Upgrade 3

Ambari Managed HDF Upgrade

Date of Publish: 2018-08-13

http://docs.hortonworks.com

Contents

Pre-upgrade tasks

Review credentials

If you are using the NiFi Certificate Authority (CA), you must perform as series of activities to ensure that your certificate credentials are correctly propagated after upgrade.

About this task

> **Important:** Performing these steps requires regeneration of keystores and truststores. If you have added additional keystore or truststore certificates, you must manually re-add these certificates after you force regenerate certificates.

.

Procedure

1. Ensure that your **admin token** is greater than or equal to 16 characters. If it is less, you can reset it with a value of 16 or more characters.
    a. In the Ambari UI, select the **NiFi configs** tab and search for the **nifi.toolkit.tls.token** in the **Advanced nifi-ambari-ssl-config** section.
    b. Enter a new token value with 16 or more characters.
    c. Enable **NiFi CA Force Regeneration** to enforce creating a new certificate.
    d. Save your configuration.
2. If you are using key, keystore, and truststore passwords that are auto-generated rather than stored in Ambari, you must provide a 16 character password in the Ambari UI to ensure credentials are not lost during upgrade. From the **NiFi configs** tab, specify values for the following fields and save your configuration.
    - Set the Keystore password in **nifi.security.keystorePasswd**.
    - Set the Key password in **nifi.security.keyPasswd**.

      Note: This value must match the Keystore password.
    - Set the Truststore password in **nifi.security.truststorePasswd**.
3. If you are using a secured NiFi Registry with the NiFi CA and auto-generated key, keystore, and truststore passwords, you must provide a 16 character in the Ambari UI. From the **NiFi Registry configs** tab, specify values for the following fields and save your configuration.
    - Set the Keystore password in **nifi.security.keystorePasswd**.
    - Set the Key password in **nifi.security.keyPasswd**.

      Note: This value must match the Keystore password.
    - Set the Truststore password in **nifi.security.truststorePasswd**.
4. Regenerate your certificates by restarting NiFi and NiFi Registry.
5. Deselect **NiFi CA Force Regeneration** and save this configuration.

Stop Services

You must stop Ambari Metrics, Log Search, and the Amber Server and Agents before you upgrade. If you are upgrading HDF services on an HDP cluster, you must stop NiFi and NiFi Registry before upgrading to Ambari 2.7.0, HDP 3.0.0, and HDF 3.2.0. When you are installing HDF services on an HDP cluster, manual installation is required post-Ambari upgrade.

Procedure

1. If you are running Ambari Metrics in your cluster, stop the service and put it in Maintenance Mode.

   From Ambari Web, browse to Services > Ambari Metrics and select Stop from the Service Actions menu. Then, select Turn on Maintenance Mode from the Service Actions menu.
2. If you are running Log Search in your cluster, stop the service.

   From Ambari Web, browse to Services > Log Search and select Stop from the Service Actions menu. Then, select Turn on Maintenance Mode from the Service Actions menu.
3. If you have HDF services installed on an HDP cluster, you must stop them before upgrade.

   - If you have services from an HDF 3.1.x cluster, stop NiFi and NiFi Registry and confirm that you do not have any of the service processes running.
   - If you have services from HDF 3.0.x stop NiFi, Schema Registry, and SAM, and confirm that you do not have any of the service processes running.
4. Stop the Ambari Server. On the host running Ambari Server:

   ```
   ambari-server stop
   ```
5. Stop all Ambari Agents. On each host in your cluster running an Ambari Agent:

   ```
   ambari-agent stop
   ```

Verify NiFi Toolkit Version

Ambari 2.7.0 verifies your NiFi Toolkit versions, and requires that you have one that matches the version of NiFi and NiFi Registry that you have installed. Before you begin the upgrade process, verify that you have the necessary version of the NiFi Toolkit.

Procedure

1. Identify the version of NiFi and NiFi Registry that you have installed.

   ```

   run hdf-select status | nifi
   run hdf-select status | nifi-registry
   ```

   This returns the stack version of NiFi and NiFi Registry. For example, if you have HDF 3.1.x installed, hdf-select returns something similar to:

   ```
   nifi - 3.1.1.0-35
   nifi-registry - 3.1.1.0-35
   ```
2. Ensure that var/lib/ambari-agent/tmp contains a version of the NiFi Toolkit that contains this stack version. For example, based on the above returns, you should have a NiFi Toolkit directory named nifi-toolkit-1.5.0.3.1.1.0-35.
3. If you do not have the required directory, copy an existing NiFi Toolkit directory and rename it using version and build number appropriate for your installation.
4. Ensure that the service user is the owner of the copied directory.

Upgrade Ambari and the HDF Management Pack

The first step in upgrading to HDF 3.2.0 is to upgrade to Ambari 2.7.0 and to upgrade to the HDF 3.2.0 management pack.

Preparing to Upgrade Ambari

- Be sure to review the Ambari 2.7.0.0 release notes for Known Issues and Behavioral Changes.
- You must have root, administrative, or root-equivalent authorization on the Ambari Server host and all Ambari Agent hosts in the cluster.
- You must backup the Ambari Server database.
- You must make a safe copy of the Ambari Server configuration file found at /etc/ambari-server/conf/ambari.properties.
- If your cluster is SSO-enabled, do not stop Knox before upgrading Ambari.

The following table lists recommended (✓), and unsupported (X) upgrade paths.

From / To	Ambari 2.7.x	Ambari 2.6.x
Ambari 2.6.x	✓	X
Ambari 2.5x	X	✓
Ambari 2.4x	X	✓

During Ambari upgrade, the existing /var/lib/ambari-server/ambari-env.sh file is overwritten and a backup copy of ambari-env.sh (with extension .rpmsave) is created. If you have manually modified ambari-env.sh (for example, to change Ambari Server heap), you will need to re-apply your changes to the new file.

