop
```

Stopping MiNiFi as a service:

1. From a terminal window, enter:

```
sudo service minifi stop
```