Cloudera Flow Management 2.1.5

# Managing NiFi in Cloudera Manager

**Date published: 2019-06-26**
**Date modified: 2023-11-02**

## CLOUDERA

# Legal Notice

# Contents

# Using the Actions menu

You can manage the NiFi service in Cloudera Manager on CDP Private Cloud Base using the Actions menu available on the service page. This menu provides a list of commands relevant to the role type you are viewing. These commands typically include stopping, starting, or restarting the instance, and may include additional commands depending on your role.

## About this task

Follow these steps to access the Actions menu.

## Procedure

1. From Cloudera Manager, click the Clusters tab in the left-hand navigation.
2. Click NiFi in the list of services to display the NiFi service page.
3. Click the Actions drop-down next to the service name.

   The following commands are available:

   **Start**

   > This action starts the NiFi service.

   **Restart**

   > This action runs two other actions, 'Stop NiFi' and 'Start NiFi', one after another.

   **Rolling Restart**

   > This action consists of three actions, and it runs a specific restart of the NiFi service. The Rolling restart action first runs 'Stop NiFi', then it runs 'Start NiFi', and finally it runs 'NifiNodeWaitUntilHealthy'. The goal of the last action is to wait until the NiFi service is started and works correctly. It is needed in the rolling restart because the 'Start NiFi' action just runs the start NiFi service command and does not wait until the NiFi service actually starts and also does not check if the service works correctly.

   **Migrate File-based Authorizations to Ranger**

   > This action migrates policies from authorizations.xml to Ranger. For more information, see *Migrating NiFi File-Based authorization to Ranger*.

   **Create required NiFi objects**

   > This action calls the CreateRequiredNiFiObjects.sh script, which is used to create NiFi objects like: ReportLineageToAtlas, StandardRestrictedSSLContextService, or MonitorDiskUsage. This script also adds component policies to Ranger, which are described in the ranger-component-policies.xml file. The action is automatically executed after the first NiFi start. If needed, you can rerun it later manually.

   **Remove Ranger repository**

   > This action removes the related NiFi repository/service from Ranger.

   > ⚠️ **Important:** Use this action with caution as it causes NiFi to fail. If you ran the command accidentally, 1. stop NiFi, 2. create a marker file, and 3. start NiFi again. After completing these actions, Ranger NiFi repository/service, all related policies, and all NiFi objects will be created/added.

   **Reset File-based Authorizer Users and Policies**

   > This action automatically removes the users.xml and authorizations.xml configuration files on each host with NiFi. It also creates backups of these two configurations in the same directory, but with

the file names changed (users.xml{current-date}, authorizations.xml{current-date}). This action helps you to regenerate users.xml and authorizations.xml config files on all NiFi hosts.

**Create Marker File**

This action creates the /var/lib/nifi/marker.txt file on a random NiFi host. You need this file to understand exactly which NiFi instance should create a NiFi Ranger repository/service and basic policies during the NiFi start. Without this file, Ranger repository/service is not created and NiFi service fails. The action runs before 'Start NiFI' only during the first NiFi start (after installation).

**Create First Run File**

This action creates the /var/lib/nifi/first_run.txt file on all NiFi hosts. You can use this file to check whether it is the first start of NiFi. For example during the NiFi start process (controller.sh), it is checked that it is not the first start of NiFi and groups are updated for some policies in the NiFi Ranger repository/service, if needed. After the /var/lib/nifi/first_run.txt file is created, it is removed automatically.

**Run Post Upgrade Changes**

This action runs the run_post_upgrade_changes.sh script. This script adds the required changes for a new CFM version after an upgrade. It also automatically runs during the first run after an upgrade, if the CFM version has changed.

> ⚠️ **Important:** Do not execute this action manually, only in specific cases when you need to fix things in the scope of run_post_upgrade_changes.sh.

**Wait until the NiFi Node is healthy**

This action runs the 'nifi_wait_until_healthy' script, and the system waits until the NiFi service is started and works correctly.

**Stop**

This action stops the NiFi service.

**Related Information**

Migrating NiFi File-Based authorization to Ranger