Get the Ambari Repository

The first step in upgrading Ambari is obtaining the public repositories.

Before you begin

Check your current directory before you download the new repository file to make sure that there are no previous versions of the ambari.repo file. If you do not, and a previous version exists, the new download is saved with a numeric extension, such as ambari.repo.1. Make sure that the version you copy is the new version.

Procedure

1. Get the new Ambari repo and replace the old repository file with the new repository file on all hosts in your cluster.

   Select the repository appropriate for your environment:

   - For RHEL/CentOS/Oracle Linux 7:

     ```
     wget -nv http://public-repo-1.hortonworks.com/ambari/centos7/2.x/
     updates/2.7.0.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
     ```

- For Amazon Linux 2:

  ```
  wget -nv http://public-repo-1.hortonworks.com/ambari/amazonlinux2/2.x/
  updates/2.7.0.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
  ```
- For SLES 12:

  ```
  wget -nv http://public-repo-1.hortonworks.com/ambari/sles12/2.x/
  updates/2.7.0.0/ambari.repo -O /etc/zypp/repos.d/ambari.repo
  ```
- For Ubuntu 14:

  ```
  wget -nv http://public-repo-1.hortonworks.com/ambari/ubuntu14/2.x/
  updates/2.7.0.0/ambari.list -O /etc/apt/sources.list.d/ambari.list
  ```
- For Ubuntu 16:

  ```
  wget -nv http://public-repo-1.hortonworks.com/ambari/ubuntu16/2.x/
  updates/2.7.0.0/ambari.list -O /etc/apt/sources.list.d/ambari.list
  ```
- For Debian 9:

  ```
  wget -nv http://public-repo-1.hortonworks.com/ambari/debian9/2.x/
  updates/2.7.0.0/ambari.list -O /etc/apt/sources.list.d/ambari.list
  ```
- For RHEL/CENTOS/Oracle Linux 7 running on IBM Power Systems:

  ```
  wget -nv http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/
  updates/2.7.0.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
  ```

  Note:

  If your cluster does not have access to the Internet, set up a local repository with this data before you continue. See *Using a Local Repository* in the *Apache Ambari Installation* for more information.

  Note:

  Ambari Server does not automatically turn off iptables. Check that your installation setup does not depend on iptables being disabled. After upgrading the server, you must either disable iptables manually or make sure that you have appropriate ports available on all cluster hosts. For information on disabling your iptables, see *Configuring iptables* in the *Apache Ambari Installation*.

2. If you are upgrading an HDF-only cluster, open the ambari.repo or ambari.list file, depending on your operating system. Remove the following line:

   ```
   #json.url = http://public-repo-1.hortonworks.com/HDP/hdp_urlinfo.json
   ```

   Note: This is not necessary if you are upgrading HDF services on an HDP cluster.

Related Information

Using a Local Repository

Configuring iptables

Upgrade Ambari Server

Procedure

1. Upgrade Ambari Server. On the host running Ambari Server:

- For RHEL/CentOS/Oracle Linux:

  ```
  yum clean all
  yum info ambari-server
  ```

  In the info output, visually validate that there is an available version containing "2.6"

  ```
  yum upgrade ambari-server
  ```
- For SLES:

  ```
  zypper clean
  zypper info ambari-server
  ```

  In the info output, visually validate that there is an available version containing "2.6"

  ```
  zypper up ambari-server
  ```

  > **Important:**
  >
  > When performing upgrade on SLES, you will see a message "There is an update candidate for 'ambari-server', but it is from different vendor. Use 'zypper install ambari-server-2.6.1-143.noarch' to install this candidate". You will need to use yast to update the package, as follows:
  >
  > **a.** Display the command line UI for YaST, by entering:
  >
  > ```
  > > yast
  > ```
  >
  > **b.** Choose **Software > Software Management**, then click the **Enter** button.
  >
  > **c.** In the **Search Phrase** field, enter ambari-server, then click the **Enter** button.
  >
  > **d.** On the right side you will see the search result ambari-server 2.6. Click **Actions**, choose **Update**, then click the **Enter** button.
  >
  > **e.** Go to **Accept**, and click **enter**.
- For Ubuntu/Debian:

  ```
  apt-get clean all
  apt-get update
  apt-cache show ambari-server | grep Version
  ```

  In the info output, visually validate that there is an available version containing "2.6"

  ```
  apt-get install ambari-server
  ```

2. Check for upgrade success by noting progress during the Ambari Server installation process.

   As the process runs, the console displays output similar, although not identical, to the following:

   Setting up Upgrade Process Resolving Dependencies --> Running transaction check

   If the upgrade fails, the console displays output similar to the following:

   Setting up Upgrade Process No Packages marked for Update

   A successful upgrade displays output similar to the following:

   Updated: ambari-server.noarch 0:2.6.1-143 Complete!

Upgrade the Ambari Agents

Procedure

1. Upgrade all Ambari Agents. On each host in your cluster running an Ambari Agent:

    - For RHEL/CentOS/Oracle Linux:

        ```
        yum upgrade ambari-agent
        ```

    - For SLES:

        ```
        zypper up ambari-agent
        ```

        Note:

        Ignore the warning that begins with "There are some running programs that use files deleted by recent upgrade".

        Important:

        When performing upgrade on SLES, you will see a message "There is an update candidate for 'ambari-agent', but it is from different vendor. Use 'zypper install ambari-agent-2.6-143.noarch' to install this candidate". You will need to use yast to update the package, as follows:

        a. Display the command line UI for YaST by entering:

        ```
        > yast
        ```

        b. Choose **Software** > **Software Management**, then click **Enter**.

        c. In the **Search Phrase** field, enter **ambari-agent**, then click the **Enter** button.

        d. On the right side you will see the search result ambari-agent 2.6. Click **Actions**, choose **Update**, then click the **Enter** button.

        e. Go to **Accept**, and click **enter**.

    - For Ubuntu/Debian:

        ```
        apt-get update
        apt-get install ambari-agent
        ```

2. After the upgrade process completes, check each host to make sure the new files have been installed:

For RHEL/CentOS/Oracle Linux 7:	`rpm -qa | grep ambari-agent`
For SLES 12:	`rpm -qa | grep ambari-agent`
For Ubuntu 14:	`dpkg -l ambari-agent`
For Ubuntu 16:	`dpkg -l ambari-agent`
For Debian 9:	`dpkg -l ambari-agent`

Upgrade the HDF Management Pack

A management pack bundles service definitions, stack definitions, and stack add-on service definitions so they do not need to be included with the Ambari core functionality and can be updated in between major releases. Upgrade the management pack to ensure that you have the latest versions of the available components.

Before you begin

Get the HDF Management Pack location and build number from the *HDF Release Notes*.

Procedure

1. Back up your Ambari resources folder:

```
cp -r /var/lib/ambari-server/resources /var/lib/ambari-server/
resources.backup
```

2. Upgrade the HDF management pack with the command appropriate for your operating system:

RHEL/CentOS/Oracle Linux 7:

```
ambari-server upgrade-mpack \
--mpack=http://public-repo-1.hortonworks.com/HDF/centos7/3.x/updates/
<version>/tars/hdf_ambari_mp/hdf-ambari-mpack-<version>-<build-
number>.tar.gz \
--verbose
```

SUSE Linux Enterprise Server (SLES) v12 SP1

```
ambari-server upgrade-mpack \
--mpack=http://public-repo-1.hortonworks.com/HDF/sles12/3.x/updates/
<version>/tars/hdf_ambari_mp/hdf-ambari-mpack-<version>-<build-
number>.tar.gz \
--verbose
```

Debian 9:

```
ambari-server upgrade-mpack \
--mpack=http://public-repo-1.hortonworks.com/HDF/debian9/3.x/updates/
<version>/tars/hdf_ambari_mp/hdf-ambari-mpack-<version>-<build-
number>.tar.gz \
--verbose
```

Ubuntu 14:

```
ambari-server upgrade-mpack \
--mpack=http://public-repo-1.hortonworks.com/HDF/ubuntu14/3.x/updates/
<version>/tars/hdf_ambari_mp/hdf-ambari-mpack-<version>-<build-
number>.tar.gz \
--verbose
```

Ubuntu 16:

```
ambari-server upgrade-mpack \
--mpack=http://public-repo-1.hortonworks.com/HDF/ubuntu16/3.x/updates/
<version>/tars/hdf_ambari_mp/hdf-ambari-mpack-<version>-<build-
number>.tar.gz \
--verbose
```

Related Information

HDF Release Notes

Upgrade the Ambari Database Schema

Procedure

1. Upgrade Ambari Server database schema. On the host running Ambari Server:

   ```
   ambari-server upgrade
   ```

2. Confirm there is only one ambari-server*.jar file in /usr/lib/ambari-server. If there is more than one JAR file with name ambari-server*.jar, move all JARs except ambari-server-2.7.0.0.jar to /tmp before proceeding with upgrade.
3. Start the Ambari Server. On the host running Ambari Server:

   ```
   ambari-server start
   ```

4. Start all Ambari Agents. On each host in your cluster running an Ambari Agent:

   ```
   ambari-agent start
   ```

5. Open Ambari Web.

   Point your browser to http://<your.ambari.server>:8080

   where <your.ambari.server> is the name of your ambari server host. For example, c6401.ambari.apache.org.

   > **Important:**
   >
   > Refresh your browser so that it loads the new version of the Ambari Web code. If you have problems, clear your browser cache manually, then restart Ambari Server.

Confirm the Ambari upgrade for HDF services

Procedure

1. If NiFi Registry is installed, go to Ambari UI and NiFi Registry configurations. Confirm that the configuration version has changed (due to newly added properties). Search for nifi.registry.db.password and if blank set to nifireg. Save Configuration. Do not start NiFi Registry if previously stopped (such as in HDF on HDP).
2. If NiFi is installed, go to the Ambari UI and NiFi configurations. Confirm that the configuration version has changed (due to newly added properties). If the **Save** button is enabled click **Save**. Do not start NiFi if previously stopped (such as in HDF on HDP).
3. If you are upgrading HDF 3.0.2 services on an HDP cluster, and Schema Registry is installed, go to the Ambari UI and Schema Registry configurations. Confirm that the configuration version has changed (due to newly added properties).

   Confirm existing database settings and make any corrections needed. On Streamline Config tab go to the "Setup Database and Database User" toggle and hit the "Set Recommendations" button to ensure default value of toggle false is set (do not enable toggle).

   If Save button is enabled click Save, add comments. Do not start Schema Registry.
4. If you are upgrading HDF 3.0.2 services on an HDP cluster and SAM is installed, go to the Ambari UI and Streamline configurations. Confirm that the configuration version has changed (due to newly added properties).

   Confirm existing database settings and make any corrections needed. On Streamline Config tab go to the "Setup Database and Database User" toggle and hit the "Set Recommendations" button to ensure default value of toggle false is set (do not enable toggle).

   If **Save** button is enabled click Save, add comments. Do not start SAM.

Upgrade HDF

Depending on your existing installation you must now upgrade your HDF-only cluster, your HDF 3.0.x cluster, or your HDF 3.1.x cluster. Perform one of the following workflows.

Upgrading an HDF Cluster

Prerequisites

To perform an HDF upgrade using Ambari, your cluster must meet the following prerequisites. These prerequisites are required because they allow Ambari to know whether the cluster is in a healthy operating mode and can be successfully managed from Ambari.

Table 1: Ambari-managed HDF Upgrade Prerequisites

Disk Space	Be sure to have adequate space on /usr/hdf for the target HDF installation.
Ambari Agent Heartbeats	All Ambari Agents must be communicating and heartbeating to Ambari Server. Any hosts that are not heartbeating must be in Maintenance Mode.
Host Maintenance Mode	The following two scenarios are checked: • Any hosts in Maintenance Mode must not be hosting any Service Master Components. • Any host in Maintenance Mode that is not hosting Master Components is allowed but you will receive a warning. You can proceed with your upgrade but these hosts will not be upgraded and before you can finalize the upgrade, you must delete the hosts from the cluster.
Service Maintenance Mode	No Services can be in Maintenance Mode.
Services Started	All Services must be started.
Service Checks	All Service Checks must pass. Be sure to run Service Actions > Run Service Check on all services (and remediate if necessary) prior to attempting an HDF upgrade.

Registering Your Target Version

Registering your target version makes Ambari aware of the Hortonworks stack to which you want to upgrade, provides the public repository location, and specifies your public or private repository delivery preference.

Procedure

1. Click the **Admin** tab, and then click **Stack and Versions**.
2. Click the **Versions** tab.
3. Click the **Manage Versions** button.
4. Click the **+ Register Version** button.
5. Select the target version you want to register, specify whether it will be a public or private repository, and select your operating system.
6. Click **Save**.

Results

From the **Versions** tab, you now see your target HDF version registered, but not yet installed.

Installing Your Target Version

Installing your target version downloads the public repositories containing software packages for your target version onto each node in your cluster.

Procedure

1. From the **Versions** tab, identify the target version you just registered, and click the **Install on ...** button.
2. Click **OK** to confirm.
3. You can monitor the progress of the install by clicking **Installing**.

Results

When the installation completes, you are able to see both your current and target HDF versions from **Admin | Stack and Versions | Versions**. Your target version has an active **Upgrade** button.

Upgrade Ambari Metrics

About this task

Procedure

1. Confirm that Ambari Metrics service is stopped and in Maintenance Mode.

   If Ambari Metrics service is not stopped, from Ambari Web, browse to Services > Ambari Metrics and select Stop from the Service Actions menu.
2. On every host in your cluster running a Metrics Monitor, run the following commands:

   For RHEL/CentOS/Oracle Linux:

   ```
   yum clean all
   ```

   ```
   yum upgrade ambari-metrics-monitor ambari-metrics-hadoop-sink
   ```

   For SLES:

   ```
   zypper clean
   ```

   ```
   zypper up ambari-metrics-monitor ambari-metrics-hadoop-sink
   ```

   For Ubuntu/Debian:

   ```
   apt-get clean all
   ```

   ```
   apt-get update
   ```

   ```
   apt-get install ambari-metrics-assembly
   ```
3. Execute the following command on all hosts running the Metrics Collector:

For RHEL/CentOS/Oracle Linux:

```
yum upgrade ambari-metrics-collector
```

For SLES:

```
zypper up ambari-metrics-collector
```

For Ubuntu/Debian:

```
apt-get clean all
```

```
apt-get update
```

```
apt-get install ambari-metrics-collector
```

4. Execute the following command on the host running the Grafana component:

   For RHEL/CentOS/Oracle Linux:

   ```
   yum upgrade ambari-metrics-grafana
   ```

   For SLES:

   ```
   zypper up ambari-metrics-grafana
   ```

   For Ubuntu/Debian:

   ```
   apt-get clean all
   ```

   ```
   apt-get update
   ```

   ```
   apt-get install ambari-metrics-grafana
   ```

   Note: DO NOT START the Ambari Metrics System service. It will be started automatically during the HDP upgrade process.

Backup and Upgrade Ambari Infra

The Ambari Infra Solr instance is used to index data for Ranger, and Log Search. The version of Solr used by Ambari Infra in Ambari 2.6 is Solr 5. The version of Solr used by the Ambari Infra in Ambari 2.7 is Solr 7. When moving from Solr 5 to Solr 7 indexed data needs to be backed up from Solr 5, migrated, and restored into Solr 7 as there are on disk format changes, and collection-specific schema changes. The Ambari Infra Solr components must also be upgraded. Fortunately scripts are available to do both, and are explained below.

This process will be broken up into four steps:

Generate Migration Config

The migration utility requires some basic information about your cluster and this step will generate a configuration file that captures that information.

Back up Ambari Infra Solr Data

This process will backup all indexed data either to a node-local disk, shared disk (NFS mount), or HDFS filesystem.

Remove existing collections & Upgrade Binaries

This step will remove the Solr 5 collections, upgrade Ambari Infra to Solr 7, and create the new collections

	with the upgraded schema required by HDP 3.0 services. This step will also upgrade LogSearch binaries if they are installed.
Migrate & Restore	This step will migrate the backed up data to the new format required by Solr 7 and restore the data into the new collections. This step will be completed after the HDP 3.0 Upgrade has been completed in the Post-upgrade Steps section of the upgrade guide

Generate Migration Config

The utility used in this process is included in the ambari-infra-solr-client package. This package must be upgraded before the utility can be run. To do this:

1. SSH into a host that has a Infra Solr Instance installed on it. You can locate this host by going to the Ambari Web UI and clicking Hosts. Click on the Filter icon and type Infra Solr Instance: All to find each host that has an Infra Solr Instance installed on it.
2. Upgrade the ambari-infra-solr-client package.

   ```
   yum clean all
   ```

   ```
   yum upgrade ambari-infra-solr-client -y
   ```
3. If you are using a custom username for running Infra Solr, for example a username that is not ‘infra-solr’ additional scripts need to be downloaded. To do this, again only if you are using a custom username for Infra Solr, perform the following steps:

   a.
   ```
   wget --no-check-certificate -O
   /usr/lib/ambari-infra-solr-client/migrationConfigGenerator.py
   https://raw.githubusercontent.com/apache/ambari/trunk/ambari-infra/
   ambari-infra-
   solr-client/src/main/python/migrationConfigGenerator.py
   ```
   b.
   ```
   chmod +x /usr/lib/ambari-infra-solr-client/migrationConfigGenerator.py
   ```
   c.
   ```
   wget --no-check-certificate -O /usr/lib/ambari-infra-solr-client/
   migrationHelper.py
   https://raw.githubusercontent.com/apache/ambari/trunk/ambari-infra/
   ambari-infra-
   solr-client/src/main/python/migrationHelper.py
   ```
   d.
   ```
   chmod +x /usr/lib/ambari-infra-solr-client/migrationHelper.py
   ```
4. You can now proceed to configuring and running the migration tool from the same host.

   Run the following commands as root, or with a user that has sudo access:

   Export the variable that will hold the full path and filename of the configuration file.

   ```
   export CONFIG_INI_LOCATION=ambari_solr_migration.ini
   ```

   Ensure the script generates cleanly and there are no yellow warning texts visible. If so, review the yellow warnings.
5. Run the migrationConfigGenerator.py script, located in the /usr/lib/ambari-infra-solr-client/ directory, with the following parameters:

--ini-file $CONFIG_INI_LOCATION	This is the previously exported environmental variable that holds the path and filename of the configuration file that will be generated.
--host ambari.hortonworks.local	This should be the hostname of the Ambari Server.
--port 8080	This is the port of the Ambari Server. If the Ambari Server is configured to use HTTPS, please use the HTTPS port and add the -s parameter to configure HTTPS as the communication protocol.
--cluster cl1	This is the name of the cluster that is being managed by Ambari. To find the name of your cluster, look in the upper right and corner of the Ambari Web UI, just to the left of the background operations and alerts.
--username admin	This is the name of a user that is an “Ambari Admin” .
--password admin	This is the password of the aforementioned user.
--backup-base-path=/my/path	This is the location where the backed up data will be stored. Data will be backed up to this local directory path on each host that is running an Infra Solr instance in the cluster. So, if you have 3 Infra Solr server instances and you use --backup-base-path=/home/solr/backup, this directory will be created on all 3 hosts and the data for that host will be backed up to this path. If you are using a shared file system that is mounted on each Infra Solr instance in the cluster, please use the --shared-drive parameter instead of --backup-base-path. The value of this parameter should be the path to the mounted drive that will be used for the backup. When this option is chosen, a directory will be created in this path for each Ambari Infra Solr instance with the backed up data. For example, if you had an NFS mount /export/solr on each host, you would use --shared-drive=/exports/solr. Only use this option if this path exists and is shared amongst all hosts that are running the Ambari Infra Solr.
--java-home /usr/jdk64/jdk1.8.0_112	This should point to a valid Java 1.8 JDK that is available at the same path on each host in the cluster that is running an Ambari Infra Solr instance.
If the Ranger Audit collection is being stored in HDFS, please add the following parameter, --ranger-hdfs-base-path	The value of this parameter should be set to the path in HDFS where the Solr collection for the Ranger Audit data has been configured to store its data. Example: --ranger-hdfs-base-path=/user/infra-solr

Example Invocations:

If using HTTPS for the Ambari Server:

```
/usr/bin/python /usr/lib/ambari-infra-solr-client/
migrationConfigGenerator.py
```

```
              --ini-file $CONFIG_INI_LOCATION --host
 c7401.ambari.apache.org --port 8443 -s
              --cluster cl1 --username admin --password admin --backup-
base-path=/my/path
              --java-home /usr/jdk64/jdk1.8.0_112
```

If using HTTP for the Ambari Server:

```
/usr/bin/python /usr/lib/ambari-infra-solr-client/
migrationConfigGenerator.py
              --ini-file $CONFIG_INI_LOCATION --host
 c7401.ambari.apache.org --port 8080 --cluster
              cl1 --username admin --password admin --backup-base-path=/
my/path --java-home
              /usr/jdk64/jdk1.8.0_112
```

Back up Ambari Infra Solr Data

Once the configuration file has been generated, it’s recommended to review the ini file created by the process. There is a configuration section for each collection that was detected. If, for whatever reason, you do not want to backup a specific collection you can set enabled = false and the collection will not be backed up. Ensure that enabled = true is set for all of the collections you do wish to back up. Only the Atlas, and Ranger collections will be backed up. Log Search will not be backed up.

To execute the backup, run the following command from the same host on which you generated the configuration file:

```
# /usr/lib/ambari-infra-solr-client/ambariSolrMigration.sh --ini-file
 $CONFIG_INI_LOCATION
        --mode backup | tee backup_output.txt
```

During this process, the script will generate Ambari tasks that are visible in the Background Operations dialog in the Ambari Server.

Once the process has completed, please retain the output of the script for your records. This output will be helpful when debugging any issues that may occur during the migration process, and the output contains information regarding the number of documents and size of each backed up collection.

Remove Existing Collections & Upgrade Binaries

Once the data base been backed up, the old collections need to be deleted, and the Ambari Infra Solr, and Log Search (if installed) components need to be upgraded. To do all of that, run the following script:

```
# /usr/lib/ambari-infra-solr-client/ambariSolrMigration.sh --ini-file
 $CONFIG_INI_LOCATION
        --mode delete | tee delete_output.txt
```

During this process, the script will generate Ambari tasks that are visible in the Background Operations dialog in the Ambari Server.

Once the process has completed, please retain the output of the script for your records. This output will be helpful when debugging any issues that may occur during the migration process.

Upgrade Ambari Log Search

If you have Ambari Log Search installed, you must upgrade Ambari Log Search after upgrading Ambari.

Before you begin

Before starting this upgrade, ensure the Ambari Infra components have been upgraded.

Procedure

1. Make sure Ambari Log Search service is stopped. From Ambari Web, browse to Services > Log Search and select Stop from the Service Actions menu.
2. On every host in your cluster running a Log Feeder, run the following commands:

   For RHEL/CentOS/Oracle Linux:

   ```
   yum clean all
   ```

   ```
   yum upgrade ambari-logsearch-logfeeder
   ```

   For SLES:

   ```
   zypper clean
   ```

   ```
   zypper up ambari-logsearch-logfeeder
   ```

   For Ubuntu/Debian:

   ```
   apt-get clean all
   ```

   ```
   apt-get update
   ```

   ```
   apt-get install ambari-logsearch-logfeeder
   ```
3. Execute the following command on all hosts running the Log Search Server:

   For RHEL/CentOS/Oracle Linux:

   ```
   yum upgrade ambari-logsearch-portal
   ```

   For SLES:

   ```
   zypper up ambari-logsearch-portal
   ```

   For Ubuntu/Debian:

   ```
   apt-get install ambari-logsearch-portal
   ```
4. Start Log Search Service.

   From Ambari Web, browse to Services > Log Search select Service Actions then choose Start.

Verifying Symbolic Links for SAM and Schema Registry

After you register and install your target version, but before you proceed with an Express Upgrade, verify that the symbolic links to the SAM and Schema Registry configuration directories on each host are still valid. If the links are not valid, fix them before upgrading.

Procedure

1. Confirm on each host running Registry or SAM that symlinks for the configuration directories are still valid.

   ```

   #For Schema Registry
   ls -l /etc/registry/conf

   #For SAM
   ```

```
ls -l /etc/streamline/conf
```

2. If the link appears to be broken, perform the following steps to fix it.

   a. Remove the broken symbolic link.

   ```

   #Remove broken Schema Registry link
   rm -rf /etc/registry/conf

   #Remove broken SAM link
   rm -rf /etc/streamline/conf
   ```

   b. Create a physical SAM or Schema Registry configuration directory.

   ```

   #Create a Schema Registry conf directory
   mkdir -p /etc/registry/conf

   #Create SAM conf directory
   mkdir -p /etc/streamline/conf
   ```

   c. Copy the configuration files from your backup, into the new directory.

   ```

   #Copy Schema Registry backup files
   cp /etc/registry/conf.backup/* /etc/registry/conf

   #Copy SAM backup files
   cp /etc/streamline/conf.backup/* /etc/streamline/conf
   ```

Upgrade HDF

Upgrading HDF installs your target software version onto each node in your cluster. Note that the Express Upgrade is the only option available to HDF 3.2.

About this task

Rolling Upgrade is not supported by NiFi. When you select the Rolling Upgrade option for the HDF stack, NiFi stops all services, and performs an Express Upgrade. Express Upgrades stops the NiFi Service completely, and restarts it with the new version installed.

Before you begin

Turn off maintenance mode for LogSearch and Ambari Metrics if required action appears when attempting to upgrade stack.

Procedure

1. From **Admin | Stack and Versions | Versions**, click **Upgrade**.
2. In the **Upgrade Options** pop-up window, click **Express Upgrade**, and specify if you would like customized upgrade failure tolerance. If you select:
   - **Skip all Service Check failures** – Ambari skips any Service Check failures and completes the upgrade without requiring user intervention to continue. After all the Services have been upgraded Checks, you are presented with summary of the failures and an option to continue the upgrade or pause.
   - **Skip all Slave Component failures** – Ambari skips any Slave Component failures and completes the Slave components upgrade without requiring user intervention to continue. After all Slave Components have been upgraded, you are presented with a summary of the failures and an option to continue the upgrade or pause.
3. Click **Proceed**.

4. Once the upgrade completes, again confirm that you have performed the required manual steps and click **Finalize**.

Results

From **Admin | Stack and Versions | Versions**, you are now able to see only the HDF version to which you upgraded.

Start Ambari LogSearch and Metrics

After you have upgraded your HDF cluster, you should ensure that Ambari LogSearch and Metrics are both running.

Procedure

1. From the Ambari UI, verify whether Ambari LogSearch and Metrics are running.
2. If they are not running, manually start them before proceeding.

Migrate and Restore Ambari Infra

Follow the steps below to restore the data you previously backed up.

Procedure

1. SSH to the host where the migrationConfigGenerator.py was run prior to the HDP Upgrade. This will be from one of your Ambari Infra Solr instances. Please ensure you are in the current working directory containing theambari_solr_migration.ini file.
2. Export the variable used to hold the path to the ini file.

   ```
   export CONFIG_INI_LOCATION=ambari_solr_migration.ini
   ```

3. Migrate the data to ensure it’s in the right format to import into Solr 7. Please note that this script can take a long time to run depending on the volume of data backed up. It is recommended to run this script using the nohup command.

   ```
   nohup /usr/lib/ambari-infra-solr-client/ambariSolrMigration.sh --ini-file
               $CONFIG_INI_LOCATION --mode migrate-restore >
    migrate_restore_output.txt 2>&1
             &
   ```

   To observe the progress of the migrate and restore process, simply tail the migrate_restore_output.txt file.

   ```
   tail -f migrate_restore_output.txt
   ```

4. Re-index the migrated data into your current collections so the backed up data is visible in all of the tools using the Infra Solr instances. Please note that this script can take a long time to run depending on the volume of data backed up. It is recommended to run this script using the nohup command.

   ```
   nohup /usr/lib/ambari-infra-solr-client/ambariSolrMigration.sh --ini-file
               $CONFIG_INI_LOCATION --mode transport > transport_output.txt
    2>&1 &
   ```

   To observe the progress of the transport process, simply tail the transport_output.txt file:

   ```
   tail -f transport_output.txt
   ```

Migrate Ambari Metrics Data

Use the following steps to migrate data from the previous AMS schema to the new AMS schema.

Procedure

1. Ensure the Ambari Metrics System is started. If it is not, in the Ambari Web UI, click Ambari Metrics, then select Actions > Start.
2. SSH into a host that is running a Metrics Collector. You can locate this host by going to the Ambari Web UI and clicking Hosts. Click on the Filter icon and type in "Metrics Collector: All" to find each host that has a Metrics Collector installed on it.
3. SU to the Ambari Metrics user. This is 'ams' by default, but if you don’t know which user is configured in your cluster go to the Ambari Web UI and click Cluster Admin > Service Accounts, and then look for "Ambari Metrics User".

```
# su ams
```

4. Run the command to migrate data from the old Ambari Metrics schema to the new.

```
$ /usr/sbin/ambari-metrics-collector --config /etc/ambari-metrics-
collector/conf/
            upgrade_start /etc/ambari-metrics-collector/conf/
metrics_whitelist
```

5. Once the upgrade process has started, logs are available in the <ams-log-dir>/ambari-metrics-migration.log file.

Update Ranger Passwords

Ranger password validation has been updated for HDF 3.2.0, and to conform to these new password policies, the following Ranger passwords need to be updated to ensure that they have at least 8 characters with minimum one alphabet and one numeric. These passwords cannot contain the following special characters: " ' \ `

- Ranger Admin
- Ranger Ambari Admin

The following new passwords need to be populated with valid passwords that also conform to the password policy:

- Ranger Usersync User’s Password
- Ranger Tagsync User’s Password
- Ranger KMS Keyadmin User’s Password

Upgrading HDF 3.0.2 services on an HDP cluster

Upgrade HDP

Before you begin

You have obtained the necessary HDP, HDP UTILs, and HDF base URLs from the *HDF Release Notes*.

Procedure

1. In Ambari, go to **Manage Versions** and register your new HDP version.
2. Select HDP 3.0.0 and update base urls for HDP/HDF/ HDP UTILs.
3. Install the new packages from registered version HDP 3.0.0.0.
4. Perform an express upgrade to HDP 3.0.0 as described in the HDP upgrade documentation.

Related Information

Upgrading HDP

HDF Release Notes

Upgrade HDF services

Before you begin

Ensure that you have backed up your SAM and Schema Registry databases.

Procedure

1. Confirm HDF components installed along with version (e.g. 3.0.2.0-76).

```
[root@host registry]# yum list installed | grep HDF

hdf-select.noarch                 3.0.2.0-76.el6           @HDF-3.0

nifi_3_0_2_0_76.x86_64           1.2.0.3.0.2.0-76.el6     @HDF-3.0

                                  0.7.0.3.0.2.0-76.el6     @HDF-3.0

registry_3_0_2_0_76.noarch       0.3.0.3.0.2.0-76.el6     @HDF-3.0

storm__3_0_2_0_76.x86_64          1.1.0.3.0.2.0-76.el6     @HDF-3.0

streamline_3_0_2_0_76.x86_64     0.5.0.3.0.2.0-76.el6     @HDF-3.0

zookeeper_3_0_2_0_76.noarch      3.4.6.3.0.2.0-76.el6     @HDF-3.0
```

2. Display the current version associated with each component (in following example 3.0.2.0-76 is current version).

```
[root@host registry]# hdf-select status | grep 3.0.2.0-76

nifi - 3.0.2.0-76
registry - 3.0.2.0-76
storm-client - 3.0.2.0-76
storm-nimbus - 3.0.2.0-76
storm-supervisor - 3.0.2.0-76
streamline - 3.0.2.0-76
zookeeper-client - 3.0.2.0-76
zookeeper-server - 3.0.2.0-76
```

3. Install binaries for the HDF services to which you want to upgrade.

   See the *HDF Release Notes* for the repository information, including service version and build number.

```
[root@host ~]# yum install -y <service>_<version>_<build-number>*
```

   For example:

```
[root@host ~]# yum install -y nifi_3_2_0_0_520*
```

   Note: Only run the yum install command on Ambari Agent hosts where NiFi is already installed.

4. Use hdf-select to ensure appropriate links to new installed version. Note that SAM also brings down Storm and ZooKeeper dependencies that also needs to be accounted for.

   For example,

```
[root@host ~]# hdf-select set nifi 3.2.0.0-<build-no>
[root@host ~]# hdf-select set registry 3.2.0.0-<build-no>
[root@host ~]# hdf-select set streamline 3.2.0.0-<build-no>
[root@host ~]# hdf-select set storm-nimbus 3.2.0.0-<build-no>
[root@host ~]# hdf-select set storm-supervisor 3.2.0.0-<build-no>
[root@host ~]# hdf-select set zookeeper-client 3.2.0.0-<build-no>
```

```
WARNING: Replacing link /usr/bin/zookeeper-client from /usr/hdp/current/
zookeeper-client/bin/zookeeper-client

[root@host ~]# hdf-select set zookeeper-server 3.2.0.0-<build-no>

WARNING: Replacing link /usr/bin/zookeeper-server from /usr/hdp/current/
zookeeper-server/bin/zookeeper-server
WARNING: Replacing link /usr/bin/zookeeper-server-cleanup from /usr/hdp/
current/zookeeper-server/bin/zookeeper-server-cleanup
```

5. Confirm that hdf select shows new version for hdf components that were updated.

```
[root@host ~]# hdf-select status | grep 3.2.0.0-<build-no>
nifi - 3.2.0.0-<build-no>
registry - 3.2.0.0-<build-no>
storm-nimbus - 3.2.0.0-<build-no>
storm-supervisor - 3.2.0.0-<build-no>
streamline - 3.2.0.0-<build-no>
zookeeper-client - 3.2.0.0-<build-no>
zookeeper-server - 3.2.0.0-<build-no>
```

6. Log into Ambari and start NiFi, NiFi Registry, SAM and Schema Registry. Confirm all applications started and pre-existing flows/topologies/configurations are available.

Upgrading HDF 3.1.0 services on an HDP cluster

Upgrade HDP

Before you begin

You have obtained the necessary HDP, HDP UTILs, and HDF base URLs from the *HDF Release Notes*.

Procedure

1. In Ambari, go to **Manage Versions** and register your new HDP version.
2. Select HDP 3.0.0 and update base urls for HDP/HDF/ HDP UTILs.
3. Install the new packages from registered version HDP 3.0.0.0.
4. Perform an express upgrade to HDP 3.0.0 as described in the HDP Upgrade documentation.

Related Information

Upgrading HDP

HDF Release Notes

Upgrade HDF services

About this task

Procedure

1. Confirm that the NiFi components are installed along with version (3.1.0.0).

```
[root@host registry]# yum list installed | grep nifi
nifi_3_1_0_0_<build-no>.x86_64
nifi-registry_3_1_0_0_<build-no>.x86_6
```

2. Display the current version associated with each component. In the following example, 3.1.0.0 is current version.

   ```
   [root@host registry]# hdf-select status | grep nifi
   nifi - 3.1.0.0-<build-no>
   nifi-registry - 3.1.0.0-<build-no>
   ```

3. Install binaries for the HDF services to which you want to upgrade.

   See the *HDF Release Notes* for the repository information, including service version and build number.

   ```
   [root@host ~]# yum install -y <service>_<version>_<build-number>*
   ```

   For example:

   ```
   [root@host ~]# yum install -y nifi_3_2_0_0_520*
   ```

   Note: Only run the yum install command on Ambari Agent hosts where NiFi is already installed.

4. Use hdf-select to ensure appropriate links to new installed version.

   ```
   [root@host ~]# hdf-select set nifi 3.2.0.0-<build-no>
   [root@host ~]# hdf-select set nifi -registry 3.2.0.0-<build-no>
   ```

5. Confirm that hdf-select shows the new version for the HDF services you updated.

   ```
   [root@host ~]# hdf-select status | grep 3.2.0.0
   nifi - 3.2.0.0-<build-no>
   nifi-registry - 3.2.0.0-<build-no>
   ```

6. Log into Ambari and start NiFi and NiFi Registry.
7. You may add Schema Registry or SAM to your cluster, as needed.

Post-Upgrade Tasks

Restarting NiFi Certificate Authority

After you have upgraded to your target HDF version, you will need to restart the NiFi Certificate Authority (CA).

Procedure

1. From **Services | NiFi | Configs**, click **Restart**.
2. Click **Confirm Restart All**.

Review Storm Configurations

If you have configured STORM_EXT_CLASSPATH or STORM_EXT_CLASSPATH_DAEMON prior to upgrade, you must update the values to add a wild card.

Procedure

1. From the left-hand services navigation pane, select **Storm | Configs | Advance**
2. Update the STORM_EXT_CLASSPATH or STORM_EXT_CLASSPATH_DAEMON values to include a wild card.

Example

If STORM_EXT_CLASSPATH=/foo/bar/lib, update the value to STORM_EXT_CLASSPATH=/foo/bar/lib/*.

Adding New NiFi Provenance Policies

Apache NiFi 1.7.0 introduces new policies for controlling access to provenance events from a component. When upgrading to Apache NiFi 1.7.0 these new policies will not exist. As a result, users who could previously access provenance events will no longer have access to these provenance events. To ensure continued user access to provenance events, a NiFi administrator must update NiFi policies after upgrade.

About this task

By introducing the new view provenance events policy for controlling access to the event itself these operators can better understand the dataflow and track what is happening while the administrators can still maintain tight control of the flowfile attributes and content. To avoid users losing access to provenance event information, perform the following steps:

Procedure

1. Create new view provenance policies for your components.
2. Assign users to the new policies.

Results

Users can now resume provenance event access.

NiFi Component Property Name Changes

Some NiFi component properties have been renamed.

Old properties become unsupported user-defined properties and the components become invalid. To make the components valid again, migrate the values from the old properties to the new ones, and then remove the old properties.

Component	Old property	New property
ReportLineageToAtlas	kafka-kerberos-service-name-kafka	kafka-kerberos-service-name
GetCouchbaseKey and PutCouchbaseKey	Couchbase Cluster Controller Service	cluster-controller-service
	Bucket Name	bucket-name
	Document Type	document-type
	Document Id	document-id
	Persist To	persist-to
	Replicate To	replicate-to